DARKLLM: LEARNING LANGUAGE-DRIVEN ADVERSARIAL ATTACKS WITH LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

While vision and multimodal foundation models underpin critical tasks from perception to complex reasoning, they remain highly vulnerable to adversarial attacks. However, traditional adversarial attacks are typically limited to single, predefined objectives, tightly coupling each attack to a specific model or task, which restricts their scalability and flexibility in real-world scenarios. In this work, we present **DarkLLM**, a novel attack framework that trains an LLM to translate natural-language attack instructions into latent attack vectors, which are then decoded into visual adversarial perturbations. By leveraging natural-language instruction tuning, DarkLLM not only unifies targeted, untargeted, segmentation, and multi-model attacks within a single framework, but also achieves flexible and controllable adversarial generation, enabling each instruction to produce a perturbation that induces desired behaviors across heterogeneous models. Through extensive experiments across 4 tasks, 13 datasets, and 15 models, we demonstrate that DarkLLM with only 1B parameters can follow attacker instructions and generate highly effective attacks against CLIP, SAM, and frontier LLMs, revealing a systemic vulnerability in modern foundation models.

1 INTRODUCTION

Large vision and multimodal foundation models (Li et al., 2023; Liu et al., 2023a) have emerged as the central interface for perceiving and reasoning over visual content, underpinning a wide spectrum of applications such as image retrieval (Radford et al., 2021), segmentation (Kirillov et al., 2023), and multimodal assistants (Achiam et al., 2023; Team et al., 2023). Yet their impressive capabilities are shadowed by a long-standing weakness: *small, often imperceptible adversarial perturbations can reliably induce incorrect or even adversary-specified model behaviors* (Szegedy et al., 2013; Goodfellow et al., 2014b). Consequently, understanding and stress-testing the adversarial robustness of large-scale vision and multimodal models has become a pressing and practically relevant safety problem (Madry et al., 2017; Ma et al., 2026).

Over the past decade, adversarial research has been dominated by gradient-based optimization (Goodfellow et al., 2014b; Madry et al., 2017) and generative adversarial modeling (Goodfellow et al., 2014a; Xiao et al., 2018), revealing pervasive vulnerabilities across both classical architectures and modern foundation models (Moosavi-Dezfooli et al., 2017; Zhao et al., 2023). Yet two fundamental challenges persist: *transferability* and *scalability*. First, an effective attack should be transferable across heterogeneous models and tasks—often even jointly, as contemporary systems routinely integrate recognition, segmentation, and multimodal reasoning modules (Zhang et al., 2022; Zhou et al., 2024; Zhao et al., 2023). Equally importantly, a practical attack framework should be scalable, allowing different adversarial intents to be flexibly specified without requiring objective-specific pipeline redesign or repeated optimization (Zhang et al., 2025; Fang et al., 2024). However, existing approaches struggle to satisfy these requirements simultaneously. Most untargeted attacks that aim to induce arbitrary errors are tailored to specific architectures or foundation models (e.g., ViT (Wei et al., 2022; Mahmood et al., 2021; Shao et al., 2021), CLIP (Zhang et al., 2022; Zhou et al., 2023; Fang et al., 2025), or SAM (Zhou et al., 2024; Zhang et al., 2023b; Lu et al., 2024)), while representative targeted attacks, such as SU (Wei et al., 2023), SASD (Wu et al., 2024), TTAA (Wang et al., 2023), TTP (Naseer et al., 2021), SSA-CWA (Dong et al., 2023), and M-Attack (Li et al., 2025b), steer model outputs toward adversary-specified targets but typically follow a rigid "one-target, one-optimization" paradigm, requiring re-optimization for each new adversarial objective. These limitations raise a

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.

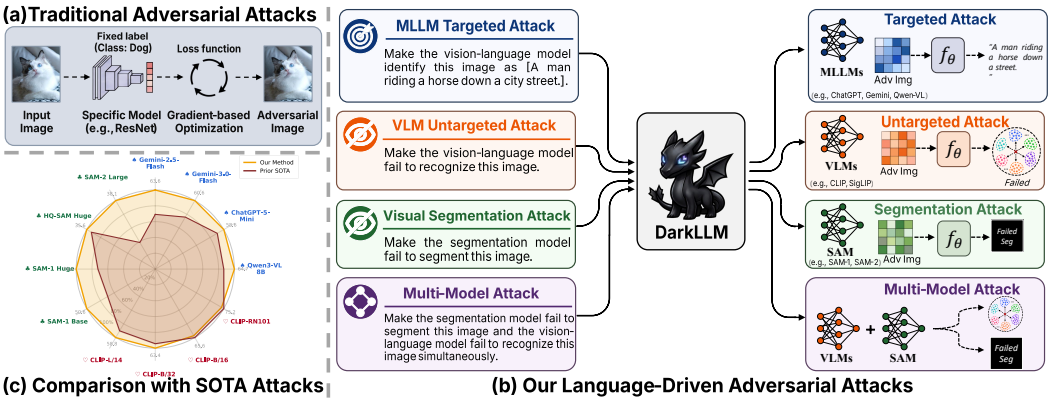


Figure 1: (a) An illustration of traditional adversarial attacks. (b) DarkLLM translates natural-language instructions into customized visual perturbations, enabling unified and controllable adversarial attacks. (c) DarkLLM achieves strong attack success rates across diverse models.

critical question: *Is it possible to design a new adversarial attack framework that is transferable and scalable across models and objectives?*

In parallel, the emergence of Large Language Models (LLMs) (Brown et al., 2020) has fundamentally reshaped how complex tasks are specified and solved. LLMs excel at interpreting semantically rich, compositional instructions and translating them into structured, domain-specific outputs, enabling breakthroughs across visual understanding (Liu et al., 2023a; Xie et al., 2024), generation (Tian et al., 2024; Chen et al., 2025), and medical analysis (Peng et al., 2023). Within the safety domain, LLMs have already been exploited to generate jailbreak prompts (Liu et al., 2023b; 2024), automate safety evaluations (Ying et al., 2026), and design textual backdoors (Li et al., 2025a). Yet, existing work largely remains confined to the textual domain, where LLMs are used to generate adversarial or malicious text, rather than pixel-level adversarial perturbations, motivating an intriguing problem: *Can the semantic reasoning capabilities of LLMs be leveraged for visual adversarial generation?*

In this work, we propose a novel language-driven attack framework, **DarkLLM**, which leverages natural-language instructions as the interface for adversarial generation. Figure 1 highlights the distinction between DarkLLM and existing attack strategies. Unlike conventional attacks, DarkLLM reformulates adversarial generation as a language-conditioned synthesis problem, enabling each instruction to be translated into a visual perturbation that induces desired adversarial behaviors across heterogeneous foundation models. Specifically, our approach trains an LLM to act as an adversarial controller, mapping semantically rich attack instructions into latent attack representations, which are then decoded by a lightweight perturbation generator into pixel-level adversarial noise. Through large-scale instruction tuning, DarkLLM learns to associate different attack intents with corresponding generators, allowing targeted, untargeted, segmentation, and multi-model attacks to be flexibly generated within a unified framework. We systematically analyze several key factors for improving attack effectiveness, including prompt ensemble strategies, loss function design, surrogate model selection, and training strategies. Finally, we conduct extensive experiments across 4 tasks and 15 models (CLIP (Radford et al., 2021), SAM (Kirillov et al., 2023; Ravi et al., 2024), Gemini (Team et al., 2023), ChatGPT (Achiam et al., 2023)), demonstrating that DarkLLM achieves strong adversarial effectiveness across all evaluated foundation models.

Our main contributions are summarized as follows:

- We propose **DarkLLM**, a novel language-driven adversarial attack framework that reformulates visual adversarial generation as an instruction-conditioned synthesis problem, enabling diverse attack objectives to be flexibly specified through natural language.
- We develop a unified adversarial generation paradigm that leverages LLM-based adversarial controllers and modular perturbation generators to jointly support targeted, untargeted, segmentation, and multi-model attacks within a single framework.
- We conduct extensive experiments across 4 tasks and 15 models, covering CLIP, SAM, and frontier LLMs. Both qualitative and quantitative results demonstrate that DarkLLM achieves strong adversarial effectiveness and robustness against common image defenses.

2 Related Work

Conventional Adversarial Attacks. The vulnerability of deep neural networks to adversarial examples remains a foundational challenge in machine learning (Szegedy et al., 2013). Traditional research divides into white-box attacks, which leverage full model access (e.g., gradients) to craft perturbations via optimization methods like FGSM (Goodfellow et al., 2014b) and PGD (Madry et al., 2017), and black-box attacks, which operate under restricted access. Black-box strategies typically rely on query-based estimation (Ilyas et al., 2018; Andriushchenko et al., 2020) or transfer-based generation using surrogate models (Dong et al., 2018; Xie et al., 2019; Huang et al., 2025; Li et al., 2025b). While effective on specific tasks, conventional methods typically operate under a fixed optimization framework tailored to a single target. This constraint severely limits their scalability across the diverse and dynamic task spectrum of modern foundation models.

Adversarial Attacks on Foundation Models. The rise of large vision foundation models has introduced architectures with unprecedented generalization (Radford et al., 2021; Kirillov et al., 2023; Liu et al., 2023a). Yet, recent studies confirm that this capability does not equate to robustness; these models remain highly susceptible to adversarial perturbations (Zhang et al., 2022; Zhao et al., 2023). Existing research has systematically exposed these vulnerabilities across different domains: adversarial attacks targeting CLIP's modality alignment include specific methods like Co-Attack (Zhang et al., 2022), AdvCLIP (Zhou et al., 2023), C-PGC (Fang et al., 2025) and X-Transfer (Huang et al., 2025); vulnerabilities in SAM's segmentation boundaries have been exploited by attacks such as Attack-SAM (Zhang et al., 2023b), DarkSAM (Zhou et al., 2024), UAD (Lu et al., 2024), and UAP-SAM2 (Zhou et al., 2025); and the multimodal reasoning of large VLMs has been compromised by approaches like AttackVLM (Zhao et al., 2023), JailbreakV-28K (Luo et al., 2024), AnyAttack (Zhang et al., 2025), FOA-Attack (Jia et al., 2025), and M-Attack (Li et al., 2025b). However, these efforts have largely operated in isolation, treating each model and task as a distinct optimization problem with specialized objectives. In contrast, our work aims to unify diverse attack settings within a single language-driven framework, enabling transferable adversarial generation across heterogeneous architectures.

3 Proposed Method

Threat Model. We focus on a transfer-based black-box attack scenario. The adversary aims to synthesize an adversarial example x_{adv} using accessible surrogate models $\mathcal{F}_s$, such that it successfully deceives unseen target models $\mathcal{F}_t$. We formulate this process as an *Language-guided adversarial attack*. Unlike traditional methods bound to fixed labels, our framework flexibly constructs diverse adversarial objectives via natural language. Let $\mathcal{D} = \{(x, y)\}$ be a dataset of clean images and annotations. The goal is to craft $x_{\text{adv}} = x + \delta$ that misleads $\mathcal{F}_t$, where the specific adversarial goal is dynamically instantiated from a user instruction I. For instance, given an instruction targeting a multimodal large language model, the objective is to align x_{adv} with a malicious target caption y_t (Zhang et al., 2025); conversely, for a promptable segmentation model like SAM, the instruction directs the attack to suppress all mask predictions (i.e., "segment nothing"), regardless of visual prompts (Zhou et al., 2024).

