BENDING EARLY, CURBING DEEP: TEST-TIME ADVERSARIAL DEFENSE FOR LARGE VISION-LANGUAGE MODELS BASED ON RESIDUAL CURVATURE

Anonymous authors
Paper under double-blind review

ABSTRACT

Although Large Visual-Language Models (LVLMs) have achieved significant advances, their vulnerability to adversarial perturbations poses a persistent challenge to their reliable deployment in the real world. Prevalent test-time defenses largely focus on mitigating adversarial effects while overlooking how adversarial deviations propagate through the ViT-based visual encoder. In particular, their reliance on multi-view transformations and closed-set class space limits their applicability to generative tasks in LVLMs. In this paper, we find that adversarial perturbations first induce a geometric abnormality of early residual trajectories by shifting them from differential toward common component, with the resulting deviations subsequently amplified predominantly through deep MLP branches. Therefore, we propose Curvature-based Universal Residual Bending (CURB), a test-time defense that selectively bends abnormally flattened residual trajectories through input updates to curb adversarial growth. CURB does not depend on downstream outputs space, allowing universal and direct operation across open-ended generation and zero-shot classification. Experiments on LLaVA-1.5-7B and InternVL3.5-8B show consistent robustness gains in various tasks, together with substantial reductions in adversarial representation deviations. Codes are provided in Section 5.

1 INTRODUCTION

Large vision-language models (LVLMs) have demonstrated strong multimodal reasoning across downstream tasks such as image captioning (Lin et al., 2014), visual question answering (Singh et al., 2019), and zero-shot classification (Deng et al., 2009). Despite recent progress, they remain vulnerable to even subtle adversarial perturbations, raising serious safety concerns for real-world deployment (Xiong et al., 2025; Yu et al., 2026). Adversarial training (Madry et al., 2018) has proven to be an effective defense strategy. However, additional training and modification of the pretrained visual backbone can induce a substantial performance gap between clean and adversarial inputs.

Test-time defenses (Li et al., 2026; Hai et al., 2026; Kim & Um, 2026a) instead operate directly at inference time and therefore provide a more flexible alternative without retraining the model. However, most existing test-time defenses are developed primarily for CLIP-based classification and do not naturally extend to open-ended LVLMs inference. First, multi-view transformations (Sheng et al., 2025; Liu et al., 2026c) are widely combined with various optimization or prediction for robustness. While affordable for classification, they are particularly expensive for LVLMs, as transformed inputs require additional autoregressive decoding process. Second, most optimization objectives (Zhu et al., 2025; Liu et al., 2026b) are primarily designed for CLIP-based classifiers operating over a closed class space. Such closed-set objectives are difficult to transfer to LVLMs, whose outputs are autoregressively generated sequences without a shared candidate space. Although lightweight test-time defenses (Xing et al., 2025; Mirza et al., 2026) have recently been explored, they typically provide limited robustness under stronger attacks compared with transformation-based methods. More fundamentally, existing methods largely operate on the model outputs while overlooking how a small input perturbation propagates through the internal representations.

Since ViT-based visual encoders with global self-attention are among the most widely adopted configurations in LVLMs, we focus on this setting and trace the deviation between clean and adversarial

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

representations layer by layer throughout the visual encoder. Our analysis reveals a uniform feature corruption. The clean–adversarial representation discrepancy grows toward deeper layers, and nearly all positive discrepancy growth in the deep layers is contributed by the MLP branches. We then trace this deep discrepancy backward to the early geometry of residual propagation. Specifically, adversarial inputs exhibit reduced discrete curvature between the early consecutive residual updates, indicating an abnormally flattened trajectory with suppressed directional differentiation. Geometrically, this curvature reduction shifts the early residual updates from the differential component toward common component. And we further find that their deep MLP branches exhibit a stronger response to the common residual component. These observations reveal a connection between an early residual geometric abnormality and the subsequent deep MLP discrepancy growth. Moreover, we suggest that a finite-step increase in early residual curvature curbs the discrepancy amplification contributed by downstream MLP branches in this situation.

To this end, we propose Curvature-based Universal Residual Bending (CURB), a test-time defense that curbs adversarial propagation at its earliest measurable directional abnormality. CURB measures early residual updates of the frozen visual encoder and selectively activates bending when the residual trajectory is abnormally flattened. For activated inputs, CURB performs pixel updates to increase directional variation in the early residual trajectory before adversarial discrepancies are further amplified in deeper layers. The optimization requires gradients only through the first few layers, and operates independently of downstream output space. Consequently, the early residual curvature-based bending is universal across tasks and can be directly applied to both closed-set classification and open-ended generation. Extensive experiments demonstrate that CURB achieves consistent robustness improvements across VQA, caption, and classification tasks under various attack settings.

2 Rethinking Test-Time Defense for LVLMs

In this section, we first discuss the limitations of existing test-time defenses in Section 2.1, and summarize them in Section A.2. We then study adversarial propagation in the latent space in Section 2.2. Experiments are conducted on TextVQA under 100-step APGD attacks (Schlarmann et al., 2024), with an ℓ_∞ perturbation budget of $4/255$, using two LVLMs: LLaVA-1.5-7B (Liu et al., 2024) and InternVL3.5-8B (Wang et al., 2025a). Detailed settings is provided in Appendix C.1.

2.1 Limitations of Test-Time Defenses for Vision-Language Models

Closed-Set Objectives Do Not Transfer to Open-Ended Generation Existing test-time defenses (Hai et al., 2026; Liu et al., 2026c; Mirza et al., 2026) are predominantly designed for classification because CLIP-ViT-based classifiers operate over a closed class space. Let c_k denote the k-th class label, given an image x, the prediction distribution is defined as:

$$[P(y = c_k \mid x)]_{k=1}^{C} = \mathrm{softmax}\Big([\mathrm{sim}(f_{\mathrm{I}}(x), f_{\mathrm{T}}(c_k))]_{k=1}^{C}\Big), \tag{1}$$

where every test sample shares a fixed candidate set $\mathbb{Y} = \{c_1, c_2, \ldots, c_C\}$. This closed-set formulation provides a normalized probability distribution over aligned semantic candidates, making it straightforward to build stable optimization objectives. In contrast, an LVLM produces an open-ended sequence conditioned on the image x and question q:

$$P(y_{1:T_l} \mid x, q) = \prod_{t=1}^{T_l} P(y_t \mid x, q, y_{<t}), \tag{2}$$

where T_l is the generated-sequence length, y_t is the token generated at step t, and $y_{<t}$ denotes all preceding tokens. Since semantically equivalent answers may differ in tokens, sequence lengths, and linguistic expressions, token-level probabilities from different generations are not naturally aligned, making direct sequence-level comparison difficult.

Multi-View Inference Is Prohibitively Expensive for LVLMs Constructing multiple augmented views, a widely used strategy, makes the computational cost particularly prohibitive for generative LVLMs. For CLIP-based classification, the complexity of K-view inference consists of visual encoding and similarity computation with C class features:

$$\mathcal{O}\big(K\left[L_v\left(N_v^2 d_v + N_v d_v^2\right) + C d_e\right]\big). \tag{3}$$

But for VQA and captioning, each view additionally requires a complete language-model inference:

$$\mathcal{O}\big(K\left[L_v\left(N_v^2 d_v + N_v d_v^2\right) + L_l\left(N_0^2 d_l + N_0 d_l^2\right) + L_l\left(T_l N_0 d_l + T_l^2 d_l + T_l d_l^2\right)\right]\big). \tag{4}$$

Here L_v and L_l denote the numbers of visual and language layers, N_v and N_0 denote the visual-token and multimodal context lengths before decoding, d_v and d_l denote the corresponding hidden dimensions, d_e denotes embedding dimension. Thus, every additional view introduces an expensive autoregressive decoding process for LVLMs. Detailed complexity is provided in Appendix B.1.

Internal Adversarial Propagation Remains Overlooked Recent lightweight defenses (Xing et al., 2025; Jiang et al., 2026; Zhang et al., 2025) directly use deviations in the final visual representation as optimization signals. However, such endpoint-based objectives determine rectification directions using proxy signals estimated from corrupted visual representations, making them vulnerable under stronger attacks (Hai et al., 2026).

2.2 Tracing Adversarial Propagation through the Visual Encoder.

Observation 1: Deep Adversarial Discrepancy Growth Is Dominated by MLP Branches To characterize altered feature propagation through the visual encoder between a pair of clean x_c and adversarial x_a input, we define the clean-adversarial squared Frobenius discrepancy at layer l as:

$$E_l(x_a, x_c) = \|\boldsymbol{H}_l(x_a) - \boldsymbol{H}_l(x_c)\|_F^2, \tag{5}$$

where $\boldsymbol{H}_l(x)$ denotes the aligned visual token representation at the l-th layer. As shown in Figure 1 (a), the clean-adversarial representation discrepancy grows toward deeper layers. This indicates that the pixel-level perturbation constrained at the input progressively evolves into larger deep feature discrepancies during propagation in the visual encoder.

To identify where the additional discrepancy is introduced, we decompose each visual Transformer block into its Attention and MLP residual updates.

Proposition 1. *For any two visual Transformer layers $s < L$, let G_l^{Attn} and G_l^{MLP} denote the contributions of the Attention and MLP residual branches to the discrepancy change at layer l. Then, the change in squared Frobenius discrepancy between layers s and L admits the exact decomposition*

$$E_L(x_a, x_c) - E_s(x_a, x_c) = \sum_{l=s}^{L-1} G_l^{\mathrm{Attn}}(x_a, x_c) + \sum_{l=s}^{L-1} G_l^{\mathrm{MLP}}(x_a, x_c). \tag{6}$$

This decomposition is an exact algebraic identity and does not rely on linearizing the Transformer block or approximating its Jacobian. The complete derivation is provided in the Appendix B.2.

Based on this decomposition, we further analyze the cumulative discrepancy contributions across the last three visual Transfomer layers. Figure 1 (b) shows that, over the deep interval, nearly all of the positive discrepancy growth is contributed from the MLP terms. Therefore, the additional late-stage adversarial deviation does not accumulate uniformly across Transformer blocks, but is overwhelmingly expressed via the MLP residual branches.

Observation 2: Adversarial Perturbations Suppress Early Residual Directional Differentiation We next examine whether adversarial perturbations alter the local geometry of representation propagation already in the early visual layers. Consider two consecutive full-block residual updates, denoted as the first and second respectively: $\boldsymbol{U}_1(x) = \boldsymbol{H}_2(x) - \boldsymbol{H}_1(x)$ and $\boldsymbol{U}_2(x) = \boldsymbol{H}_3(x) - \boldsymbol{H}_2(x)$, and define their residual discrete curvature as

$$C_2(x) = 1 - \cos\left(\boldsymbol{U}_1(x), \boldsymbol{U}_2(x)\right). \tag{7}$$

We focus on C_2 because discrete curvature requires two consecutive residual updates, making C_2 the earliest point at which a directional change in the residual trajectory can be measured. This allows us to probe adversarial propagation at its earliest observable geometric abnormality, before the deviation undergoes further transformations through deeper visual layers.

Proposition 2. *For two nonzero consecutive residual updates $\boldsymbol{U}_1(x)$ and $\boldsymbol{U}_2(x)$, let $q(x)$ denote the squared magnitude ratio between the differential component $\boldsymbol{R}(x)$ to the common component $\boldsymbol{M}(x)$ in their normalized directions. For $C_2(x) < 2$, it admits*

$$C_2(x) = \frac{2q(x)}{1 + q(x)}, \qquad \frac{\mathrm{d}C_2(x)}{\mathrm{d}q(x)} = \frac{2}{(1 + q(x))^2} > 0. \tag{8}$$

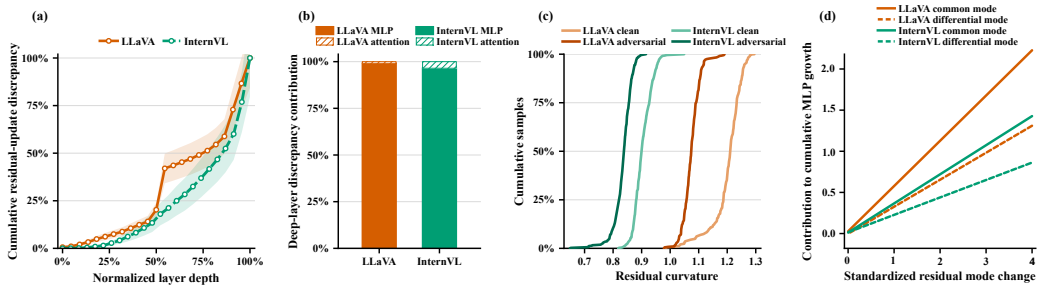

Figure 1: (a) Cumulative mean-squared Frobenius discrepancy $E_l(x_a, x_c)$ between clean and adversarial residual updates, normalized by the cohort-mean cumulative discrepancy at the final layer, across normalized layer depth. Shaded regions indicate ± 1 sample standard deviation. (b) The percentage share of the clean–adversarial discrepancy growth $\Delta E(x_a, x_c)$ over the last three layers attributable to attention $G_l^{\mathrm{Attn}}(x_a, x_c)$ and MLP $G_l^{\mathrm{MLP}}(x_a, x_c)$, respectively. (c) Smoothed empirical cumulative distributions of early residual curvature, defined as Equation 7. The vertical axis reports the percentage of samples whose curvature is at or below the corresponding horizontal-axis value. (d) Common- and differential-component contributions to cumulative MLP discrepancy growth $\sum_{l=s}^{L-1} G_l^{\mathrm{MLP}}(x_a, x_c)$ under $\Delta_m + \Delta_r = 0$, reconstructed from the response coefficients.

C_2 is therefore a strictly increasing function of the differential-to-common ratio $q(x)$. A smaller C_2 therefore means that the differential component is reduced relative to the common component between consecutive residual updates. The complete definition and proof are provided in Appendix B.3.

Based on this geometric characterization, we compare the early residual curvature between clean and adversarial inputs. Figure 1 (c) shows that adversarial samples consistently exhibit lower C_2 than their paired clean samples. This observation identifies an early structural abnormality that appears before the large deep-layer discrepancy: adversarial perturbations flatten the early residual trajectory, suppressing directional differentiation between consecutive early residual updates.

Observation 3: Deep MLPs Preferentially Amplify the Early Common Residual Component Observation 2 shows that adversarial perturbations reduce the early residual curvature $C_2(x)$, shifting the consecutive residual updates toward the common component $\boldsymbol{M}(x)$ while relatively suppressing the differential component $\boldsymbol{R}(x)$. However, this geometric change alone does not explain why the early abnormality eventually manifests as the deep MLP discrepancy growth observed in Observation 1. We therefore examine whether deep MLPs respond differently to these two early residual modes.

We characterize the deep MLP response using the cumulative MLP discrepancy contribution $\sum_{l=s}^{L-1} G_l^{\mathrm{MLP}}(x_a, x_c)$, and locally model its dependence on the common and differential residual modes as

$$\sum_{l=s}^{L-1} G_l^{\mathrm{MLP}}(x_a, x_c) = \gamma_m \Delta_m(x_a, x_c) + \gamma_r \Delta_r(x_a, x_c) \tag{9}$$

where $\Delta_m(x_a, x_c) = \|\boldsymbol{M}(x_a)\|_F^2 - \|\boldsymbol{M}(x_c)\|_F^2$ and $\Delta_r(x_a, x_c) = \|\boldsymbol{R}(x_a)\|_F^2 - \|\boldsymbol{R}(x_c)\|_F^2$ denote the clean-adversarial changes in the early common and differential residual components, respectively, while γ_m and γ_r quantify the response strengths of the cumulative deep MLP discrepancy growth to these two component-wise changes. Since the common and differential components satisfy $\Delta_m + \Delta_r = 0$ (Appendix B.3, Eqs. (40)–(41)), we retain the two terms only to explicitly expose their opposing contributions, while only the relative response $\gamma_m - \gamma_r$ is identifiable.

As shown in Figure 1 (d), consistent $\gamma_m > \gamma_r$ reveals a mode-selective response of the deep MLP branches: deviations along the early common residual mode receive a stronger response than deviations along the differential mode. From the geometric characterization in Observation 2, $C_2(x_a) < C_2(x_c)$ implies $\Delta_m(x_a, x_c) > 0$ and $\Delta_r(x_a, x_c) < 0$ (Details are provided in Appendix B.3). Therefore, adversarial perturbations effectively concentrate more early residual direction into the common mode, to which deep MLPs exhibit a higher response gain.

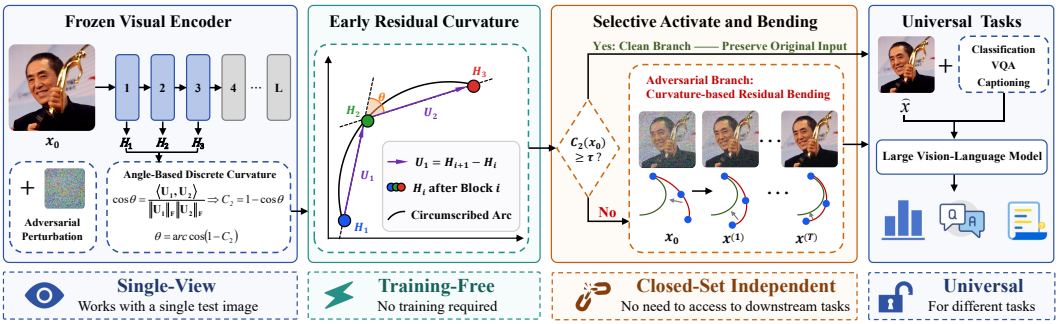

Figure 2: Given a test image x_0, CURB first extracts the residual states from the early visual Transformer blocks and measures the discrete curvature between two consecutive residual updates. After determining abnormally flattened trajectory, CURB updates the input pixels by increasing the residual curvature under a bounded perturbation constraint. Finally, the rectified image is passed to the original downstream model for task-specific inference.

Observation 4: Finite-Step Curvature Increase Curbs Deep MLP Discrepancy Amplification We further ask whether a finite-step increase in C_2 is sufficient to suppress the cumulative clean–adversarial discrepancy amplification through deep MLP branches.

Proposition 3. *Consider the ViT-based vision encoder with global self-attention in every layer. Assume that, along the considered optimization trajectory, the local mode-selective MLP response relation in Equation 9 holds with $\gamma_m > \gamma_r$. For any T-step update sequence $x_0, \ldots, x_T$ ($x_0 = x_a$) within an input region $\mathbb{X}$ and a fixed reference x_c, if*

$$C_2(x_T) > C_2(x_0), \tag{10}$$

then

$$\sum_{l=s}^{L-1} G_l^{\mathrm{MLP}}(x_T, x_c) < \sum_{l=s}^{L-1} G_l^{\mathrm{MLP}}(x_0, x_c). \tag{11}$$

Thus, finite-step increase in early residual curvature can reduce the cumulative MLP discrepancy contribution relative to the fixed clean reference. The proof is provided in Appendix B.4, and the experimental results are shown in Table 2.

3 Proposed Method

Our analysis in Section 2 shows that adversarial perturbations first manifest as an abnormally flattened residual trajectory in the early visual layers, before the resulting representation discrepancy is amplified in the deeper layers. Motivated by this observation, we propose Curvature-Based Universal Residual Bending (CURB) as shown in Figure 2, a test-time defense that rectifies the input image by increasing directional variation in its early residual trajectory.