3.1 DarkLLM Framework

Embedding as Attack. Conventional adversarial attacks typically formulate adversarial objectives as task-specific optimization targets, tightly coupling each attack behavior with a dedicated optimization pipeline. While effective for individual settings, such formulations make attacks difficult to adapt across heterogeneous models, tasks, and objectives, resulting in fragmented methods separately designed for classification, segmentation, or targeted generation. In contrast, DarkLLM reformulates adversarial generation as a language-conditioned synthesis process, where natural-language instructions are interpreted by an LLM and mapped into high-level latent attack embeddings that condition downstream perturbation generators, enabling diverse attack behaviors to be flexibly generated within a unified framework.

To this end, we propose the *Embedding as Attack* paradigm, a novel approach that repurposes a Large Language Model (LLM) to act as a versatile and language-guided adversarial controller. The overall pipeline is illustrated in Figure 2. Specifically, central to this paradigm is the introduction of

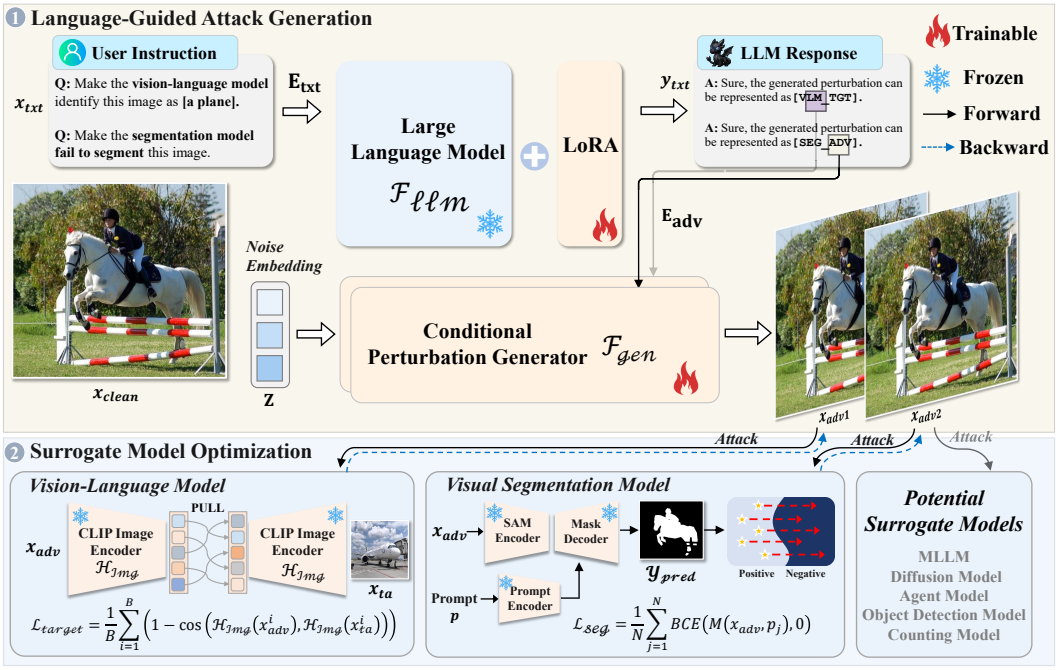


Figure 2: The framework of DarkLLM consists of two main stages. **1)** Language-Guided Attack Generation, where a LLM controller interprets a user's instruction and directs a conditional generator to synthesize the perturbation. **2)** Surrogate Model Optimization, where the generated adversarial example is optimized using a diverse suite of surrogate models to encourage transferability.

a new set of special control tokens (e.g., `[VLM_TGT]`, `[SEG_ADV]`) into the LLM's vocabulary, where each token serves as a dedicated trigger for invoking a corresponding adversarial generator to synthesize a specific type of perturbation. Our framework begins when the core LLM controller, $\mathcal{F}_{\text{LLM}}$, receives the language embeddings $\mathbf{E}_{\text{txt}}$ of a user's instruction x_{txt}. Based on its interpretation of the instruction, the controller decides whether to generate an adversarial attack. If so, it strategically embeds the specific control token within its textual response y_{txt}:

$$y_{txt} = \mathcal{F}_{\text{LLM}}(\mathbf{E}_{\text{txt}}). \tag{1}$$

The presence of the control token in the generated response serves as a trigger, activating the attack generation module. To be specific, we extract the final-layer hidden state $\tilde{\mathbf{h}}_{\text{adv}}$ corresponding to the control token, which represents the LLM's latent understanding of the required attack. This latent attack embedding is then projected through a two-layer MLP, γ, to form a refined attack signal, $\mathbf{E}_{\text{adv}}$. This signal conditions a trainable perturbation generator, $\mathcal{F}_{\text{gen}}$, guiding it to synthesize the appropriate adversarial perturbation δ_{adv}. The generation process is formulated as:

$$\mathbf{E}_{\text{adv}} = \gamma(\tilde{\mathbf{h}}_{\text{adv}}), \delta_{\text{adv}} = \mathcal{F}_{\text{gen}}(\mathbf{E}_{\text{adv}}). \tag{2}$$

Finally, the perturbation is added to a clean image x_{clean} to produce the final adversarial example, $x_{\text{adv}} = x_{\text{clean}} + \delta_{\text{adv}}$.

Conditional Perturbation Generator. To translate the high-level adversarial representation $\mathbf{E}_{\text{adv}}$ into an image-space perturbation δ_{adv}, we employ a conditional decoder network that progressively refines latent noise embeddings through attention blocks and spatial upsampling, as briefly illustrated in Figure 3. Specifically, the generator $\mathcal{F}_{\text{gen}}$, parameterized by θ_G, accepts two inputs: the LLM-derived attack embedding $\mathbf{E}_{\text{adv}} \in \mathbb{R}^{1\times C}$ and a shared learnable noise embedding $\mathbf{Z} \in \mathbb{R}^{D\times 3\times 3}$, where C and D denote their respective channel dimensions. The noise embedding is randomly initialized, optimized during training, and fixed at inference. Since generation depends only on this fixed embedding and the attack instruction rather than the source image, each instruction-conditioned perturbation can be reused across images:

$$\delta_{\text{adv}} = \mathcal{F}_{\text{gen}}(\mathbf{E}_{\text{adv}}, \mathbf{Z}; \theta_G), \quad \|\delta_{\text{adv}}\|_\infty \leq \epsilon. \tag{3}$$

Each generator block first applies self-attention to capture dependencies among noise features, followed by cross-attention to inject the adversarial intent derived from the LLM. The noise features serve as queries, while $\mathbf{E}_{\text{adv}}$ provides both the key and value. For noise features $\mathbf{X}$, the residual update is:

$$\mathbf{X}' = \mathbf{X} + \text{CrossAttn}\big(\text{LN}(\mathbf{X}), \mathbf{E}_{\text{adv}}\big), \tag{4}$$

where LN denotes layer normalization. With a single conditioning token, the attention weights are identically one, broadcasting a shared projected value. This value is added to the noise features to preserve their spatial differences at this update. Cross-attention thus provides global semantic conditioning without forcing spatially uniform perturbations. Transposed convolutions subsequently upsample the conditioned features. We further use Adaptive Group Normalization (AdaGN) (Dhariwal & Nichol, 2021) to modulate normalized features with channel-wise scale and shift parameters predicted from $\mathbf{E}_{\text{adv}}$, guiding refinement toward the user's specified adversarial goal. Finally, the output passes through Tanh, is resized to the target image dimensions by bilinear interpolation, and is scaled by ϵ to satisfy the perturbation budget.

3.2 Training the DarkLLM

We employ two representative classes of large vision foundation models as surrogates for optimizing DarkLLM: the coarse-grained vision-language model, CLIP (Radford et al., 2021), and the fine-grained promptable visual segmentation model, SAM (Kirillov et al., 2023).

Optimization for Fine-Grained Segmentation. To extend our framework to fine-grained visual segmentation, we incorporate SAM as a key surrogate model. However, directly attacking SAM poses two challenges: (1) *Prompt Transferability:* perturbations optimized for a specific object or prompt type (e.g., a point prompt) often fail to generalize to other prompts; (2) *Backbone Transferability:* perturbations optimized on one SAM backbone exhibit degraded transfer to other backbones. To improve prompt transferability, we adopt a multi-prompt ensemble optimization strategy. For each image, we randomly sample a diverse set of prompts $\mathcal{P} = \{p_1, \ldots, p_N\}$ covering both point and box prompts, and optimize a single perturbation to suppress the predicted masks under all sampled prompts, thereby encouraging prompt-agnostic "segment nothing" behavior. Nevertheless, backbone transfer remains non-trivial. Through careful ablation studies, we find that poor cross-backbone transfer largely stems from widely used regression-style losses (e.g., Mean Squared Error), which over-emphasize matching backbone-specific activation magnitudes and cause perturbations to overfit a particular SAM variant. To address this issue, we instead optimize a binary cross-entropy (BCE) objective that pushes mask predictions toward the all-background decision boundary, yielding gradients that are more semantically aligned and less sensitive to backbone-specific activations. Concretely, we define:

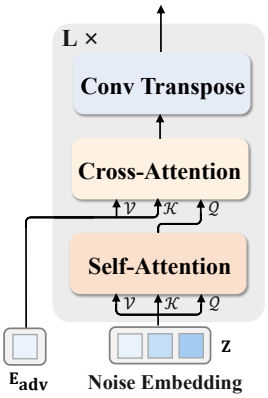


Figure 3: A brief illustration of our conditional perturbation generator.

$$\mathcal{L}_{\text{seg}} = \frac{1}{N} \sum_{j=1}^{N} \text{BCE}(M(x_{\text{adv}}, p_j), \mathbf{0}), \tag{5}$$

where $M(x_{\text{adv}}, p_j)$ denotes the mask logits predicted by SAM for the image x_{adv} under prompt p_j. We also use a MobileSAM (Zhang et al., 2023a) as an additional surrogate to improve transferability.