3.1 Curvature-Guided Input Rectification.

Given the original test input x_0, CURB seeks a rectified image x whose early residual trajectory exhibits stronger directional variation. We formulate the rectification objective as

$$\max_{x} \; C_2(x) \qquad \text{s.t.} \quad \|x - x_0\|_\infty \le T\alpha, \quad x \in [0,1]^{H\times W\times 3}, \tag{12}$$

where T is the number of optimization steps and α is the per-step rectification magnitude. Starting from $x^{(0)} = x_0$, the input is iteratively updated as

$$x^{(t+1)} = \mathrm{clip}_{[0,1]}\left[x^{(t)} + \alpha \ \mathrm{sign}\left(\nabla_{x^{(t)}} C_2\left(x^{(t)}\right)\right)\right], \quad t = 0, \ldots, T-1, \tag{13}$$

where $\mathrm{clip}_{[0,1]}$ constrains the updated pixels to the valid image range. Increasing C_2 discourages excessive alignment between the early residual updates, thereby increasing directional variation in

Table 1: Clean and adversarial (Adv.) performance on generative tasks via LLaVA-1.5-7B.

Methods	COCO		Flickr30k		TextVQA		VQAv2		Average	
	Clean	Adv.	Clean	Adv.	Clean	Adv.	Clean	Adv.	Clean	Adv.
CLIP	115.5	2.7	77.5	0.8	37.1	0.2	74.6	0.1	76.2	1.0
TeCoA	88.3	35.3	48.6	19.5	20.7	9.3	63.2	31.7	55.2	23.6
FARE	102.4	40.9	61.6	22.8	54.2	24.6	27.6	10.9	65.0	26.3
AdPO	109.2	47.8	65.7	24.1	30.0	10.1	70.0	32.4	86.7	28.6
ET3	111.3	61.1	76.1	37.1	34.9	17.1	73.1	40.0	73.9	38.8
CURB (Ours)	**115.4**	**116.0**	**77.2**	**73.9**	**36.9**	**28.0**	**74.5**	**65.3**	**76.0**	**70.8**

the early representation trajectory. The bounded iterative updates, following finite-step curvature increase proposed in Equation 10, preserve the visual content of the original test input, and the resulting image $x^{(T)}$ is used for downstream inference.

3.2 Selective Test-Time Inference.

Applying rectification indiscriminately may unnecessarily modify clean inputs. We therefore use the same early curvature signal to selectively activate CURB. Given a fixed threshold τ, the defended input is obtained as:

$$\hat{x} = \begin{cases} x_0, & C_2(x_0) \geq \tau, \\ x^{(T)}, & C_2(x_0) < \tau, \end{cases} \tag{14}$$

and the final prediction $\hat{y} = f_{\text{LVLM}}(\hat{x}, q)$ is then produced by the original frozen model.

4 Experiments

We evaluate CURB from four perspectives. In Section 4.1, we assess its robustness on generative tasks and analyze the source of its gains. In Section 4.2, we evaluate its effectiveness on zero-shot classification tasks. In Section 4.3, we examine its robustness against adaptive attacks. In Section 4.4, we ablate the number of optimization steps in Equation 13, the gating threshold τ in Equation 14, and the curvature measurement point in Equation 12. All experiments are conducted on an NVIDIA RTX 4090 GPUs, and implementation details are provided in Appendix C.1.

4.1 Main Results on Generative tasks

Attack Setup and Baselines Following prior work (Schlarmann et al., 2024), we evaluate LVLM's robustness under 100-step APGD with an ℓ_∞ perturbation budget of $4/255$, and evaluations on clean data use the full datasets, while adversarial results are computed over 500 perturbed inputs. We first evaluate CURB with LLaVA-1.5-7B against four training-time methods TeCo (Mao et al., 2023), FARE (Schlarmann et al., 2024), and AdPO (Liu et al., 2026a), and a test-time defense ET3 (Mirza et al., 2026). Details are provided in Appendix C.2.

Datasets and Metrics We consider image captioning (COCO (Lin et al., 2014), Flickr30k (Plummer et al., 2015)), and visual question answering (TextVQA (Singh et al., 2019), VQAv2 (Goyal et al., 2017)) as the generative benchmarks. We report the CIDEr score (Vedantam et al., 2015) for captioning and VQA accuracy (Antol et al., 2015) for visual-question answering. To further investigate the source of CURB's performance gains, we further report the the final clean–adversarial representation squared Frobenius discrepancy $E_L(x_a, x_c)$ on LLaVA-1.5-7B and InternVL3.5-8B.

Results As shown in Table 1, CURB achieves the best adversarial robustness of 70.8, significantly outperforming advanced defense methods. Meanwhile, CURB also maintain the clean performance most across all the benchmarks. ET3 optimizes the input in a closed-set ImageNet-21K proxy outputs space and subsequently transfers the transformed image to the downstream LVLMs. Therefore, the potential mismatch between the proxy semantic space and the query-conditioned visual representations required by generative tasks is overlooked, which limits its effectiveness.

Table 2: Clean and adversarial performance, and the final clean–adversarial representation squared Frobenius discrepancy $E_L(x_a, x_c)$ on generative tasks via LLaVA-1.5-7B and InternVL3.5-8B.

Methods		COCO			Flickr30k			TextVQA			VQAv2			Average		
		Clean	Adv.	E_L	Clean	Adv.	E_L	Clean	Adv.	E_L	Clean	Adv.	E_L	Clean	Adv.	E_L
LLaVA	Baseline	115.5	2.7	0.61	77.5	0.8	0.60	37.1	0.2	0.22	74.6	0.1	0.47	76.2	0.9	0.48
	+CURB	115.4	**116.0**	0.39	77.2	**73.9**	0.38	36.9	**28.0**	0.17	74.5	**65.3**	0.32	76.0	**70.8**	0.32
InternVL	Baseline	111.0	7.4	0.52	88.0	4.4	0.49	76.8	0.0	0.41	75.5	1.5	0.42	88.6	3.3	0.46
	+CURB	106.7	**81.0**	0.34	84.1	**56.0**	0.35	75.8	**41.3**	0.31	74.6	**56.9**	0.30	85.3	**58.8**	0.33

Table 3: Clean and adversarial performance on zero-shot classification tasks via CLIP-ViT-L/14.

Methods	Caltech101		Pets		Cars		Flower102		Aircraft		DTD		EuroSAT		UCF101		Average	
	Clean	Adv.	Clean	Adv.	Clean	Adv.	Clean	Adv.	Clean	Adv.	Clean	Adv.	Clean	Adv.	Clean	Adv.	Clean	Adv.
CLIP	95.2	0.0	93.1	0.0	76.8	0.0	76.2	0.0	30.0	0.0	52.4	0.0	55.1	0.0	73.7	0.0	69.1	0.0
Lighweight Single-View Inference																		
TTC	88.7	7.7	92.2	7.6	67.8	2.2	76.5	7.5	31.7	0.5	49.7	6.2	64.1	0.2	75.0	2.2	68.2	4.3
DOC	94.6	5.3	90.4	3.1	75.1	0.7	76.6	3.5	28.3	0.1	51.8	6.9	50.5	0.00	72.9	0.6	67.5	2.5
CURB	**95.3**	**45.1**	**93.1**	**28.7**	**73.9**	**34.3**	**75.4**	**35.3**	**29.6**	**11.5**	**48.9**	**28.7**	**54.5**	**17.8**	**73.2**	**17.4**	**68.0**	**22.9**
Multi-View Inference ("+" denotes the variant with multi-view ensemble)																		
MTA	95.8	83.1	93.7	64.9	78.4	36.6	76.1	44.2	32.7	8.0	53.4	27.2	47.8	7.5	74.7	47.5	69.1	39.9
R-TPT	95.7	88.2	93.7	72.9	77.2	49.1	76.2	55.6	31.7	17.2	54.0	38.0	44.3	20.4	74.3	55.6	68.4	49.6
TTP	95.1	88.6	93.1	76.3	76.8	51.1	76.1	58.7	29.2	17.7	52.3	41.3	55.0	21.6	73.6	57.4	68.9	51.6
A-TPT	95.8	90.1	94.1	80.3	78.8	52.5	76.5	61.2	29.6	18.4	53.6	39.5	46.2	25.9	73.1	61.4	68.5	53.7
CURB$^+$	**95.6**	**93.6**	**93.5**	**82.1**	**77.4**	**58.9**	**75.8**	**61.9**	**29.6**	**22.0**	**52.8**	**42.1**	**48.4**	**36.8**	**73.2**	**64.2**	**68.3**	**57.7**

Table 2 shows that CURB consistently improves the adversarial robustness of the base models. This improvement is accompanied by a substantial reduction in the final visual feature shift, supporting our hypothesis that enhancing early residual differentiation suppresses adversarial deviation in deeper layers. Specifically, $E_L(x_a, x_c)$ decreases by 33.3% for LLaVA and 28.2% for InternVL, suggesting that CURB's gains arise from a generalizable representation-level mechanism rather than model-specific engineering.

4.2 Main Results on Zero-shot Classification tasks

Setup Following previous work (Sheng et al., 2025), we evaluate zero-shot classification with CLIP-ViT-L/14 on eight fine-grained benchmarks (Caltech101 (Fei-Fei et al., 2004), Pets (Parkhi et al., 2012), Cars (Krause et al., 2013), Flower102 (Nilsback & Zisserman, 2008), Aircraft (Maji et al., 2013), DTD (Cimpoi et al., 2014), EuroSAT (Helber et al., 2019), UCF101 (Soomro et al., 2012)) under 100-step PGD (Madry et al., 2018) attack with an ℓ_∞ perturbation budget of $4/255$. We compare CURB with six test-time defense methods, including four multi-view methods, MTA (Zanella & Ben Ayed, 2024), R-TPT (Sheng et al., 2025), TTP (Li et al., 2026), and A-TPT (Hai et al., 2026), as well as two single-view methods, TTC (Xing et al., 2025) and DOC (Jiang et al., 2026). For a fair comparison, CURB is applied to the original input under the single-view setting, while under the multi-view setting, it only uses the same augmentations of AugMix (Hendrycks et al., 2020) followed by ensemble prediction. Details are provided in Appendix C.3.