Optimization for Vision-Language Models. Achieving high cross-model transferability is paramount when attacking the diverse family of CLIP models. To this end, we adopt a surrogate scaling strategy inspired by X-Transfer (Huang et al., 2025). Specifically, during distributed training, each computational rank (e.g., a GPU) is assigned a different surrogate from a large pool of CLIP models with varying architectures and pre-training datasets, ensuring that gradients from a diverse set of models are jointly incorporated and thereby enhancing overall transferability. Our framework supports both untargeted and targeted attacks, with the specific attack objective dynamically determined by the user's instruction. In both cases, optimization is performed in the CLIP embedding space using cosine similarity. For untargeted attacks, the objective is to disrupt the original image–text alignment by minimizing the similarity between the adversarial image embedding and the benign image embedding. For targeted attacks, the objective is reversed: to maximize the similarity between the adversarial image embedding and the embedding of a target image x_{ta}. These objectives

Table 1: Non-targeted ASR (%) results for zero-shot classification, image-text (I-T) retrieval, and image captioning tasks. The best results are **boldfaced**, and the second-best are underlined. *Results for classification and retrieval are averaged over 4 black-box CLIP encoders.*

Method	Variant	Zero-Shot Classification								I-T Retrieval		Image Captioning	
		C-10	C-100	Food	GTSRB	ImageNet	Cars	STL	Avg	TR@1	IR@1	MSCOCO	Flickr30k
GD-UAP Mopuri et al. (2018)	Seg	63.2	78.2	36.1	58.2	22.2	19.9	9.8	41.0	26.3	21.8	8.1	6.5
	CLS	66.5	80.4	55.2	61.7	29.8	28.8	20.1	48.9	29.5	24.7	10.3	8.6
TRM-UAP Liu et al. (2023c)	GoogleNet	73.3	84.0	60.7	71.1	33.0	30.5	24.7	53.9	30.0	25.7	6.6	10.0
	RN152	58.0	73.9	54.4	63.7	27.9	27.8	19.1	46.4	26.1	22.1	2.8	7.3
Meta-UAP Weng et al. (2024)	Ensemble	82.7	93.2	59.8	64.9	39.4	39.2	27.7	58.1	37.1	32.1	16.3	15.0
	Ensemble-Meta	72.3	88.5	53.3	59.3	35.8	34.6	22.1	52.2	35.3	29.3	11.3	12.7
ETU Zhang et al. (2024)	RN50-Flickr	44.2	64.0	32.7	46.9	21.8	21.6	14.5	35.1	18.0	15.5	7.4	5.2
	ViT-B/16-Flickr	76.2	89.7	67.8	68.5	51.5	53.4	40.7	63.9	49.0	46.5	17.8	16.2
C-PGC Fang et al. (2025)	RN101-COCO	34.8	60.5	37.8	47.9	31.8	30.3	21.0	37.7	33.2	26.3	1.46	2.1
	ViT-B/16-COCO	74.4	90.3	65.1	72.2	53.0	56.7	36.8	64.0	51.6	52.7	11.5	10.5
X-Transfer Huang et al. (2025)	Vanilla	69.1	85.8	60.6	64.7	37.2	34.3	19.4	53.0	33.7	30.4	9.6	11.4
	Base	86.2	**97.3**	**83.6**	**86.2**	63.4	69.7	**49.3**	**76.5**	59.7	55.2	**21.0**	18.5
DarkLLM (Ours)	Ensemble	**87.4**	**97.3**	83.5	85.8	**64.7**	**71.7**	45.0	**76.5**	**62.9**	**59.3**	19.3	**19.0**

are formalized by the following loss functions:

$$\mathcal{L}_{\text{untarget}} = \frac{1}{B}\sum_{i=1}^{B} \cos\big(\mathcal{H}_{\text{img}}(x^i_{\text{adv}}), \mathcal{H}_{\text{img}}(x^i_{\text{clean}})\big)\,, \tag{6}$$

$$\mathcal{L}_{\text{target}} = \frac{1}{B}\sum_{i=1}^{B} \big(1 - \cos\big(\mathcal{H}_{\text{img}}(x^i_{\text{adv}}), \mathcal{H}_{\text{img}}(x^i_{\text{ta}})\big)\big)\,, \tag{7}$$

where $\mathcal{H}_{\text{img}}$ is the image encoder, and B is the batch size.

Decoupled Progressive Training. Unifying heterogeneous attack types within a single instruction-following framework is non-trivial, as different attacks require different levels of semantic understanding and exhibit highly imbalanced convergence behaviors. In our early joint-training experiments, we observed that untargeted attacks converge much faster than language-sensitive targeted attacks, and their rapidly updated parameters dominate optimization, significantly degrading targeted performance. To address this issue, we adopt an effective *decoupled progressive* training strategy. We assign unique control tokens (e.g., `[VLM_TGT]`, `[VLM_ADV]`, `[SEG_ADV]`, `[MULTI_ADV]`) and dedicated noise generators to each attack category, preventing gradient interference across incompatible objectives. Training proceeds in two stages. *(1) Targeted Instruction Tuning:* We first instruction-tune the LLM controller via LoRA on targeted QA-style attack data until convergence, enabling precise parsing and execution of semantically rich attack commands. *(2) Unified Attack Training:* We then initialize the controller with stage-1 weights and train all attack types jointly, while freezing the targeted generator to preserve its capability and learning additional untargeted and multi-model attacks through their respective generators. This decoupled progression yields stable and efficient training, while offering modularity and scalability for incorporating new attack types.

4 Experiments

4.1 Experimental Setup

Implementation Details. We implement DarkLLM using the XTuner (Contributors, 2023) codebase. The LLM is initialized from InternVL2.5 (Chen et al., 2024) and fine-tuned using LoRA (Hu et al., 2022) on 8 NVIDIA H200 GPUs. Training follows our decoupled progressive strategy. In stage 1, we jointly train on ImageNet (Deng et al., 2009) for 55 epochs and CC3M (Sharma et al., 2018) for 10 epochs using the AdamW optimizer with an initial learning rate of 1×10^{-4}. The per-GPU batch size is 128. In stage 2, we fine-tune the model on ImageNet for 8 epochs with a batch size of 64. The overall training objective is defined as a weighted sum of individual surrogate losses:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{t}}\mathcal{L}_{\text{txt}} + \lambda_{\text{seg}}\mathcal{L}_{\text{seg}} + \lambda_{\text{ta}}\mathcal{L}_{\text{target}} + \lambda_{\text{un}}\mathcal{L}_{\text{untarget}}, \tag{8}$$

The hyperparameters $\lambda_{\text{t}}, \lambda_{\text{seg}}, \lambda_{\text{ta}}$, and λ_{un} are set to 1.0, 10.0, 10.0, and 8.0, respectively, to balance different attack objectives. We apply an L_∞-norm constraint on all perturbations. Specifically, we set $\epsilon = 12/255$ for untargeted VLM attacks following (Huang et al., 2025; Fang et al., 2025),

Table 2: Targeted ASR (%) and AvgSim results for image captioning tasks across different commercial MLLMs. The best results among universal methods are **boldfaced**, and the second-best results are underlined.

Method	Variant	Qwen3-VL-8B		GPT-4o		GPT-4.1		GPT-5-mini		Gemini-2.5-Flash		Gemini-3-Flash	
		ASR	AvgSim	ASR	AvgSim	ASR	AvgSim	ASR	AvgSim	ASR	AvgSim	ASR	AvgSim
Sample-wise Adversarial Methods													
AttackVLM Zhao et al. (2023)	B/16	3.0	0.02	3.0	0.01	3.0	0.02	0.0	0.00	1.0	0.01	0.0	0.00
	L/14	4.0	0.03	3.0	0.02	1.0	0.01	1.0	0.00	2.0	0.02	1.0	0.00
	Ensemble	3.0	0.02	2.0	0.02	3.0	0.01	3.0	0.01	2.0	0.02	2.0	0.01
M-Attack Li et al. (2025b)	B/16	31.0	0.15	43.0	0.19	50.0	0.20	26.0	0.12	25.0	0.11	21.0	0.08
	L/14	52.0	0.28	65.0	0.31	66.0	0.31	34.0	0.16	53.0	0.25	42.0	0.17
	Ensemble	87.0	0.47	86.0	0.47	89.0	0.51	63.0	0.29	77.0	0.38	69.0	0.32
Universal Adversarial Methods													
AnyAttack Zhang et al. (2025)	coco-bi	56.0	0.24	60.0	0.22	59.0	0.25	53.0	0.22	44.0	0.19	46.0	0.18
	coco-cos	5.0	0.02	7.0	0.04	4.0	0.03	5.0	0.04	7.0	0.04	8.0	0.03
	flickr-bi	36.0	0.17	40.0	0.16	47.0	0.19	32.0	0.16	32.0	0.14	23.0	0.09
	flickr-cos	25.0	0.11	29.0	0.14	34.0	0.14	27.0	0.12	30.0	0.14	20.0	0.09
DarkLLM (Ours)	InternVL-1B	49.5	0.24	64.6	0.26	**67.6**	**0.27**	54.5	**0.24**	57.5	0.24	52.5	0.22
	InternVL-2B	**64.7**	**0.29**	**67.7**	**0.28**	64.6	**0.27**	**58.6**	**0.24**	**63.6**	**0.25**	**60.6**	**0.23**

Table 3: The mIoU (%) results for promptable image segmentation task across different SAM models. The best results are **boldfaced**, and the second-best results are underlined. Red numbers indicate the absolute mIoU drop caused by DarkLLM relative to the clean results.

Setting		SAM-Base			SAM-Huge			HQ-SAM-Huge			SAM2-Hiera-Large		
Prompt	Method	ADE	COCO	CITY	ADE	COCO	CITY	ADE	COCO	CITY	ADE	COCO	CITY
Point	Clean	50.3	44.9	38.0	53.6	48.5	39.7	52.4	47.3	32.0	57.2	55.2	41.7
	AttackSAM	46.8	42.0	24.6	51.7	47.5	32.2	51.0	46.5	26.8	55.9	55.0	41.4
	UAD	42.7	39.4	8.6	48.0	44.6	**16.0**	46.5	43.6	14.4	52.6	52.7	26.0
	DarkSAM-pt	43.2	41.6	29.6	53.1	48.7	38.4	50.1	46.7	30.6	52.5	49.1	38.5
	DarkSAM-bx	46.1	43.3	32.6	52.7	48.7	38.4	48.4	45.2	28.2	52.5	48.7	39.2
	UAP-SAM2	42.8	39.6	28.2	47.2	44.9	32.9	42.1	40.6	24.9	45.9	44.1	29.2
	PATA	44.0	40.1	13.0	48.7	45.0	23.3	47.5	44.3	21.2	53.0	52.7	31.5
	PATA++	43.9	40.1	13.2	48.6	44.9	23.3	47.5	44.2	21.3	52.7	52.8	31.7
	DarkLLM (Ours)	**2.9**(47.4↓)	**3.5**(41.4↓)	**0.4**(37.6↓)	**21.5**(32.1↓)	**24.3**(24.2↓)	19.2(20.5↓)	**18.9**(33.5↓)	**21.9**(25.4↓)	**14.2**(17.8↓)	**20.2**(37.0↓)	**23.4**(31.8↓)	**15.1**(26.6↓)
Box	Clean	72.8	71.4	61.2	74.5	73.2	61.5	69.4	65.2	50.2	74.9	73.9	62.7
	AttackSAM	70.8	69.4	49.3	73.5	72.4	56.3	68.4	64.9	45.9	74.3	73.5	59.0
	UAD	68.3	68.0	39.7	71.3	71.1	47.9	65.8	63.7	33.5	73.0	72.7	53.4
	DarkSAM-pt	67.4	67.5	51.3	73.7	72.9	58.8	67.6	64.4	46.1	73.9	73.2	59.5
	DarkSAM-bx	68.9	68.5	53.8	73.6	72.9	59.1	66.1	63.4	44.2	74.1	73.2	60.0
	UAP-SAM2	69.1	68.3	54.5	71.2	71.2	55.9	63.5	61.9	42.2	71.3	71.2	55.4
	PATA	68.8	68.1	43.1	71.7	71.1	50.4	66.4	63.8	38.4	73.1	72.7	54.9
	PATA++	68.7	68.0	43.5	71.6	71.1	50.5	66.4	63.8	38.6	73.1	72.7	55.0
	DarkLLM (Ours)	**27.4**(45.4↓)	**30.5**(40.9↓)	**30.2**(31.0↓)	**46.9**(27.6↓)	**55.1**(18.1↓)	**43.4**(18.1↓)	**40.9**(28.5↓)	**47.1**(18.1↓)	**32.3**(17.9↓)	**49.5**(25.4↓)	**55.7**(18.2↓)	**38.8**(23.9↓)

$\epsilon = 16/255$ for targeted MLLM attacks following (Zhang et al., 2025; Li et al., 2025b; Jia et al., 2025), and $\epsilon = 10/255$ for SAM attacks following (Zhou et al., 2024; 2025).