Results As shown in Table 3, CURB consistently achieves the best robustness across different settings. Under multi-view inference, CURB achieves an average adversarial robustness of 57.7%, outperforming the SOTA method A-TPT by 4.0% on average. More importantly, a set of defense focus on lightweight single-view transformation and inference. But under a strong 100-step adversarial attack, the lightweight defenses TTC and DOC almost completely lose their effectiveness. In contrast, CURB retains relatively substantial robustness, achieves the adversarial robustness of 22.9%. These results demonstrate that CURB provides a lightweight yet effective intervention point for test-time adversarial defense.

4.3 Main results under adaptive attacks

Setup To assess the robustness of CURB against stronger adaptive attacks, we combine Backward Pass Differentiable Approximation (BPDA) (Athalye et al., 2018) with AutoAttack (Croce & Hein, 2020) as a stronger adversarial attack. For comparison with the test-time defense ET3 (Mirza et al.,

Table 4: Adversarial robustness under standard (Adv.) and adaptive (Adv.*) attacks on generative tasks via LLaVA-1.5-7B with different vision encoder.

Methods		COCO		Flickr30k		TextVQA		VQAv2		Average	
		Adv.	Adv.*	Adv.	Adv.*	Adv.	Adv.*	Adv.	Adv.*	Adv.	Adv.*
CLIP	baseline	2.8		0.9		0.0		0.0		0.9	
	+ET3	19.2	3.5	16.0	5.0	12.0	2.0	19.6	7.4	16.7	4.5
	+CURB	**34.2**	**31.3**	**31.3**	**30.6**	**19.4**	**17.1**	**26.5**	**23.7**	**27.9**	**25.7**
TeCoA	baseline	25.4		22.4		7.8		23.0		19.7	
	+ET3	45.0	41.4	31.6	29.6	9.8	9.8	37.4	33.4	30.9	28.6
	+CURB	**62.5**	**58.3**	**57.6**	**55.1**	**20.1**	**19.2**	**48.7**	**46.6**	**47.2**	**44.8**
FARE	baseline	28.1		28.0		13.8		26.6		24.1	
	+ET3	36.2	34.3	39.5	39.9	19.8	17.8	35.8	31.8	32.8	31.0
	+CURB	**74.8**	**72.1**	**69.5**	**68.7**	**24.4**	**22.1**	**61.0**	**58.9**	**57.4**	**55.5**

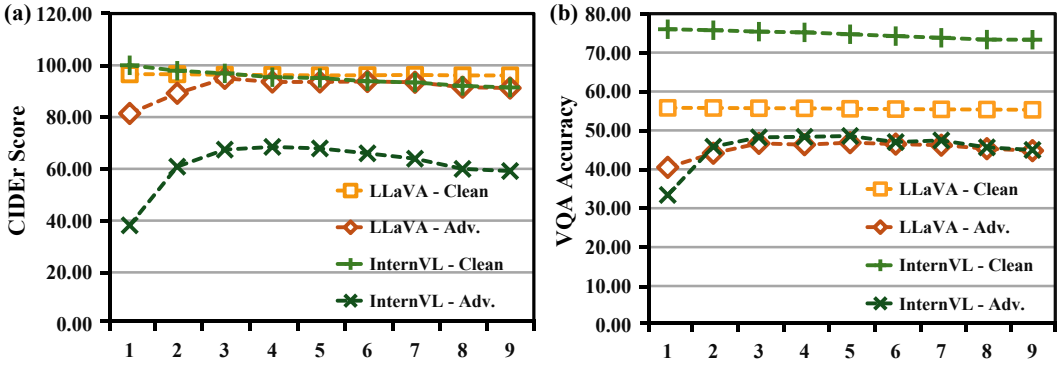

Figure 3: Effect of varying the number of optimization steps from 1 to 9 on clean and adversarial performance across generative tasks via LLaVA-1.5-7B and InternVL3.5-8B. (a) Image captioning performance measured by CIDEr score. (b) VQA performance measured by VQA accuracy.

2026), which relies on robustly trained vision encoders, we evaluate LLaVA-1.5-7B on generative tasks using three pre-trained vision encoders: standard CLIP, robustly trained TeCoA-CLIP, and FARE-CLIP. Details are provided in Appendix C.5.

Results As shown in Table 4, CURB substantially improves the robustness of all three vision encoders under both standard and adaptive attacks, with only a modest performance drop under the adaptive setting. These results indicate that CURB's robustness gains do not primarily rely on the attacker being unaware of the defense. Notably, even with the original CLIP encoder, CURB improves the average adversarial robustness from 0.9 to 25.7 under adaptive attacks, demonstrating that its effectiveness does not rely on a robustly trained vision encoder.

4.4 Ablation Studies

Cross-Task Ablation of Optimization Steps We evaluate CURB with varying numbers of optimization steps on VQA and image captioning tasks. As shown in Figure 3, adversarial robustness improves rapidly within the first few steps and then stabilizes. In contrast, clean performance remains largely stable, with only minor degradation because a small number of clean inputs are misidentified as adversarial by the gating threshold τ. Notably, the optimal number of steps is consistent across tasks within each architecture: LLaVA-1.5-7B performs best at 3 steps on both VQA and captioning, whereas InternVL3.5-8B performs best at 4 steps. This task-consistent behavior indicates that CURB generalizes reliably across different generative tasks. Detailed results on each dataset are provided in Appendix C.5.

Ablation of Gating Threshold and Optimization Direction We further ablate the gating mechanism and the curvature optimization direction, as shown in Table 5. Removing the gate and increasing early curvature for all samples (↑) improves adversarial performance but consistently degrades

Table 5: Ablation of selective gating and optimization direction across generative benchmarks via LLaVA-1.5-7B and InternVL3.5-8B. CURB ↑ and CURB ↓ remove the gate and optimize all samples by increasing and decreasing early curvature, respectively.

Methods		COCO		Flickr30k		TextVQA		VQAv2		Average	
		Clean	Adv.	Clean	Adv.	Clean	Adv.	Clean	Adv.	Clean	Adv.
LLaVA	+CURB ↑	110.7	119.2	74.3	75.8	34.3	31.1	70.7	68.4	72.5	73.6
	+CURB ↓	3.2	1.3	1.7	0.1	0.0	0.0	0.1	0.0	1.3	0.4
	+CURB	115.4	116.0	77.2	73.9	36.9	28.0	74.5	65.3	76.0	70.8
InternVL	+CURB ↑	100.1	85.7	79.5	59.4	71.0	44.3	78.2	58.1	82.2	61.9
	+CURB ↓	11.2	2.3	6.3	0.3	0.7	0.0	1.1	0.0	4.8	0.7
	+CURB	106.7	81.0	84.1	56.0	75.8	41.3	74.6	56.9	85.3	58.8

Table 6: Layer-wise ablation of the curvature measurement position on generative tasks via LLaVA-1.5-7B. C_l denotes the residual discrete curvature measured from layers $l-1$ to $l+1$.

Methods		COCO		Flickr30k		TextVQA		VQAv2		Average	
		Clean	Adv.	Clean	Adv.	Clean	Adv.	Clean	Adv.	Clean	Adv.
LLaVA	+baseline	115.5	2.7	77.5	0.8	37.1	0.2	74.6	0.1	76.2	1.0
	$+C_2$	115.4	116.0	77.2	73.9	36.9	28.0	74.5	65.3	76.0	70.8
	$+C_5$	115.5	108.5	77.4	68.5	36.5	23.5	74.5	60.9	76.0	65.4
	$+C_8$	114.9	94.7	76.9	58.8	36.2	16.6	74.8	53.2	75.7	55.8

clean performance, indicating that selective activation is necessary to avoid unnecessary rectification of naturally well-behaved inputs. In contrast, reversing the optimization direction to decrease early curvature (↓) causes a dramatic collapse in both clean and adversarial performance. These results confirm that the robustness gain is specifically associated with increasing early residual curvature, while the gate primarily preserves clean performance by suppressing unnecessary updates.

Layer-Wise Ablation of the Curvature Measurement We focus on the earliest curvature measurement point along the residual trajectory. To validate this choice, we conduct a layer-wise ablation of the curvature measurement position. As shown in Table 6, the robustness gain of CURB decreases as curvature is measured at deeper layers, while clean performance remain stable. These results are consistent with our previous motivation: the earliest measurement captures directional abnormalities before adversarial deviations are further amplified by downstream layers. In contrast, deeper-layer curvature is measured after additional nonlinear MLP transformations, where adversarial discrepancies have already accumulated.

5 Conclusions and Limitations

In this work, we studied test-time adversarial defense through the internal propagation dynamics of LVLMs. We found that adversarial perturbations suppress early residual directional differentiation, while the resulting discrepancy is predominantly amplified through deep MLP branches. Based on this observation, we proposed CURB, a training-free defense that selectively increases early residual curvature to counteract the amplification of adversarially induced feature discrepancies. Experiments across generative and zero-shot classification tasks demonstrated consistent robustness gains, supporting the effectiveness and cross-task generality of CURB. We suggest that early residual geometry provides a simple yet effective intervention point for test-time adversarial robustness. Our analysis focused on ViT-based visual encoders with global self-attention in every layer, where residual propagation follows a relatively homogeneous layer-wise structure. Whether the same pattern persists in architectures with heterogeneous attention topologies, such as windowed, hierarchical, or hybrid visual encoders, remains to be analyzed. Extending the analysis to these architectures is therefore an important direction for future work.

AI USE STATEMENT

In this work, we used generative AI tools for refining paper writing, assisting in the analysis of experimental results, and implementing parts of the experimental code. We have not used generative AI tools for generating synthetic data, formulating mathematical claims or proofs, or proposing the core hypotheses of this work, and the remaining required-disclosure tasks are not applicable. Additionally, we used generative AI tools for editing the paper to improve readability and identifying relevant literature. We have reviewed all AI-assisted work: every methodological suggestion was evaluated by the authors against their own experiments before adoption, all experiments were run and verified by the authors, every AI-suggested reference was checked against the original publication, and AI-generated code was tested for correctness. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We are committed to conducting reliable research, and experiments in this work are fully reproducible. We will release the complete codes and original results at `anonymous.4open.science/r/CURB-1`.

REFERENCES

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual Question Answering. In *Proc. IEEE Int. Conf. Comp. Vis.*, pp. 2425–2433, 2015.

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples. In *Proc. Int. Conf. Mach. Learn.*, pp. 274–283, 2018.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing Textures In The Wild. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 3606–3613, 2014.

Francesco Croce and Matthias Hein. Reliable Evaluation of Adversarial Robustness with an Ensemble of Diverse Parameter-Free Attacks. In *Proc. Int. Conf. Mach. Learn.*, pp. 2206–2216, 2020.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 248–255, 2009.

Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn. Workshops*, pp. 178, 2004.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining And Harnessing Adversarial Examples. In *Proc. Int. Conf. Learn. Representations*, 2015.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 6904–6913, 2017.

Jia-Wei Hai, Yijun Wang, and Xiu-Shen Wei. Towards Fine-Grained Robustness: Attention-Guided Test-Time Prompt Tuning for Vision-Language Models. In *Proc. Int. Conf. Mach. Learn.*, 2026.

Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A Novel Dataset And Deep Learning Benchmark For Land Use And Land Cover Classification. *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, 12:2217–2226, 2019.

Dan Hendrycks, Norman Mu, Ekin D. Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. AugMix: A Simple Data Processing Method To Improve Robustness And Uncertainty. In *Proc. Int. Conf. Learn. Representations*, 2020.

Chengze Jiang, Minjing Dong, Xinli Shi, and Jie Gui. Diversifying Counterattacks: Orthogonal Exploration for Robust CLIP Inference. In *Proc. Conf. AAAI*, pp. 5359–5368, 2026.

Sunoh Kim and Daeho Um. When CLIP Sees More, It Fights Back Harder: Multi-View Guided Adaptive Counterattacks for Test-Time Adversarial Robustness. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 15689–15699, 2026a.

Sunoh Kim and Daeho Um. SS-TPT: Stability and Suitability-Guided Test-Time Prompt Tuning for Adversarially Robust Vision-Language Models. In *Proc. Int. Conf. Mach. Learn.*, 2026b.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3D Object Representations For Fine-Grained Categorization. In *Proc. IEEE Int. Conf. Comp. Vis. Workshops*, pp. 554–561, 2013.

Zhiwei Li, Yitian Pang, Weining Wang, Zhenan Sun, and Qi Li. TTP: Test-Time Padding for Adversarial Detection and Robust Adaptation on Vision-Language Models. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 1513–1522, 2026.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *Proc. Eur. Conf. Comp. Vis.*, pp. 740–755, 2014.

Chaohu Liu, Tianyi Gui, Yu Liu, and Linli Xu. AdPO: Enhancing the Adversarial Robustness of Large Vision-Language Models with Preference Optimization. In *Proc. Int. Conf. Learn. Representations*, 2026a.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved Baselines with Visual Instruction Tuning. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 26296–26306, 2024.

Jiaxiang Liu, Jiawei Du, Xiao Liu, Shangyang Li, Songchen Ma, Changshuo Wang, Prayag Tiwari, and Mingkun Xu. Self-Calibrated Consistency can Fight Back for Adversarial Robustness in Vision-Language Models. In *Proc. Int. Conf. Mach. Learn.*, 2026b.

Liangsheng Liu, Si Chen, Jiamin Wu, Weiwei Feng, Zhixin Cheng, Xiaotian Yin, Wenfei Yang, and Tianzhu Zhang. Adversarial Attacks Already Tell the Answer: Directional Bias-Guided Test-Time Defense for Vision-Language Models. In *Proc. Int. Conf. Learn. Representations*, 2026c.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards Deep Learning Models Resistant To Adversarial Attacks. In *Proc. Int. Conf. Learn. Representations*, 2018.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-Grained Visual Classification Of Aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

Chengzhi Mao, Scott Geng, Junfeng Yang, Xin Wang, and Carl Vondrick. Understanding Zero-Shot Adversarial Robustness For Large-Scale Models. In *Proc. Int. Conf. Learn. Representations*, 2023.

Mujtaba Hussain Mirza, Antonio D'Orazio, Odelia Melamed, and Iacopo Masi. A Provable Energy-Guided Test-Time Defense Boosting Adversarial Robustness of Large Vision-Language Models. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 8598–8609, 2026.

Maria-Elena Nilsback and Andrew Zisserman. Automated Flower Classification Over A Large Number Of Classes. In *Proc. 6th Indian Conf. Comp. Vis., Graphics & Image Process.*, pp. 722–729, 2008.

Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats And Dogs. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 3498–3505, 2012.

Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k Entities: Collecting Region-to-Phrase Correspondences for Richer Image-to-Sentence Models. In *Proc. IEEE Int. Conf. Comp. Vis.*, pp. 2641–2649, 2015.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In *Proc. Int. Conf. Mach. Learn.*, pp. 8748–8763, 2021.

Christian Schlarmann and Matthias Hein. On the Adversarial Robustness of Multi-Modal Foundation Models. In *Proc. IEEE/CVF Int. Conf. Comp. Vis. Workshops*, pp. 3677–3685, 2023.

Christian Schlarmann, Naman Deep Singh, Francesco Croce, and Matthias Hein. Robust CLIP: Unsupervised Adversarial Fine-Tuning Of Vision Embeddings For Robust Large Vision-Language Models. In *Proc. Int. Conf. Mach. Learn.*, pp. 43685–43704, 2024.

Lijun Sheng, Jian Liang, Zilei Wang, and Ran He. R-TPT: Improving Adversarial Robustness of Vision-Language Models through Test-Time Prompt Tuning. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 29958–29967, 2025.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards VQA Models that Can Read. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 8317–8326, 2019.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A Dataset Of 101 Human Action Classes From Videos In The Wild. Technical Report CRCV-TR-12-01, Center for Research in Computer Vision, University of Central Florida, November 2012.

Baoshun Tong, Hanjiang Lai, Yan Pan, and Jian Yin. On the Zero-Shot Adversarial Robustness of Vision-Language Models: A Truly Zero-Shot and Training-Free Approach. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 19921–19930, 2025.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. CIDEr: Consensus-Based Image Description Evaluation. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 4566–4575, 2015.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. InternVL3.5: Advancing Open-Source Multimodal Models in Versatility, Reasoning, and Efficiency. *arXiv preprint arXiv:2508.18265*, 2025a.

Xin Wang, Kai Chen, Jiaming Zhang, Jingjing Chen, and Xingjun Ma. TAPT: Test-Time Adversarial Prompt Tuning for Robust Inference in Vision-Language Models. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 19910–19920, 2025b.

Songlong Xing, Zhengyu Zhao, and Nicu Sebe. CLIP is Strong Enough to Fight Back: Test-Time Counterattacks towards Zero-Shot Adversarial Robustness of CLIP. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 15172–15182, 2025.

Maolin Xiong, Yuling Chen, Haiwei Sang, Hui Dou, and Long Chen. Enhancing The Transferability Of Adversarial Attacks Through Dual-Frequency Domain Transforms And High-Frequency Gradient Guided. *Chin. J. Electron.*, 34(5):1438–1448, 2025.

Hong-Tao Yu, Yuxin Peng, Serge Belongie, and Xiu-Shen Wei. Benchmarking Large Vision-Language Models On Fine-Grained Image Tasks: A Comprehensive Evaluation. In *Proc. Int. Conf. Learn. Representations*, 2026.

Maxime Zanella and Ismail Ben Ayed. On the Test-Time Zero-Shot Generalization of Vision-Language Models: Do We Really Need Prompt Learning? In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 23783–23793, 2024.

Mingkun Zhang, Keping Bi, Wei Chen, Jiafeng Guo, and Xueqi Cheng. CLIPure: Purification in latent space via CLIP for adversarially robust zero-shot classification. In *Proc. Int. Conf. Learn. Representations*, 2025.

Xingyu Zhu, Beier Zhu, Shuo Wang, Kesen Zhao, and Hanwang Zhang. Enhancing CLIP Robustness via Cross-Modality Alignment. In *Advances in Neural Inf. Process. Syst.*, 2025.

APPENDIX CONTENTS

A Related Works **14**

A.1 Adversarial Attacks 14

A.2 Test-time Adversarial Defenses 14

B Appendix for Section 2 **14**

B.1 Complexity of Multi-View Inference 14

B.2 Proof for Proposition 1: Exact Residual-Branch Decomposition 15

B.3 Proof for Proposition 2: Geometry of Early Residual Curvature 16

B.4 Proof for Proposition 3: Finite-Step Curvature Increase. 17

C Appendix for Section 4 **18**

C.1 General Settings 18

C.2 Details of Evaluation On Generative Tasks 18

C.3 Details of Evaluation On Zero-Shot Classification 18

C.4 Details Under Adaptive Attacks 19

C.5 Analysis of Inference Efficiency 19

A Related Works

A.1 Adversarial Attacks

Adversarial attacks on LVLMs can induce significant degradation to model Inference. A variety of attack methods have been proposed (Madry et al., 2018; Croce & Hein, 2020; Goodfellow et al., 2015) to generate slight perturbations. Classical attacks like FGSM(Goodfellow et al., 2015) directly perturb the direction of loss gradients. Recently, standard measures of robustness like PGD (Madry et al., 2018) project gradient descent onto a perturbed L_p ball with an ε_a-bounded constraint. Among them, Schlarmann et al. (Schlarmann et al., 2024) strengthen the standard APGD attacks (Croce & Hein, 2020) on the pipeline of their early work (Schlarmann & Hein, 2023).