Baselines, Tasks, and Datasets. Please refer to the Appendix C. for details due to page limitations.

4.2 Main Results

Untargeted Attacks on Vision-Language Models. We evaluate the non-targeted attack performance of DarkLLM across three vision-language tasks, with results summarized in Table 1. Two key observations emerge. First, DarkLLM achieves strong and consistent performance across zero-shot classification and image captioning. It outperforms prior methods designed for both image classifiers (e.g., GD-UAP (Mopuri et al., 2018), TRM-UAP (Liu et al., 2023c)) and CLIP-based attacks (e.g., ETU (Zhang et al., 2024), C-PGC (Fang et al., 2025)), while remaining competitive with the strong baseline X-Transfer (Huang et al., 2025). On image captioning, DarkLLM attains ASRs of 19.3% on MSCOCO and 19.0% on Flickr30k, slightly surpassing X-Transfer. Second, DarkLLM demonstrates clear advantages on image–text retrieval, achieving higher ASR than all baselines, including X-Transfer, indicating improved transferability across CLIP encoders.

Targeted Attacks on Frontier MLLMs. We evaluate the black-box transferability of DarkLLM across six frontier commercial MLLMs, with results summarized in Table 2. As a universal method, DarkLLM consistently achieves strong performance compared to the previous state-of-the-art baseline, AnyAttack (Zhang et al., 2025), outperforming it on most evaluated models. Notably, DarkLLM achieves 67.7% ASR on GPT-4o and 60.6% on Gemini-3-Flash under the InternVL-2B setting, surpassing AnyAttack by clear margins. We further observe that scaling the LLM model from InternVL-1B to InternVL-2B improves most ASR and semantic similarity metrics. While sample-wise methods such as M-Attack (Li et al., 2025b) can achieve higher ASRs through per-image optimization, DarkLLM attains competitive results using a single universal perturbation, offering a more efficient and scalable alternative. Finally, as illustrated in Figure 4 and Figure 8, DarkLLM

Table 4: Non-targeted and targeted ASR (%) results of DarkLLM across different model families. The best results are **boldfaced**, and the second-best results are underlined. Red numbers indicate the absolute ASR improvement of DarkLLM over domain-specific methods.

Setting		SAM2-Hiera-Large			CLIP (Avg@4)				Commercial MLLMs			
Prompt	Method	ADE	COCO	CITY	Food	Image	Cars	STL	QwenVL	GPT-4o	GPT-5	Gemini
Box	DarkSAM	2.0	1.4	6.5	5.4	4.1	5.0	0.9	–	–	–	–
	X-Transfer	3.3	2.2	5.1	**83.6**	**63.4**	**69.7**	**49.3**	–	–	–	–
	AnyAttack	2.6	1.7	4.1	50.4	35.1	31.1	27.9	**56.0**	60.0	53.0	46.0
	DarkLLM	**24.8**(22.8↑)	**19.2**(17.8↑)	**30.9**(24.4↑)	68.8(14.8↓)	50.5(12.9↓)	50.1(19.6↓)	33.1(16.2↓)	49.5(6.5↓)	**64.6**(4.6↑)	**54.5**(1.5↑)	**52.5**(6.5↑)

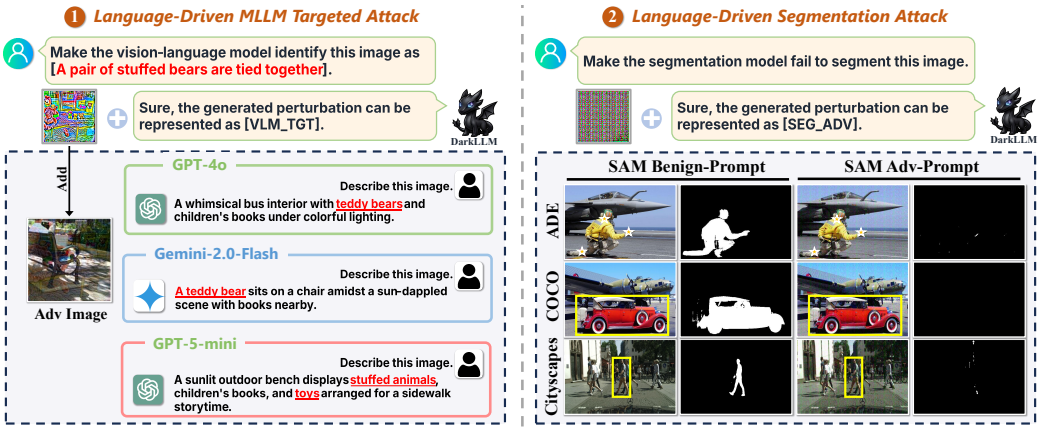
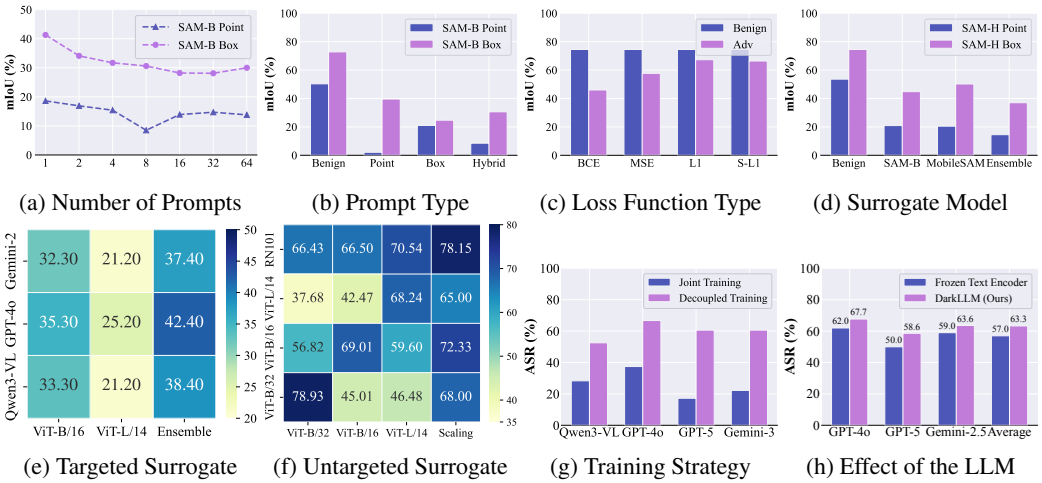


Figure 4: Visualization of DarkLLM for language-driven attacks.



(a) Number of Prompts (b) Prompt Type (c) Loss Function Type (d) Surrogate Model

(e) Targeted Surrogate (f) Untargeted Surrogate (g) Training Strategy (h) Effect of the LLM

Figure 5: Ablation study on the effects of different factors on the attack performance of DarkLLM.

is able to steer LLMs toward generating adversary-specified content via language instructions, highlighting the potential risks of language-guided adversarial generation.

Untargeted Attacks on Segmentation Models. We evaluate DarkLLM across four SAM variants and three datasets. Table 3 highlights three findings. **(1)** DarkLLM substantially reduces mIoU under both point and box prompts, demonstrating cross-prompt transferability without restricting the attack to a single prompt type. **(2)** The attack remains effective on SAM-Huge, HQ-SAM-Huge, and SAM2-Hiera-Large, demonstrating strong cross-backbone transfer beyond the SAM-Base surrogate. **(3)** DarkLLM outperforms UAD (Lu et al., 2024) in 23 of the 24 evaluated settings, including all 12 box-prompt settings and 11 of the 12 point-prompt settings. Figures 4 and 9–11 illustrate disrupted mask predictions across SAM models. Additional results are reported in Tables 13–15.

Unified Attack Scalability. We assess the scalability of DarkLLM across heterogeneous models, tasks, and domains. Table 4 reveals two findings. **(1)** DarkLLM achieves strong and consistent performance across all tasks. It induces substantial degradation on SAM2 across ADE, COCO, and CITY, while simultaneously attaining high ASRs on CLIP and commercial MLLMs. Compared

to prior methods, DarkLLM outperforms DarkSAM on segmentation and achieves competitive or superior performance to X-Transfer and AnyAttack across CLIP and MLLMs. **(2)** More importantly, Table 4 highlights a limitation of model-specific attacks: methods such as X-Transfer, tailored for CLIP, exhibit limited effectiveness on SAM2. In contrast, DarkLLM generates a *single* perturbation that simultaneously degrades SAM2 and maintains strong transferability to CLIP and MLLMs.

4.3 Method Analysis

In this section, we investigate the different factors on the attack performance of DarkLLM. Additional experiments of generalization, stability, ablations, loss-weight sensitivity, defenses, and perceptual quality are provided in Appendices C.2–C.4.

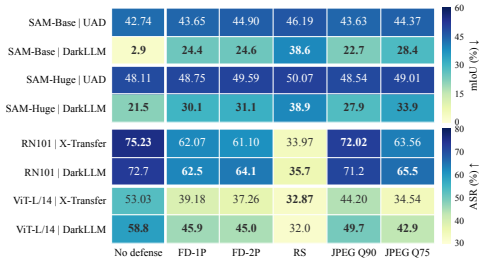


Figure 6: Attack effectiveness under defenses.

Attack Optimization on SAM. We analyze factors affecting attack effectiveness on SAM. Figure 5(a) shows that using multiple prompts improves attack effectiveness, with $N{=}8$ balancing performance and efficiency. Joint optimization with point and box prompts improves cross-prompt transfer, suggesting that prompt diversity reduces dependence on a specific prompt type (Figure 5(b)). BCE yields stronger transfer to SAM-Huge than MSE, ℓ_1, and Smooth-ℓ_1, highlighting the importance of loss selection for cross-backbone transfer (Figure 5(c)).

Surrogate Model Selection. We examine how surrogate diversity affects attack transferability. Combining SAM-Base with MobileSAM strengthens transfer to SAM-Huge over SAM-Base alone (Figure 5(d)). For targeted attacks, we use CLIP surrogates following (Zhao et al., 2023; Zhang et al., 2025). Ensembling ViT-B/16 and ViT-L/14 improves ASR across the evaluated MLLMs over either surrogate alone (Figure 5(e)). For untargeted attacks, a diverse CLIP surrogate pool generally improves transferability (Figure 5(f)). Together, these results suggest that optimizing against multiple surrogates helps reduce dependence on an individual model, consistent with prior findings (Huang et al., 2025; Li et al., 2025b; Zhang et al., 2025).

Training Strategy and LLM Conditioning. We further analyze the effects of training strategy and conditioning encoder on targeted attack performance. Figure 5(g) shows that decoupled progressive training consistently achieves higher ASR than joint training across the evaluated MLLMs, suggesting that separating heterogeneous attack objectives helps reduce optimization interference and preserve targeted attack capability. To assess the contribution of the LLM, we replace it with a frozen CLIP ViT-L/14 text encoder, retrain the framework under otherwise identical settings and evaluate targeted attacks using three commercial MLLMs. As shown in Figure 5(h), DarkLLM outperforms the frozen text encoder on all three models, increasing the average ASR from 57.0% to 63.3%.