A.2 Test-time Adversarial Defenses

Recent test-time defenses improve VLMs robustness through different forms of test-time adaptation. A-TPT (Hai et al., 2026), R-TPT (Sheng et al., 2025), SS-TPT (Kim & Um, 2026b), TTP (Li et al., 2026), and TAPT (Wang et al., 2025b) optimize prediction entropy over augmented views, heavily relying on closed-set class spaces and multiple views. COLA (Zhu et al., 2025), SCC (Liu et al., 2026b), and DBD (Liu et al., 2026c) rectify visual representations using class-conditioned semantics and transformed views. In contrast, AOM (Tong et al., 2025), MAC (Kim & Um, 2026a), and DOC (Jiang et al., 2026) avoid class-conditioned objectives, but estimate rectification directions in corrupted visual representation, performing limited robustness under stronger attacks (Madry et al., 2018; Croce & Hein, 2020). Although ET3 (Mirza et al., 2026) extends test-time defense to generative LVLMs, it still relies on a classification-oriented proxy space for input optimization. Overall, existing test-time defenses remain tied to either predefined outputs spaces or multi-view processing, leaving an efficient defense for open-ended LVLMs underexplored.

B Appendix for Section 2

B.1 Complexity of Multi-View Inference

For CLIP-based classification, each augmented view requires one visual-encoder forward pass. The dominant costs of self-attention and feed-forward transformations are $L_v N_v^2 d_v$ and $L_v N_v d_v^2$, respectively. Since the C class-text features can be precomputed, image–text similarity computation adds only Cd_e. Therefore, the time complexity of K-view classification is:

$$\mathcal{T}_{\mathrm{CLIP}}^{(K)} = \mathcal{O}\big(K\left[L_v\left(N_v^2 d_v + N_v d_v^2\right) + Cd_e\right]\big). \tag{15}$$

For generative LVLMs, each view additionally requires language-model prefilling and autoregressive decoding. Processing the multimodal context costs:

$$\mathcal{T}_{\mathrm{prefill}} = \mathcal{O}\big(L_l\left(N_0^2 d_l + N_0 d_l^2\right)\big). \tag{16}$$

Under standard key–value caching, the token generated at step t attends to $N_0 + t - 1$ preceding tokens and incurs additional linear-projection and feed-forward costs of $\mathcal{O}(d_l^2)$. Since $\sum_{t=1}^{T_l}(N_0 + t - 1) = T_l N_0 + \mathcal{O}(T_l^2)$, the total decoding complexity is:

$$\mathcal{T}_{\mathrm{decode}} = \mathcal{O}\big(L_l\left(T_l N_0 d_l + T_l^2 d_l + T_l d_l^2\right)\big). \tag{17}$$

Combining visual encoding, context prefilling, and autoregressive decoding, the time complexity of K-view LVLM inference is:

$$\mathcal{T}_{\mathrm{LVLM}}^{(K)} = \mathcal{O}\Bigg(K \Big[L_v \left(N_v^2 d_v + N_v d_v^2 \right) + L_l \left(N_0^2 d_l + N_0 d_l^2 \right) + L_l \left(T_l N_0 d_l + T_l^2 d_l + T_l d_l^2 \right) \Big] \Bigg). \tag{18}$$

Unlike CLIP classification, different augmented views produce different visual tokens. Their multimodal prefill states and key–value caches therefore cannot generally be shared, requiring the complete generation process to be repeated for every view. Architecture-dependent terms, such as visual projection and vocabulary-logit computation, are omitted because they do not alter the linear dependence on K. Without key–value caching, the autoregressive cost would be higher.

B.2 Proof for Proposition 1: Exact Residual-Branch Decomposition

Definition 1. For a paired clean input x_c and adversarial input x_a, define the clean–adversarial representation difference at layer l as

$$\Delta \boldsymbol{H}_l(x_a, x_c) = \boldsymbol{H}_l(x_a) - \boldsymbol{H}_l(x_c). \tag{19}$$

For the l-th pre-norm Transformer block, let $\boldsymbol{U}_l^{\mathrm{Attn}}(x)$ and $\boldsymbol{U}_l^{\mathrm{MLP}}(x)$ denote the Attention and MLP residual updates, respectively:

$$\boldsymbol{U}_l^{\mathrm{Attn}}(x) = \mathrm{Attn}_l \left(\mathrm{LN}_l^{\mathrm{Attn}} \left(\boldsymbol{H}_l(x) \right) \right). \tag{20}$$

$$\boldsymbol{U}_l^{\mathrm{MLP}}(x) = \mathrm{MLP}_l \left(\mathrm{LN}_l^{\mathrm{MLP}} \left(\boldsymbol{H}_l(x) + \boldsymbol{U}_l^{\mathrm{Attn}}(x) \right) \right). \tag{21}$$

where $\mathrm{LN}_l^{\mathrm{Attn}}$ and $\mathrm{LN}_l^{\mathrm{MLP}}$ denote the layer-normalization operations preceding the Attention and MLP sublayers, respectively. Accordingly, $\boldsymbol{H}_{l+1}(x) = \boldsymbol{H}_l(x) + \boldsymbol{U}_l^{\mathrm{Attn}}(x) + \boldsymbol{U}_l^{\mathrm{MLP}}(x)$.

We define the corresponding clean–adversarial branch differences as

$$\Delta \boldsymbol{U}_l^{\mathrm{Attn}}(x_a, x_c) = \boldsymbol{U}_l^{\mathrm{Attn}}(x_a) - \boldsymbol{U}_l^{\mathrm{Attn}}(x_c). \tag{22}$$

$$\Delta \boldsymbol{U}_l^{\mathrm{MLP}}(x_a, x_c) = \boldsymbol{U}_l^{\mathrm{MLP}}(x_a) - \boldsymbol{U}_l^{\mathrm{MLP}}(x_c). \tag{23}$$

where $\Delta \boldsymbol{U}_l^{\mathrm{Attn}}(x_a, x_c)$ and $\Delta \boldsymbol{U}_l^{\mathrm{MLP}}(x_a, x_c)$ measure the clean–adversarial differences introduced by the Attention and MLP residual branches, respectively.

Proof. For the intermediate clean–adversarial representation difference after the Attention residual update, define

$$\Delta \boldsymbol{H}_l^{\mathrm{Attn}}(x_a, x_c) = \Delta \boldsymbol{H}_l(x_a, x_c) + \Delta \boldsymbol{U}_l^{\mathrm{Attn}}(x_a, x_c). \tag{24}$$

After the subsequent MLP residual update,

$$\Delta \boldsymbol{H}_{l+1}(x_a, x_c) = \Delta \boldsymbol{H}_l^{\mathrm{Attn}}(x_a, x_c) + \Delta \boldsymbol{U}_l^{\mathrm{MLP}}(x_a, x_c). \tag{25}$$

Using the identity $\|\boldsymbol{X} + \boldsymbol{Y}\|_F^2 = \|\boldsymbol{X}\|_F^2 + 2\langle \boldsymbol{X}, \boldsymbol{Y} \rangle_F + \|\boldsymbol{Y}\|_F^2$, for matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$ of the same shape, where $\langle \cdot, \cdot \rangle_F$ denotes the Frobenius inner product, the discrepancy change induced by the Attention and MLP branch is

$$\left\| \Delta \boldsymbol{H}_l^{\mathrm{Attn}}(x_a, x_c) \right\|_F^2 - \left\| \Delta \boldsymbol{H}_l(x_a, x_c) \right\|_F^2 = 2 \left\langle \Delta \boldsymbol{H}_l(x_a, x_c), \Delta \boldsymbol{U}_l^{\mathrm{Attn}}(x_a, x_c) \right\rangle_F + \left\| \Delta \boldsymbol{U}_l^{\mathrm{Attn}}(x_a, x_c) \right\|_F^2, \tag{26}$$

$$\left\| \Delta \boldsymbol{H}_{l+1}(x_a, x_c) \right\|_F^2 - \left\| \Delta \boldsymbol{H}_l^{\mathrm{Attn}}(x_a, x_c) \right\|_F^2 = 2 \left\langle \Delta \boldsymbol{H}_l^{\mathrm{Attn}}(x_a, x_c), \Delta \boldsymbol{U}_l^{\mathrm{MLP}}(x_a, x_c) \right\rangle_F + \left\| \Delta \boldsymbol{U}_l^{\mathrm{MLP}}(x_a, x_c) \right\|_F^2. \tag{27}$$

Accordingly, we can therefore define the contribution of the Attention and MLP as

$$G_l^{\mathrm{Attn}}(x_a, x_c) = 2\left\langle \Delta \boldsymbol{H}_l(x_a, x_c), \Delta \boldsymbol{U}_l^{\mathrm{Attn}}(x_a, x_c)\right\rangle_F + \left\|\Delta \boldsymbol{U}_l^{\mathrm{Attn}}(x_a, x_c)\right\|_F^2. \tag{28}$$

$$G_l^{\mathrm{MLP}}(x_a, x_c) = 2\left\langle \Delta \boldsymbol{H}_l^{\mathrm{Attn}}(x_a, x_c), \Delta \boldsymbol{U}_l^{\mathrm{MLP}}(x_a, x_c)\right\rangle_F + \left\|\Delta \boldsymbol{U}_l^{\mathrm{MLP}}(x_a, x_c)\right\|_F^2. \tag{29}$$

Since $E_l(x_a, x_c) = \|\Delta \boldsymbol{H}_l(x_a, x_c)\|_F^2$, summing the Attention and MLP contributions within the l-th block yields

$$E_{l+1}(x_a, x_c) - E_l(x_a, x_c) = G_l^{\mathrm{Attn}}(x_a, x_c) + G_l^{\mathrm{MLP}}(x_a, x_c). \tag{30}$$

Summing this identity from $l = s$ to $L - 1$ gives

$$\sum_{l=s}^{L-1} \left(E_{l+1}(x_a, x_c) - E_l(x_a, x_c)\right) = \sum_{l=s}^{L-1} G_l^{\mathrm{Attn}}(x_a, x_c) + \sum_{l=s}^{L-1} G_l^{\mathrm{MLP}}(x_a, x_c). \tag{31}$$

The left-hand side telescopes, resulting in

$$E_L(x_a, x_c) - E_s(x_a, x_c) = \sum_{l=s}^{L-1} G_l^{\mathrm{Attn}}(x_a, x_c) + \sum_{l=s}^{L-1} G_l^{\mathrm{MLP}}(x_a, x_c). \tag{32}$$

□

B.3 Proof for Proposition 2: Geometry of Early Residual Curvature

Definition 2. Define the normalized directions of $\boldsymbol{U}_1(x)$ and $\boldsymbol{U}_2(x)$ as

$$\widehat{\boldsymbol{U}}_1(x) = \frac{\boldsymbol{U}_1(x)}{\|\boldsymbol{U}_1(x)\|_F}, \qquad \widehat{\boldsymbol{U}}_2(x) = \frac{\boldsymbol{U}_2(x)}{\|\boldsymbol{U}_2(x)\|_F}. \tag{33}$$

The residual curvature defined in Equation 7 can then be written as

$$C_2(x) = 1 - \left\langle \widehat{\boldsymbol{U}}_1(x), \widehat{\boldsymbol{U}}_2(x)\right\rangle_F. \tag{34}$$

Since $\|\widehat{\boldsymbol{U}}_1(x)\|_F = \|\widehat{\boldsymbol{U}}_2(x)\|_F = 1$, we have $C_2(x) \in [0, 2]$.