Attack Effectiveness Under Defenses. We evaluate DarkLLM against Feature Distillation (Liu et al., 2019) (FD-1P/2P), Randomized Smoothing (Cohen et al., 2019) (RS), and JPEG compression (Guo et al., 2017) (Q90/Q75). Figure 6 compares DarkLLM with X-Transfer (Huang et al., 2025) on ImageNet classification using CLIP-RN101 and ViT-L/14, and with UAD (Lu et al., 2024) on ADE20K-val segmentation using SAM-Base and SAM-Huge with point prompts. DarkLLM achieves higher ASR in 8 of the 10 defended classification settings and lower mIoU in all defended segmentation settings. RS provides the strongest mitigation for both methods in each task, while DarkLLM retains satisfactory classification ASR and lower segmentation mIoU than UAD.

5 Conclusion

In this work, we propose DarkLLM, a language-guided adversarial attack framework that leverages large language models as dynamic controllers for visual adversarial generation. By bridging natural-language instructions with pixel-level perturbations, DarkLLM generalizes beyond traditional optimization-based attacks and unifies targeted, untargeted, segmentation, and multi-model attacks within a single framework. Comprehensive evaluations demonstrate that our approach achieves strong transferability across diverse vision foundation models.

AI USE STATEMENT

We primarily used generative AI tools to polish the writing and improve clarity. These tools also assisted with LaTeX formatting, figure-generation code, reference checking, and feedback on the presentation. The authors take responsibility for the final content.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square attack: a query-efficient black-box adversarial attack via random search. In *European conference on computer vision*, pp. 484–501. Springer, 2020.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pp. 446–461. Springer, 2014.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024.

Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 215–223. JMLR Workshop and Conference Proceedings, 2011.

Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *international conference on machine learning*, pp. 1310–1320. PMLR, 2019.

XTuner Contributors. Xtuner: A toolkit for efficiently fine-tuning llm. `https://github.com/InternLM/xtuner`, 2023.

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3213–3223, 2016.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9185–9193, 2018.

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

Hao Fang, Jiawei Kong, Bin Chen, Tao Dai, Hao Wu, and Shu-Tao Xia. Clip-guided generative networks for transferable targeted adversarial attacks. In *European Conference on Computer Vision*, pp. 1–19. Springer, 2024.

Hao Fang, Jiawei Kong, Wenbo Yu, Bin Chen, Jiawei Li, Hao Wu, Shu-Tao Xia, and Ke Xu. One perturbation is enough: On generating universal adversarial perturbations against vision-language pre-training models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4090–4100, 2025.

Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014a.

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

Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens Van Der Maaten. Countering adversarial images using input transformations. *arXiv preprint arXiv:1711.00117*, 2017.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Hanxun Huang, Sarah Erfani, Yige Li, Xingjun Ma, and James Bailey. X-transfer attacks: Towards super transferable adversarial attacks on clip. *arXiv preprint arXiv:2505.05528*, 2025.

Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. In *International conference on machine learning*, pp. 2137–2146. PMLR, 2018.

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

Alex K, Ben Hamner, and Ian Goodfellow. Nips 2017: Defense against adversarial attack. `https://kaggle.com/competitions/nips-2017-defense-against-adversarial-attack`, 2017. Kaggle.

Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3128–3137, 2015.

Lei Ke, Mingqiao Ye, Martin Danelljan, Yifan Liu, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Segment anything in high quality. In *NeurIPS*, 2023.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4015–4026, 2023.

Jonathan Krause, Jia Deng, Michael Stark, and Li Fei-Fei. Collecting a large-scale dataset of fine-grained cars. 2013.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 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.

Yige Li, Zhe Li, Wei Zhao, Nay Myat Min, Hanxun Huang, Xingjun Ma, and Jun Sun. Autobackdoor: Automating backdoor attacks via llm agents. *arXiv preprint arXiv:2511.16709*, 2025a.

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

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 *European conference on computer vision*, pp. 740–755. Springer, 2014.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023a.

Xiaogeng Liu, Nan Xu, Muhao Chen, and Chaowei Xiao. Autodan: Generating stealthy jailbreak prompts on aligned large language models. *arXiv preprint arXiv:2310.04451*, 2023b.

Xiaogeng Liu, Peiran Li, Edward Suh, Yevgeniy Vorobeychik, Zhuoqing Mao, Somesh Jha, Patrick McDaniel, Huan Sun, Bo Li, and Chaowei Xiao. Autodan-turbo: A lifelong agent for strategy self-exploration to jailbreak llms. *arXiv preprint arXiv:2410.05295*, 2024.

Yiran Liu, Xin Feng, Yunlong Wang, Wu Yang, and Di Ming. Trm-uap: Enhancing the transferability of data-free universal adversarial perturbation via truncated ratio maximization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4762–4771, 2023c.

Zihao Liu, Qi Liu, Tao Liu, Nuo Xu, Xue Lin, Yanzhi Wang, and Wujie Wen. Feature distillation: Dnn-oriented jpeg compression against adversarial examples. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 860–868. IEEE, 2019.

Jiahao Lu, Xingyi Yang, and Xinchao Wang. Unsegment anything by simulating deformation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24294–24304, 2024.

Weidi Luo, Siyuan Ma, Xiaogeng Liu, Xiaoyu Guo, and Chaowei Xiao. Jailbreakv: A benchmark for assessing the robustness of multimodal large language models against jailbreak attacks. *arXiv preprint arXiv:2404.03027*, 2024.

Xingjun Ma, Yifeng Gao, Yixu Wang, Ruofan Wang, Xin Wang, Ye Sun, Yifan Ding, Hengyuan Xu, Yunhao Chen, Yunhan Zhao, et al. Safety at scale: A comprehensive survey of large model and agent safety. *Foundations and Trends in Privacy and Security*, 8(3-4):1–240, 2026.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Kaleel Mahmood, Rigel Mahmood, and Marten Van Dijk. On the robustness of vision transformers to adversarial examples. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 7838–7847, 2021.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1765–1773, 2017.

Konda Reddy Mopuri, Aditya Ganeshan, and R Venkatesh Babu. Generalizable data-free objective for crafting universal adversarial perturbations. *IEEE transactions on pattern analysis and machine intelligence*, 41(10):2452–2465, 2018.

Muzammal Naseer, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Fatih Porikli. On generating transferable targeted perturbations. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 7708–7717, 2021.

Cheng Peng, Xi Yang, Aokun Chen, Kaleb E Smith, Nima PourNejatian, Anthony B Costa, Cheryl Martin, Mona G Flores, Ying Zhang, Tanja Magoc, et al. A study of generative large language model for medical research and healthcare. *NPJ digital medicine*, 6(1):210, 2023.

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

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

Rulin Shao, Zhouxing Shi, Jinfeng Yi, Pin-Yu Chen, and Cho-Jui Hsieh. On the adversarial robustness of vision transformers. *arXiv preprint arXiv:2103.15670*, 2021.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2556–2565, 2018.

Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural networks*, 32:323–332, 2012.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

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

Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. Visual autoregressive modeling: Scalable image generation via next-scale prediction. *Advances in neural information processing systems*, 37:84839–84865, 2024.

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

Zhibo Wang, Hongshan Yang, Yunhe Feng, Peng Sun, Hengchang Guo, Zhifei Zhang, and Kui Ren. Towards transferable targeted adversarial examples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 20534–20543, 2023.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Zhipeng Wei, Jingjing Chen, Micah Goldblum, Zuxuan Wu, Tom Goldstein, and Yu-Gang Jiang. Towards transferable adversarial attacks on vision transformers. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 2668–2676, 2022.

Zhipeng Wei, Jingjing Chen, Zuxuan Wu, and Yu-Gang Jiang. Enhancing the self-universality for transferable targeted attacks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12281–12290, 2023.

Juanjuan Weng, Zhiming Luo, Dazhen Lin, and Shaozi Li. Learning transferable targeted universal adversarial perturbations by sequential meta-learning. *Computers & Security*, 137:103584, 2024.

Han Wu, Guanyan Ou, Weibin Wu, and Zibin Zheng. Improving transferable targeted adversarial attacks with model self-enhancement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24615–24624, 2024.

Chaowei Xiao, Bo Li, Jun-Yan Zhu, Warren He, Mingyan Liu, and Dawn Song. Generating adversarial examples with adversarial networks. *arXiv preprint arXiv:1801.02610*, 2018.

Cihang Xie, Zhishuai Zhang, Yuyin Zhou, Song Bai, Jianyu Wang, Zhou Ren, and Alan L Yuille. Improving transferability of adversarial examples with input diversity. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2730–2739, 2019.

Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. *arXiv preprint arXiv:2408.12528*, 2024.

Zonghao Ying, Aishan Liu, Siyuan Liang, Lei Huang, Jinyang Guo, Wenbo Zhou, Xianglong Liu, and Dacheng Tao. Safebench: A safety evaluation framework for multimodal large language models. *International Journal of Computer Vision*, 134(1):18, 2026.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the association for computational linguistics*, 2:67–78, 2014.

Chaoning Zhang, Dongshen Han, Yu Qiao, Jung Uk Kim, Sung-Ho Bae, Seungkyu Lee, and Choong Seon Hong. Faster segment anything: Towards lightweight sam for mobile applications. *arXiv preprint arXiv:2306.14289*, 2023a.

Chenshuang Zhang, Chaoning Zhang, Taegoo Kang, Donghun Kim, Sung-Ho Bae, and In So Kweon. Attack-sam: Towards evaluating adversarial robustness of segment anything model. *arXiv preprint arXiv:2305.00866*, 1(3):5, 2023b.

Jiaming Zhang, Qi Yi, and Jitao Sang. Towards adversarial attack on vision-language pre-training models. In *Proceedings of the 30th ACM International Conference on Multimedia*, pp. 5005–5013, 2022.

Jiaming Zhang, Junhong Ye, Xingjun Ma, Yige Li, Yunfan Yang, Yunhao Chen, Jitao Sang, and Dit-Yan Yeung. Anyattack: Towards large-scale self-supervised adversarial attacks on vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 19900–19909, 2025.

Peng-Fei Zhang, Zi Huang, and Guangdong Bai. Universal adversarial perturbations for vision-language pre-trained models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 862–871, 2024.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 586–595. IEEE, 2018.

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

Sheng Zheng, Chaoning Zhang, and Xinhong Hao. Black-box targeted adversarial attack on segment anything (sam). *IEEE Transactions on Multimedia*, 27:1901–1913, 2024.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 633–641, 2017.

Ziqi Zhou, Shengshan Hu, Minghui Li, Hangtao Zhang, Yechao Zhang, and Hai Jin. Advclip: Downstream-agnostic adversarial examples in multimodal contrastive learning. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 6311–6320, 2023.

Ziqi Zhou, Yufei Song, Minghui Li, Shengshan Hu, Xianlong Wang, Leo Yu Zhang, Dezhong Yao, and Hai Jin. Darksam: Fooling segment anything model to segment nothing. *Advances in Neural Information Processing Systems*, 37:49859–49880, 2024.

Ziqi Zhou, Yifan Hu, Yufei Song, Zijing Li, Shengshan Hu, Leo Yu Zhang, Dezhong Yao, Long Zheng, and Hai Jin. Vanish into thin air: Cross-prompt universal adversarial attacks for sam2. *arXiv preprint arXiv:2510.24195*, 2025.