We decompose the two normalized residual directions into a common component $\boldsymbol{M}(x)$ and a differential component $\boldsymbol{R}(x)$:

$$\boldsymbol{M}(x) = \frac{\widehat{\boldsymbol{U}}_1(x) + \widehat{\boldsymbol{U}}_2(x)}{2}, \qquad \boldsymbol{R}(x) = \frac{\widehat{\boldsymbol{U}}_1(x) - \widehat{\boldsymbol{U}}_2(x)}{2}. \tag{35}$$

Here, $\boldsymbol{M}(x)$ captures the directional component shared by the two consecutive residual updates, whereas $\boldsymbol{R}(x)$ captures their directional difference.

And hence we define the differential-to-common ratio as

$$q(x) = \frac{\|\boldsymbol{R}(x)\|_F^2}{\|\boldsymbol{M}(x)\|_F^2}. \tag{36}$$

Proof. From the definitions of $\boldsymbol{M}(x)$ and $\boldsymbol{R}(x)$,

$$\begin{aligned} \langle \boldsymbol{M}(x), \boldsymbol{R}(x)\rangle_F &= \frac{1}{4}\left\langle \widehat{\boldsymbol{U}}_1(x) + \widehat{\boldsymbol{U}}_2(x), \widehat{\boldsymbol{U}}_1(x) - \widehat{\boldsymbol{U}}_2(x)\right\rangle_F \\ &= \frac{1}{4}\left(\|\widehat{\boldsymbol{U}}_1(x)\|_F^2 - \|\widehat{\boldsymbol{U}}_2(x)\|_F^2\right) = 0. \end{aligned} \tag{37}$$

Thus, the common and differential components are exactly orthogonal.

From the definition of $\boldsymbol{R}(x)$ and $\boldsymbol{M}(x)$,

$$\begin{aligned}\|\boldsymbol{R}(x)\|_F^2 &= \frac{1}{4}\left\|\widehat{\boldsymbol{U}}_1(x) - \widehat{\boldsymbol{U}}_2(x)\right\|_F^2 \\ &= \frac{1}{2}\left(1 - \left\langle \widehat{\boldsymbol{U}}_1(x), \widehat{\boldsymbol{U}}_2(x)\right\rangle_F\right).\end{aligned} \tag{38}$$

$$\begin{aligned}\|\boldsymbol{M}(x)\|_F^2 &= \frac{1}{4}\left\|\widehat{\boldsymbol{U}}_1(x) + \widehat{\boldsymbol{U}}_2(x)\right\|_F^2 \\ &= \frac{1}{2}\left(1 + \left\langle \widehat{\boldsymbol{U}}_1(x), \widehat{\boldsymbol{U}}_2(x)\right\rangle_F\right).\end{aligned} \tag{39}$$

Combining this with the definition of $C_2(x)$ yields

$$C_2(x) = 2\|\boldsymbol{R}(x)\|_F^2. \tag{40}$$

$$2 - C_2(x) = 2\|\boldsymbol{M}(x)\|_F^2. \tag{41}$$

For $C_2(x) < 2$, we consequently obtain

$$q(x) = \frac{\|\boldsymbol{R}(x)\|_F^2}{\|\boldsymbol{M}(x)\|_F^2} = \frac{C_2(x)}{2 - C_2(x)}. \tag{42}$$

Solving for $C_2(x)$ gives

$$C_2(x) = \frac{2q(x)}{1 + q(x)}. \tag{43}$$

Differentiating with respect to $q(x)$ yields

$$\frac{\mathrm{d}C_2(x)}{\mathrm{d}q(x)} = \frac{2}{(1 + q(x))^2} > 0. \tag{44}$$

Hence, $C_2(x)$ is strictly increasing in the differential-to-common energy ratio $q(x)$. □

B.4 Proof for Proposition 3: Finite-Step Curvature Increase.

Proof. From the geometric characterization of $C_2(x)$ established in Proposition 2,

$$\|\boldsymbol{M}(x)\|_F^2 = \frac{2 - C_2(x)}{2}, \qquad \|\boldsymbol{R}(x)\|_F^2 = \frac{C_2(x)}{2}. \tag{45}$$

Hence, between the two endpoints x_0 and x_T,

$$\|\boldsymbol{M}(x_T)\|_F^2 - \|\boldsymbol{M}(x_0)\|_F^2 = -\frac{1}{2}\left[C_2(x_T) - C_2(x_0)\right]. \tag{46}$$

and

$$\|\boldsymbol{R}(x_T)\|_F^2 - \|\boldsymbol{R}(x_0)\|_F^2 = \frac{1}{2}\left[C_2(x_T) - C_2(x_0)\right]. \tag{47}$$

Using the mode-selective MLP response relation in Equation 9, and noting that the fixed clean reference x_c cancels when taking the endpoint difference, we obtain

$$\begin{aligned}&\sum_{l=s}^{L-1} G_l^{\mathrm{MLP}}(x_T, x_c) - \sum_{l=s}^{L-1} G_l^{\mathrm{MLP}}(x_0, x_c) \\ &= \gamma_m\left(\|\boldsymbol{M}(x_T)\|_F^2 - \|\boldsymbol{M}(x_0)\|_F^2\right) + \gamma_r\left(\|\boldsymbol{R}(x_T)\|_F^2 - \|\boldsymbol{R}(x_0)\|_F^2\right) \\ &= -\frac{\gamma_m - \gamma_r}{2}\left[C_2(x_T) - C_2(x_0)\right].\end{aligned} \tag{48}$$

Since $\gamma_m > \gamma_r$ and, by assumption, $C_2(x_T) > C_2(x_0)$, it follows that

$$\sum_{l=s}^{L-1} G_l^{\mathrm{MLP}}(x_T, x_c) - \sum_{l=s}^{L-1} G_l^{\mathrm{MLP}}(x_0, x_c) < 0. \tag{49}$$

which proves Proposition 3. □

C Appendix for Section 4

C.1 General Settings

Backbones and Visual Inputs We adopt the standard LLaVA-1.5-7B (Liu et al., 2024) and InternVL3.5-8B (Wang et al., 2025a) models. Following the prior pipeline (Schlarmann & Hein, 2023), LLaVA uses the OpenAI CLIP ViT-L/14 (Radford et al., 2021) visual encoder with 224×224 inputs, and each image x is padded to square, bicubically resized, and normalized using the CLIP statistics. InternVL follows its native dynamic-resolution preprocessing, partitioning each image into $N \in [1, 12]$ non-overlapping 448×448 tiles and adding a global thumbnail when $N > 1$, with ImageNet normalization. For both models, the residual discrete curvature C_k is computed from visual-encoder hidden states before multimodal projection, and the final clean–adversarial representation discrepancy $E_L(x_a, x_c)$ is computed from the last visual-block features before the projector.

Method Setup We use use fixed curvature of $C_2(x)$ for LLaVA and $C_4(x)$ for InternVL. For both models, the CLS token is removed, and all remaining patch-token residual updates are concatenated and flattened into a single vector before computing one global cosine similarity in Equation 7 (Including all of those dynamic InternVL tiles). We use fixed defense steps of $T = 3$ for LLaVA and $T = 4$ for InternVL, with a per-step update size of $\alpha = 0.015$. The observed input is used for initialization without a random start, the input gradient is recomputed after each step, and the updated pixels are clipped to $[0, 1]$ without additional projection. All model parameters remain frozen throughout inference, and gradients are computed only with respect to the input image. As shown in Figure 1(c), clean and adversarial samples exhibit clearly separated early residual curvature distributions. For both models, We therefore set the gating threshold to the 5th percentile of the clean curvature distribution estimated from natural ImageNet-1K samples. As shown in Figure 4, for each task, we estimate the early residual curvature distributions of paired clean and adversarial samples using one-dimensional Gaussian kernel density estimation with Scott's bandwidth rule. Across different task types, the intersections of the clean and adversarial distributions consistently occur near the 5th percentile of the clean distribution, with the resulting curvature thresholds concentrated around 1.12 for LLaVA and 0.86 for InternVL.

C.2 Details of Evaluation On Generative Tasks

We evaluate generative tasks on COCO, Flickr30k, TextVQA, and VQAv2 using their standard evaluation splits and metrics (CIDEr score (Vedantam et al., 2015) and VQA accuracy (Vedantam et al., 2015; Antol et al., 2015)). Following the protocol proposed by RobustVLM (Schlarmann et al., 2024), we evaluate clean performance on the full evaluation split and adversarial performance on randomly selected 500 samples (`seed = 42`) generated using white-box ensemble APGD under an ℓ_∞ budget of $4/255$, with 100 iterations per branch. For VQA, the prompt is "`<image> question Answer the question using a single word or phrase.`", and for captioning, it is "`<image> Provide a short caption for this image.`". In Table 1, both training time methods adopt the vision encoder trained with an ℓ_∞ budget of 4/255, and test-time methods is based on the standard CLIP-ViT-L/14 vision encoder.