A Impact Statements

DarkLLM has several potential positive broader impacts. **1)** DarkLLM presents an instruction-guided framework for generating adversarial examples using contextualized representations from large language models. This approach offers a concrete example of connecting language representations with visual perturbation generation and may inform related research in other modalities. **2)** DarkLLM provides a unified framework and evaluation across targeted, untargeted, and multi-model attacks, supporting the study of language-driven robustness assessment across different attack objectives. **3)** The mapping from language instructions to pixel-level perturbations also provides a basis for studying the interaction between visual and textual adversarial inputs.

B Ethical Considerations

We acknowledge the dual-use nature of DarkLLM, which, while designed for safety evaluation, could potentially be repurposed for malicious intent. To ensure responsible usage, we emphasize that DarkLLM is intended strictly for two legitimate purposes: (i) Red-Teaming: assisting developers in stress-testing the robustness of foundation models to identify and mitigate vulnerabilities; and (ii) Privacy Preservation: empowering individuals to shield their personal data from unauthorized analysis by ubiquitous vision systems. We strictly prohibit the use of this framework for any illegal activities, including disrupting lawful services or evading legitimate content moderation. By sharing our methodology, we aim to equip the research community with the insights necessary to build more resilient defenses against the emerging generation of language-driven adversarial threats, prioritizing the advancement of AI security over offensive proliferation.

Table 5: A summary of our considered evaluation tasks, datasets, models, and performance metrics.

Evaluation Task	Attack Objective	Evaluation Model	Evaluation Dataset	Evaluation Metric
Zero-Shot Classification	Untargeted	CLIP-RN101, CLIP-ViT-B-16 CLIP-ViT-B-32, CLIP-ViT-L-14	CIFAR10, CIFAR100, Food GTSRB, ImageNet, Cars, STL	ASR
Image-Text Retrieval	Untargeted	CLIP-RN101, CLIP-ViT-B-16 CLIP-ViT-B-32, CLIP-ViT-L-14	MS-COCO	ASR
Image Captioning	Untargeted	LLaVA-7B	MS-COCO, Flickr30k	ASR
Image Captioning	Targeted	Qwen3-VL-8B, GPT-4o GPT-4.1, GPT-5-mini, Gemini-2.5-Flash Gemini-3.0-Flash	NIPS 2017 Competition	ASR
Image Segmentation	Untargeted	SAM-B, SAM-H HQ-SAM-H, SAM2-Hiera-Large	COCO, ADE, Cityscapes	mIoU, ASR

C Experiments

C.1 Detailed Experimental Setting

More Implementation Details. We implement DarkLLM using the XTuner codebase (Contributors, 2023) on 8 NVIDIA H200 GPUs. The LLM controller is initialized from InternVL2.5 (Chen et al., 2024) and fine-tuned via LoRA (Hu et al., 2022) ($r = 128, \alpha = 256$). Optimization is performed using AdamW with $\beta = (0.9, 0.999)$ and a weight decay of 0.05. We utilize a compound learning rate scheduler: a linear warmup phase for the initial 5% of iterations, followed by a cosine annealing decay for the remainder.

Our training follows a decoupled progressive strategy designed to mitigate the convergence imbalance between different attack objectives. As detailed in Table 6, we assign unique control tokens (e.g., `[VLM_TGT]`, `[SEG_ADV]`) to route instructions to their dedicated noise generators, effectively isolating gradient updates for incompatible tasks. *Stage 1 (Targeted Instruction Tuning):* We first jointly train the LLM controller and the targeted generator using ImageNet (Deng et al., 2009) (55 epochs) and CC3M (Sharma et al., 2018) (10 epochs), with a per-GPU batch size of 128. To construct the training QA pairs, we use images from the COCO 2017 Caption training (Lin et al., 2014) set as the target images, and leverage their corresponding ground-truth captions to populate the {target_caption} placeholder in our instructional template (see Table 6). This stage focuses exclusively on mastering the semantic parsing of complex targeted instructions. *Stage 2 (Unified*

Attack Training): We then initialize the system with Stage-1 weights. In this phase, we freeze the targeted generator to preserve its learned capabilities and introduce the untargeted and multi-objective generators, which are optimized alongside the LLM controller. This stage involves joint training on all attack types for 8 epochs with a batch size of 64, enabling the LLM to learn unified control across diverse adversarial tasks.

Leveraging our modular design, we flexibly assign perturbation budgets (ϵ) tailored to each task's standard benchmarks. Specifically, we set $\epsilon = 16/255$ for targeted MLLM attacks following the protocol of AnyAttack (Zhang et al., 2025). For SAM attacks, we also use $\epsilon = 10/255$, and all baselines are evaluated under identical settings to ensure fair comparison. For untargeted VLM attacks, we use $\epsilon = 12/255$ following X-Transfer (Huang et al., 2025). Finally, for multi-model attacks, we adopt $\epsilon = 12/255$ to ensure optimal attack performance. The unified training objective balances the active surrogate losses:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{t}}\mathcal{L}_{\text{txt}} + \lambda_{\text{seg}}\mathcal{L}_{\text{seg}} + \lambda_{\text{ta}}\mathcal{L}_{\text{target}} + \lambda_{\text{un}}\mathcal{L}_{\text{untarget}}, \tag{9}$$

where $\lambda_{\text{t}} = 1.0$, $\lambda_{\text{seg}} = 10.0$, $\lambda_{\text{ta}} = 10.0$, and $\lambda_{\text{un}} = 8.0$, respectively, to balance different attack objectives.

Table 6: Example of our training QA template. {target_caption} is replaced by the actual caption during training.

```
Human : Make the vision-language model identify this image as [{target_caption}]. <STOP>
DarkLLM: Sure, the generated perturbation can be represented as [VLM_TGT]. <STOP>
Human : Make the vision-language model fail to recognize this image. <STOP>
DarkLLM: Sure, the generated perturbation can be represented as [VLM_ADV]. <STOP>
Human : Make the segmentation model fail to segment this image. <STOP>
DarkLLM: Sure, the generated perturbation can be represented as [SEG_ADV]. <STOP>
Human : Make the segmentation model fail to segment this image and the vision-language model fail
to recognize this image simultaneously. <STOP>
DarkLLM: Sure, the generated perturbation can be represented as [MULTI_ADV]. <STOP>
```

Evaluation Baselines. We benchmark DarkLLM against state-of-the-art baselines across three distinct settings. **(1)** For untargeted attacks against VLMs, we compare our approach with state-of-the-art UAP methods specifically tailored for CLIP encoders, including X-Transfer (Huang et al., 2025), C-PGC (Fang et al., 2025), and ETU (Zhang et al., 2024). We also include foundational UAP methods designed for image classifiers, such as GD-UAP (Mopuri et al., 2018), TRM-UAP (Liu et al., 2023c), and Meta-UAP (Weng et al., 2024). The details of each baseline are summarized in Table 7. **(2)** For targeted attacks against MLLMs, we conduct a systematic comparison against two leading categories of methods: the sample-wise approaches AttackVLM (Zhao et al., 2023) and M-Attack (Li et al., 2025b), and the universal attack method AnyAttack (Zhang et al., 2025). The details of each baseline are summarized in Table 8. **(3)** For untargeted attacks against SAM, our comparison includes leading sample-wise methods Attack-SAM (Zhang et al., 2023b), PATA (Zheng et al., 2024), and UAD (Lu et al., 2024), as well as the universal methods DarkSAM (Zhou et al., 2024) and UAP-SAM2 (Zhou et al., 2025). The details of each baseline are summarized in Table 9.

Evaluation Datasets. We provide a comprehensive summary of our experimental setup, including evaluation tasks, models, datasets, and metrics, in Table 5. Our evaluation spans a wide range of standard benchmarks tailored to each adversarial task. **(1)** For untargeted attacks on VLMs, our experiments cover three task families: *(i) Zero-shot Classification*, for which we use seven datasets (CIFAR-10, CIFAR-100 (Krizhevsky et al., 2009), Food-101 (Bossard et al., 2014), GTSRB (Stallkamp et al., 2012), ImageNet (Deng et al., 2009), Stanford Cars (Krause et al., 2013), and STL-10 (Coates et al., 2011)); *(ii) Image-Text Retrieval*, using the COCO-Karpathy split (Karpathy & Fei-Fei, 2015); and *(iii) Image Captioning*, evaluated on MS-COCO Caption (Chen et al., 2015) and Flickr30k (Young et al., 2014). **(2)** For targeted attacks on MLLMs, we follow prior work (Dong et al., 2023; Li et al., 2025b; Jia et al., 2025) by using a subset of 100 images from the NIPS 2017 Adversarial Attacks and Defenses Competition dataset[1] (K et al., 2017). **(3)** For untargeted attacks on

[1] https://nips.cc/Conferences/2017/CompetitionTrack

Table 7: Details of baseline attacks on vision–language models.

Method	Variant	Note
GD-UAP	Seg	Generated with segmentation model (ResNet152 backbone) as the surrogate.
	CLS	Generated with ResNet152 classifier as the surrogate.
TRM-UAP	GoogleNet	Generated with GoogleNet classifier trained on ImageNet.
	RN152	Generated with RN152 classifier trained on ImageNet.
Meta-UAP	Ensemble	Generated with an ensemble of DenseNet121, VGG16, and ResNet50 classifiers trained on ImageNet.
	Ensemble-Meta	Same as above, with meta-learning strategy.
C-GPC	RN101-COCO	Generated with CLIP ResNet101 released by OpenAI on the MSCOCO dataset.
	ViT-B/16-COCO	Generated with CLIP ViT-B/16 released by OpenAI on the MSCOCO dataset.
ETU	RN101-Flicker	Generated with CLIP ResNet101 released by OpenAI on the Flicker dataset.
	ViT-B/16-Flicker	Generated with CLIP ViT-B/16 released by OpenAI on the Flicker dataset.
X-Transfer	Vanilla	Generated with multiple CLIP encoders without surrogate scaling.
	Base	Generated with 16 CLIP encoders under the surrogate scaling strategy.

Table 8: Details of baseline attacks on multimodal large language models.

Method	Variant	Note
AttackVLM	B/16	Generated with CLIP ViT-B/16 released by OpenAI under a perturbation budget of $\epsilon = 16/255$.
	L/14	Generated with CLIP-ViT-L-14-laion2B-s32B-b82K under a perturbation budget of $\epsilon = 16/255$.
	Ensemble	Generated by jointly optimizing against CLIP ViT-B/16 and ViT-L/14 under $\epsilon = 16/255$.
M-Attack	B/16	Generated with CLIP ViT-B/16 released by OpenAI under a perturbation budget of $\epsilon = 16/255$.
	L/14	Generated with CLIP-ViT-L-14-laion2B-s32B-b82K under a perturbation budget of $\epsilon = 16/255$.
	Ensemble	Generated by jointly optimizing against CLIP ViT-B/16 and ViT-L/14 under $\epsilon = 16/255$.
AnyAttack	coco-bi	Generated by jointly optimizing against CLIP ViT-B/32 (OpenAI), ViT-B/16 (ImageNet), and ViT-L/14 EVA on MSCOCO under $\epsilon = 16/255$ using a contrastive loss.
	coco-cos	Generated by jointly optimizing against CLIP ViT-B/32 (OpenAI), ViT-B/16 (ImageNet), and ViT-L/14 EVA on MSCOCO under $\epsilon = 16/255$ using a cosine similarity loss.
	flickr-bi	Generated by jointly optimizing against CLIP ViT-B/32 (OpenAI), ViT-B/16 (ImageNet), and ViT-L/14 EVA on Flickr dataset under $\epsilon = 16/255$ using a contrastive loss.
	flickr-cos	Generated by jointly optimizing against CLIP ViT-B/32 (OpenAI), ViT-B/16 (ImageNet), and ViT-L/14 EVA on Flickr dataset under $\epsilon = 16/255$ using a cosine similarity loss.