To extend the attack to InternVL3.5-8B, whose native dynamic-resolution preprocessing produces a variable number of 448×448 visual tiles and an optional thumbnail, we losslessly pack all views into a single APGD tensor, jointly optimize their RGB values in $[0, 1]$ under the same ℓ_∞ constraint, and unpack them into the original tile sequence before forward pass. Since InternVL's standard inference interface does not expose the differentiable loss required by ensemble APGD, we implement a differentiable forward path that inserts the visual embeddings into the corresponding `<IMG_CONTEXT>` token positions and propagates the teacher-forced output loss back to the input pixels.

C.3 Details of Evaluation On Zero-Shot Classification

We evaluate all datasets on their official test splits. Following R-TPT (Sheng et al., 2025), we generate adversarial examples using standard PGD under an ℓ_∞ budget of $4/255$, a step size of $1/255$, with 100 iterations per sample. CURB is only applied to the original input and just conducting the same multi-view augmentation (Hendrycks et al., 2020) followed by ensemble. Specifically, under the single-view setting, CURB is directly applied to the standard CLIP vision encoder without any

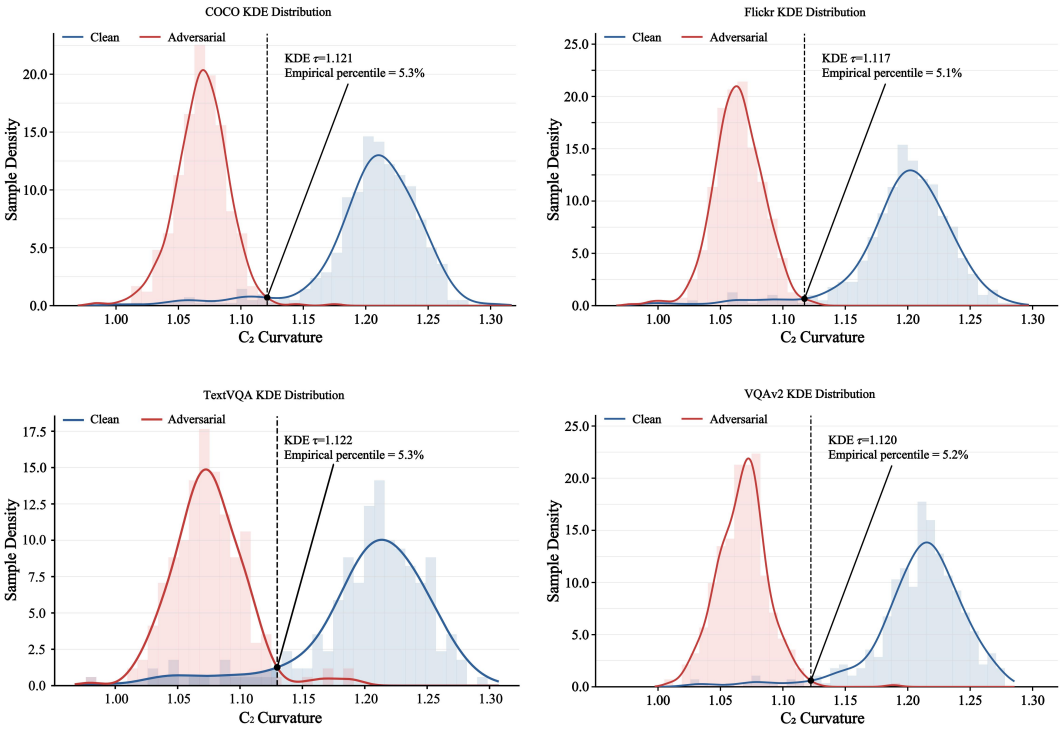

Figure 4: Distributions of early residual curvature of LLaVA-1.5-7B on (a) COCO, (b) Flickr30k, (c) TextVQA, and (d) VQAv2. Histograms show the sample densities of residual curvature for test samples, while the solid curves show their Gaussian kernel density estimates. The dashed vertical line and black marker denote the intersection of the clean and adversarial densities, and the empirical percentile reports the proportion of clean samples below this intersection.

additional modifications. Under the multi-view setting, CURB is built upon the simple ensemble pipeline, which aggregates predictions across multiple augmented views using the standard CLIP VLM (Radford et al., 2021). For multi-view generation, we adopt AugMix (Hendrycks et al., 2020) following previous methods (Sheng et al., 2025; Li et al., 2026; Hai et al., 2026) and strictly use the same model checkpoint, number of views ($N = 64$), and random seed (seed = 0).

C.4 Details Under Adaptive Attacks

Following standard practice (Schlarmann et al., 2024), we employ Backward Pass Differentiable Approximation (BPDA) (Athalye et al., 2018) combined with AutoAttack (Croce & Hein, 2020). Specifically, for adaptive attacks, we retain the same LVLM attack setup described above, but include the test-time CURB in the attack forward pass and use BPDA to approximate gradients through the iterative defense during backpropagation. Due to the higher computational cost, we conduct this adaptive evaluation on 200 samples on each dataset. For both robustly trained vision encoder, we employ the version trained with an ℓ_∞ budget of $4/255$.

C.5 Analysis of Inference Efficiency

We provide results of the optimization step ablation in Figure 3, and we report the detailed results in Tables 7 and 8. Notably, the optimal step number is consistent across all individual datasets within each architecture. For LLaVA-1.5-7B, 3 optimization steps achieve the best adversarial performance on COCO, Flickr30k, TextVQA, and VQAv2. Similarly, for InternVL3.5-8B, 4 steps consistently yield the best adversarial performance across all four datasets. This consistency further shows that the selected step number is architecture-specific. Additionally, as shown in Table 9, CURB can trade robustness for efficiency by adjusting the number of optimization steps. Reducing the steps from 3 to 1 cuts per-sample inference time to roughly 40% in average while preserving strong robustness. While training-time defenses spend a lot of resources at one time during training.

Table 7: Effect of the number of optimization steps on clean and adversarial performance across generative tasks via LLaVA-1.5-7B.

Optimization Step	COCO		Flickr30k		TextVQA		VQAv2		Average	
	Clean	Adv.	Clean	Adv.	Clean	Adv.	Clean	Adv.	Clean	Adv.
$T = 1$	115.5	100.3	77.7	62.5	37.1	22.6	74.6	58.4	76.2	60.9
$T = 2$	115.5	110.4	77.6	68.0	37.0	25.3	74.6	62.9	76.2	66.7
$T = 3$	**115.4**	**116.0**	**77.2**	**73.9**	**36.9**	**28.0**	**74.5**	**65.3**	**76.0**	**70.8**
$T = 4$	115.4	114.2	77.1	72.9	36.9	27.4	74.5	65.2	76.0	69.9
$T = 5$	115.3	113.9	76.9	73.3	36.6	27.7	74.5	66.0	75.8	70.2
$T = 6$	115.4	115.9	76.9	71.6	36.5	26.4	74.5	66.5	75.8	70.1
$T = 7$	115.3	114.1	77.3	72.8	36.4	26.5	74.5	65.9	75.8	69.8
$T = 8$	115.2	110.9	77.0	72.2	36.3	24.6	74.4	66.0	75.7	68.4
$T = 9$	115.3	110.9	76.9	71.5	36.2	24.3	74.4	65.3	75.7	68.0

Table 8: Effect of the number of optimization steps on clean and adversarial performance under ensemble APGD attacks across generative tasks via InternVL3.5-8B.

Optimization Step	COCO		Flickr30k		TextVQA		VQAv2		Average	
	Clean	Adv.	Clean	Adv.	Clean	Adv.	Clean	Adv.	Clean	Adv.
$T = 1$	110.8	50.2	87.3	26.2	76.7	25.1	75.3	41.7	88.0	35.8
$T = 2$	109.3	72.6	86.5	49.1	76.4	36.6	75.1	55.1	86.8	53.3
$T = 3$	108.1	79.5	85.7	55.4	76.2	40.4	74.6	56.1	86.1	57.8
$T = 4$	**106.7**	**81.0**	**84.1**	**56.0**	**75.8**	**41.3**	**74.6**	**56.9**	**85.3**	**58.8**
$T = 5$	106.2	80.5	84.0	55.3	75.8	40.2	73.7	56.5	84.9	58.1
$T = 6$	105.4	78.9	82.3	53.1	75.3	37.5	73.2	56.5	84.0	56.5
$T = 7$	104.7	78.6	82.1	49.3	74.6	38.4	73.0	55.4	83.6	55.4
$T = 8$	103.5	72.9	80.6	47.1	73.9	36.2	72.8	55.1	82.7	52.8
$T = 9$	102.7	72.1	80.2	46.2	73.4	35.5	73.3	54.5	82.4	52.1

Table 9: Running time and adversarial performance under ensemble APGD attacks across generative tasks via LLaVA-1.5-7B.

Methods		COCO	Flickr30k	TextVQA	VQAv2	Average
FARE	**Running time**	705600 GPU·s / 2 epochs				
	Adv.	40.9	22.8	24.6	10.9	24.8
AdPO	**Running time**	410320 GPU·s / 2 epochs				
	Adv.	47.8	24.1	10.1	32.4	28.6
CURB (1-step)	**Running time**	0.37 s/sample	0.59 s/sample	0.33 s/sample	0.23 s/sample	0.38 s/sample
	Adv.	100.3	62.5	22.6	58.4	60.9
CURB (2-step)	**Running time**	0.67 s/sample	1.13 s/sample	0.59 s/sample	0.32 s/sample	0.68 s/sample
	Adv.	110.4	68.0	25.3	62.9	66.7
CURB (3-step)	**Running time**	0.98 s/sample	1.65 s/sample	0.82 s/sample	0.40 s/sample	0.96 s/sample
	Adv.	116.0	73.9	28.0	65.3	70.8