SAM, we employ three widely-used segmentation benchmarks in the domain: ADE20K-val (Zhou et al., 2017), COCO2017-val (Lin et al., 2014), and Cityscapes-val (Cordts et al., 2016).

Evaluation Models. We benchmark DarkLLM's transferability against a diverse set of foundation models. **(1)** For untargeted VLM attacks, performance on classification and retrieval is averaged across four black-box CLIP models as shown in Table 10, while image captioning is evaluated on LLaVA-1.5-7B (Liu et al., 2023a). **(2)** For targeted MLLM attacks, our evaluation targets six frontier commercial models: Qwen3-VL, GPT-4o, GPT-4.1, GPT-5-mini, Gemini-2.5-Flash, and Gemini-3.0-Flash. **(3)** For untargeted SAM attacks, we assess cross-model transferability across the SAM family, including SAM-Base, SAM-Huge (Kirillov et al., 2023), HQ-SAM-Huge (Ke et al., 2023), and SAM2-Hiera-Large (Ravi et al., 2024).

Evaluation Metrics. We adopt task-specific metrics to comprehensively evaluate performance. **(1)** For untargeted VLM attacks, we report a normalized Attack Success Rate (ASR) defined as $(s_{\text{clean}} - s_{\text{adv}})/s_{\text{clean}}$, where s is the primary metric for each task (i.e., accuracy for classification, CIDEr (Vedantam et al., 2015) for captioning). s_{clean} and s_{adv} are the scores on clean and adversarial images, respectively. **(2)** For targeted MLLM attacks, we follow the LLM-as-a-judge protocol following (Li et al., 2025b; Jia et al., 2025). Specifically, we prompt the target MLLM itself to generate captions for both adversarial and target images, then assess their similarity via GPTScore. An attack is deemed successful if the similarity exceeds 0.3. We report both ASR and the average similarity score (AvgSim). Detailed evaluation prompts are provided in Table 11. **(3)** For untargeted SAM attacks, we report both ASR and the standard mean Intersection over Union (mIoU) to measure segmentation quality degradation.

Surrogate Models. DarkLLM uses a diverse suite of surrogate models to optimize perturbations for different attack objectives. **(1)** Specifically, for untargeted attacks on VLMs, we employ a surrogate scaling strategy utilizing an ensemble of 16 distinct CLIP variants to enhance perturbation transferability following (Huang et al., 2025), with full details provided in Table 12. **(2)** For targeted attacks on MLLMs, our optimization relies on two powerful, publicly available CLIP models: the

Table 9: Details of baseline attacks on segment anything models.

Method	Variant	Note
Attack-SAM	—	Generated with SAM-Base released by Meta under a perturbation budget of $\epsilon = 10/255$.
UAD	—	Generated with SAM-Base released by Meta under a perturbation budget of $\epsilon = 10/255$.
PATA	—	Generated with SAM-Base released by Meta under a perturbation budget of $\epsilon = 10/255$.
PATA++	—	Generated with SAM-Base released by Meta under a perturbation budget of $\epsilon = 10/255$.
DarkSAM	DarkSAM-pt	Generated with SAM-Base on the SA-1B dataset under a perturbation budget of $\epsilon = 10/255$ using point prompts.
	DarkSAM-bx	Generated with SAM-Base on the SA-1B dataset under a perturbation budget of $\epsilon = 10/255$ using box prompts.
UAP-SAM2	—	Generated with SAM2-Small on the DAVIS dataset under a perturbation budget of $\epsilon = 16/255$ using point prompts.

Table 10: Encoders used in our zero-shot classification and image–text retrieval evaluations. For each encoder, we report its architecture, pre-training dataset, and the corresponding OpenCLIP identifier.

	Architecture	Pre-training Dataset	OpenCLIP Identifier
1	RN101	WebImageText	(RN101, openai)
2	ViT-B/16	WebImageText	(ViT-B-16, openai)
3	ViT-B/32	WebImageText	(ViT-B-32, openai)
4	ViT-L/14	WebImageText	(ViT-L-14, openai)

Table 11: Details of our evaluation prompt template. During evaluation, text1 and text2 are replaced with the clean prediction and the adversarial prediction, respectively.

Rate the semantic similarity between the following two texts on a scale from 0 to 1.
Criteria for similarity measurement:
1. **Main Subject Consistency:** If both descriptions refer to the same key subject or object (e.g., a person, food, an event), they should receive a higher similarity score.
2. **Relevant Description**: If the descriptions are related to the same context or topic, they should also contribute to a higher similarity score.
3. **Ignore Fine-Grained Details:** Do not penalize differences in **phrasing, sentence structure, or minor variations in detail**. Focus on **whether both descriptions fundamentally describe the same thing.**
4. **Partial Matches:** If one description contains extra information but does not contradict the other, they should still have a high similarity score.
5. **Similarity Score Range:**
- **1.0**: Nearly identical in meaning.
– **0.8-0.9**: Same subject, with highly related descriptions.
– **0.7-0.8**: Same subject, core meaning aligned, even if some details differ.
– **0.5-0.7**: Same subject but different perspectives or missing details.
– **0.3-0.5**: Related but not highly similar (same general theme but different descriptions).
– **0.0-0.2**: Completely different subjects or unrelated meanings.
Text 1: {text1}
Text 2: {text2}
Output only a single number between 0 and 1. Do not include any explanation or additional text.

OpenAI ViT-B/16 and CLIP-ViT-L-14-laion2B-s32B-b82K. **(3)** Finally, for untargeted attacks against the SAM family, we use SAM-Base and the lightweight MobileSAM as our primary surrogates.

C.2 Additional Evaluation Results

Comprehensive Evaluations on the SAM. We conduct a comprehensive evaluation to validate DarkLLM's effectiveness across the broader SAM family. Tables 13, 14, and 15 summarize the performance against various backbones of SAM (Kirillov et al., 2023), HQ-SAM (Ke et al., 2023), and SAM2 (Ravi et al., 2024), respectively. Three key findings emerge from these results: (1) Across most evaluated settings, DarkLLM outperforms UAD on different models, datasets, and prompt types, demonstrating the superior adversarial efficacy and generalization of our approach. (2) While UAD remains competitive in several Cityscapes settings, particularly with point prompts, DarkLLM

Table 12: Surrogate encoders used for training, with architectures, pre-training datasets, and OpenCLIP identifiers.

	Architecture	Pre-training Dataset	OpenCLIP Identifier
1	ConvNext Base	LAION400M	(convnext_base, laion400m_s13b_b51k)
2	ConvNext Base-W	LAION2B	(convnext_base_w, laion2b_s13b_b82k)
3	ConvNext Base-W	LAION2B	(convnext_base_w, laion2b_s13b_b82k_augreg)
4	ConvNext Large-D	LAION2B	(convnext_large_d, laion2b_s26b_b102k_augreg)
5	ViT-B/16	DFN2B	(ViT-B-16, dfn2b)
6	ViT-B/16	DataComp	(ViT-B-16, datacomp_xl_s13b_b90k)
7	ViT-B/16	LAION2B	(ViT-B-16, laion2b_s34b_b88k)
8	ViT-B/32	LAION2B	(ViT-B-32, laion2b_s34b_b79k)
9	ViT-B/32	DataComp	(ViT-B-32, datacomp_xl_s13b_b90k)
10	ViT-B/32	LAION400M	(ViT-B-32-quickgelu, laion400m_e32)
11	ViT-L/14	LAION400M	(ViT-L-14, laion400m_e32)
12	ViT-L/14	CommonPool	(ViT-L-14, commonpool_xl_s13b_b90k)
13	ViT-L/14	MetaCLIP	(ViT-L-14-quickgelu, metaclip_fullcc)
14	ViT-L/14	LAION2B	(ViT-L-14, laion2b_s34b_b88k)
15	EVA02-L/14	Merged2B	(EVA02-L-14, merged2b_s4b_b131k)
16	ViT-L/14	DataComp	(ViT-L-14, datacomp_xl_s13b_b90k)

achieves lower mIoU in most other evaluated settings, supporting its effectiveness across different datasets and prompt types. (3) Box prompts yield higher clean and attacked mIoU than point prompts in the reported results. Nevertheless, DarkLLM reduces segmentation performance relative to the corresponding clean baseline under both prompt types.

Table 13: The mIoU (%) results for promptable image segmentation task across different SAM models.

Setting		SAM-Base			SAM-Large			SAM-Huge		
Prompt	**Method**	**ADE**	**COCO**	**CITY**	**ADE**	**COCO**	**CITY**	**ADE**	**COCO**	**CITY**
Point	Clean	50.3	44.9	38.0	53.3	48.1	40.0	53.6	48.5	39.7
	UAD (Lu et al., 2024)	42.7	39.4	8.6	46.9	44.2	13.1	48.0	44.6	16.0
	DarkLLM (Ours)	**2.9**	**3.5**	**0.4**	**18.8**	**22.0**	**18.6**	**21.5**	**24.3**	**19.2**
Box	Clean	72.8	71.4	61.2	73.9	73.0	61.6	74.5	73.2	61.5
	UAD (Lu et al., 2024)	68.3	68.0	39.7	70.3	70.5	45.5	71.3	71.1	47.9
	DarkLLM (Ours)	**27.4**	**30.5**	**30.2**	**48.4**	**55.5**	**44.9**	**46.9**	**55.1**	**43.4**

Table 14: The mIoU (%) results for promptable image segmentation task across different HQ-SAM models.

Setting		HQ-SAM-Tiny			HQ-SAM-Base			HQ-SAM-Large			HQ-SAM-Huge		
Prompt	**Method**	**ADE**	**COCO**	**CITY**	**ADE**	**COCO**	**CITY**	**ADE**	**COCO**	**CITY**	**ADE**	**COCO**	**CITY**
Point	Clean	47.8	42.9	30.4	46.0	39.7	29.8	49.9	44.3	32.0	52.4	47.3	32.0
	UAD (Lu et al., 2024)	42.0	40.2	15.1	38.5	34.9	6.1	43.4	39.8	12.9	46.5	43.6	14.4
	DarkLLM (Ours)	**15.0**	**15.5**	**16.4**	**4.5**	**5.2**	**2.1**	**17.0**	**19.3**	**14.6**	**18.9**	**21.9**	**14.2**
Box	Clean	71.4	69.8	57.6	66.3	60.6	46.6	70.1	66.8	50.2	69.4	65.2	50.2
	UAD (Lu et al., 2024)	68.3	67.2	50.2	61.1	58.4	18.1	66.1	64.4	37.9	65.8	63.7	33.5
	DarkLLM (Ours)	**34.3**	**37.7**	**38.1**	**32.0**	**37.5**	**28.3**	**45.4**	**49.1**	**35.8**	**40.9**	**47.1**	**32.3**

Table 15: The mIoU (%) results for promptable image segmentation task across different SAM2 models.

Setting		SAM2-Hiera-Tiny			SAM2-Hiera-Small			SAM2-Hiera-Base			SAM2-Hiera-Large		
Prompt	**Method**	**ADE**	**COCO**	**CITY**	**ADE**	**COCO**	**CITY**	**ADE**	**COCO**	**CITY**	**ADE**	**COCO**	**CITY**
Point	Clean	52.8	49.6	37.3	53.5	49.5	23.1	55.4	51.0	40.6	57.2	55.2	41.7
	UAD (Lu et al., 2024)	48.1	48.2	25.0	48.9	48.3	14.2	50.5	50.0	23.2	52.6	52.7	26.0
	DarkLLM (Ours)	**19.7**	**21.4**	**16.4**	**18.9**	**21.1**	**16.7**	**16.4**	**19.8**	**15.3**	**20.2**	**23.4**	**15.1**
Box	Clean	73.8	72.9	61.5	74.0	73.1	62.1	74.6	74.0	63.2	74.9	73.9	62.7
	UAD (Lu et al., 2024)	71.8	71.4	53.9	72.1	71.6	53.2	72.5	72.5	52.5	73.0	72.7	53.4
	DarkLLM (Ours)	**55.9**	**58.4**	**45.9**	**51.0**	**55.3**	**43.0**	**50.1**	**55.4**	**43.2**	**49.5**	**55.7**	**38.8**

Generalization Evaluation We evaluate the generalization of DarkLLM along two dimensions: target-caption content and instruction-template wording. For target-caption content, we use Qwen3-VL 8B to generate new captions for the target images used in the NIPS 2017 evaluation. As illustrated in Table 16, these captions provide more detailed and context-rich descriptions than the original reference captions. We incorporate them into the instruction templates in Table 6 to generate targeted visual perturbations. As reported in Table 17, DarkLLM achieves ASRs of 57.0–68.0% across five commercial MLLMs, supporting its ability to accommodate alternative target descriptions. For instruction-template wording, we evaluate five previously unseen targeted-attack templates without retraining, as listed alongside the original template in Table 18. Table 19 compares the original template with the mean and standard deviation across the five unseen templates. The mean ASR decreases by at most 1.9 percentage points, with standard deviations of 1.2–2.6 percentage points, indicating limited sensitivity to the tested wording variations. Together, these results support generalization to alternative target descriptions and instruction phrasings within the supported targeted-attack task. Our transfer results concern models within the evaluated task settings and do not establish retraining-free transfer to arbitrary new model families or previously unsupported tasks.

Table 16: Comparison between original reference captions and captions generated by Qwen3-VL 8B.

Original Captions:
(1) A dog chewing on a object held in a hand.
(2) A group of young people getting ready to go ski.
(3) A man that is laying down underneath a cat.
Qwen3-VL Generated Captions:
(1) A black and white dog rests its head on a person's lap while being gently petted with a red toy.
(2) A group of skiers with large backpacks and poles stand on a snowy slope, adjusting gear before descent.
(3) A man sleeps peacefully under white covers, while two cats rest nearby on pillows and the bedspread.

Table 17: Targeted ASR (%) and AvgSim results for image captioning tasks across different MLLMs.

Method	Base Model	GPT-4o		GPT-4.1		GPT-5-mini		Gemini-2.5-Flash		Gemini-3-Flash	
		ASR	AvgSim	ASR	AvgSim	ASR	AvgSim	ASR	AvgSim	ASR	AvgSim
DarkLLM (Ours)	InternVL-2B	63.0	0.26	68.0	0.27	61.0	0.24	60.0	0.24	57.0	0.24

Table 18: Original and unseen instruction templates for targeted attacks. [Target] denotes the target caption. All unseen templates are evaluated without retraining.

Original Template:
Make the vision-language model identify this image as [Target].

Unseen Templates:
(1) Cause the image-text model to interpret this image as [Target].
(2) Generate a universal perturbation that associates the image with the target concept [Target].
(3) Steer the vision-language system toward [Target] instead of the image's true semantic content.
(4) The perturbed image should be recognized semantically as [Target] by the vision-language model.
(5) Redirect the image representation so that it aligns with the text [Target].

Expanded Evaluation and Threshold Sensitivity To assess performance on a larger evaluation set and examine sensitivity to the success criterion, we evaluate 1,000 clean images from the NIPS 2017 dataset, each paired with a randomly sampled MSCOCO validation image as the target. We report both the original ASR@0.3 criterion and the stricter ASR@0.5 criterion, using GPT-4o as the semantic judge with temperature set to 0 throughout. As shown in Table 20, DarkLLM achieves an average ASR of 55.5% under ASR@0.3, outperforming AnyAttack by 24.8 percentage points across the three commercial MLLMs. Increasing the semantic-similarity threshold to 0.5 substantially

Table 19: Targeted ASR (%) under instruction-template variations. Original uses "Make the vision-language model identify this image as [Target]." Unseen reports mean ± standard deviation across five unseen templates, without retraining.

Instruction Setting	GPT-4o	GPT-5-mini	Gemini-2.5-Flash
Original	67.7	58.6	63.6
Unseen (Avg@5)	67.2 ± 2.2	56.7 ± 1.2	61.8 ± 2.6

reduces the absolute ASR of both methods, as successful attacks must exhibit stronger semantic alignment with the target description. Nevertheless, DarkLLM consistently outperforms AnyAttack on all three MLLMs, achieving an average ASR of 11.1% compared with 5.1%. These results show that DarkLLM's comparative advantage persists on the expanded evaluation set and under both evaluated success thresholds.

Table 20: Targeted ASR (%) on 1,000 NIPS 2017 images under two semantic-similarity thresholds. GPT-4o is used as the semantic judge with temperature set to 0.

Threshold	Method	GPT-4o	GPT-5-mini	Gemini-2.5-Flash	Average
ASR@0.3	AnyAttack	33.9	30.2	27.9	30.7
	DarkLLM (Ours)	**61.1**	**48.1**	**57.2**	**55.5**
ASR@0.5	AnyAttack	5.7	6.5	3.2	5.1
	DarkLLM (Ours)	**11.7**	**11.4**	**10.1**	**11.1**

Stability across Repeated API Evaluations To assess variability across repeated commercial API evaluations, we conduct three independent evaluation runs using the same dataset, prompts, and perturbations. Temperature is set to 0 for all evaluated commercial MLLMs and the GPT-4o semantic judge. Table 21 reports the targeted ASR for each run, together with the mean and sample standard deviation. Across the four evaluated MLLMs, the standard deviation ranges from 1.8 to 3.7 percentage points, indicating limited run-to-run variation under this fixed evaluation setup. The perturbations are held unchanged across runs, so this experiment measures variability in repeated API evaluations rather than variability from perturbation generation. These results complement the single-run evaluation by characterizing the observed variability of DarkLLM's targeted ASR under repeated calls.

Table 21: Targeted ASR (%) across three repeated API evaluation runs with fixed data, prompts, and perturbations. The final column reports the mean and sample standard deviation across runs.

Model	Run 1	Run 2	Run 3	Mean ± Std.
GPT-4o	68.7	63.6	70.7	67.7 ± 3.7
GPT-5-mini	59.6	63.6	59.6	60.9 ± 2.3
Gemini-2.5-Flash	65.7	65.7	62.6	64.7 ± 1.8
Gemini-3-Flash	58.6	59.6	55.6	57.9 ± 2.1

C.3 Ablation and Sensitivity Analysis

Comparison with a Standalone Generator We compare DarkLLM with a standalone perturbation generator trained solely on the SAM untargeted segmentation objective without the LLM controller. Both approaches are evaluated on ADE20K-val using point prompts across three SAM variants. As shown in Table 22, DarkLLM achieves comparable segmentation attack performance, with an average mIoU of 14.9% compared with 14.3% for the standalone generator, where lower values indicate stronger attacks. DarkLLM obtains slightly lower mIoU on SAM-Base and SAM-Huge, while the standalone generator performs better on SAM2-Hiera-Large. These results suggest that the unified

framework retains attack effectiveness close to standalone segmentation training while supporting multiple attack objectives through a shared language interface.

Table 22: Comparison with a standalone segmentation generator on ADE20K-val using point prompts. Results are mIoU (%), where lower values indicate stronger attacks.

Method	SAM-Base	SAM-Huge	SAM2-Hiera-Large	Average
Clean	50.3	53.6	57.2	53.7
Standalone Generator	4.0	21.8	**17.2**	**14.3**
DarkLLM (Ours)	**2.9**	**21.5**	20.2	14.9

Sensitivity to Loss Weights We investigate the sensitivity of DarkLLM to the loss weights in Eq. 9. The default values are $\lambda_{\text{seg}} = 10$, $\lambda_{\text{ta}} = 10$, and $\lambda_{\text{un}} = 8$. These values were empirically selected to keep the weighted loss terms at comparable magnitudes during early training. We independently increase or decrease each weight by 50%, while keeping the remaining weights unchanged and $\lambda_{\text{t}} = 1$. All seven configurations are trained on ImageNet for two epochs with otherwise identical settings. We evaluate ADE20K segmentation with point prompts using three SAM variants, ImageNet zero-shot classification using four CLIP backbones, and targeted image captioning on NIPS 2017 using three commercial MLLMs. Table 23 reports task-specific ASR and its average across models within each task. For segmentation, ASR measures the relative reduction in mIoU with respect to clean performance. Three observations emerge: (1) Across the seven configurations, average ASR ranges from 64.8% to 67.7% for segmentation, 57.1% to 60.5% for classification, and 57.6% to 62.0% for targeted captioning. The largest decrease from the default task average is 2.7 percentage points, indicating that the tested $\pm 50\%$ weight changes do not sharply degrade task-average performance under the evaluated training schedule. (2) The loss weights exhibit intuitive task-specific effects. Increasing λ_{seg} improves average segmentation ASR from 65.4% to 67.7%, while increasing λ_{ta} improves targeted captioning from 60.3% to 62.0%. Increasing λ_{un} also improves classification from 57.1% to 58.5%. (3) Overall, DarkLLM does not rely on fragile hyperparameter tuning. Instead, the loss weights provide a simple and interpretable mechanism for adjusting the trade-offs among different attack objectives while maintaining stable performance across tasks.

Table 23: Sensitivity to loss weights across three tasks (ASR %, higher is better). The default weights are $(\lambda_{\text{t}}, \lambda_{\text{seg}}, \lambda_{\text{ta}}, \lambda_{\text{un}}) = (1, 10, 10, 8)$. Each non-default row changes only the indicated weight by $\pm 50\%$. All configurations are trained for two epochs on ImageNet.

Weight setting	Image segmentation				Zero-shot classification					Targeted image captioning			
	SAM-Base	SAM-Huge	SAM2	Average	RN101	ViT-B/16	ViT-B/32	ViT-L/14	Average	GPT-4o	GPT-5-mini	Gemini 2.5	Average
Default	66.8	59.9	69.6	65.4	69.5	57.6	55.2	46.0	57.1	66.7	53.5	60.6	60.3
λ_{seg} +50%	67.8	64.2	71.2	67.7	72.4	61.3	57.6	50.6	60.5	62.6	55.5	56.5	58.2
λ_{seg} −50%	66.4	61.0	70.5	66.0	69.6	59.2	56.9	54.4	60.0	64.6	60.6	56.5	60.6
λ_{ta} +50%	67.2	63.2	70.5	67.0	72.0	59.3	56.8	49.2	59.3	64.7	56.6	64.7	62.0
λ_{ta} −50%	63.8	63.8	69.6	65.7	70.7	57.1	55.6	46.1	57.4	61.6	54.5	60.6	58.9
λ_{un} +50%	67.4	60.8	68.9	65.7	70.7	58.2	56.0	49.0	58.5	62.6	55.5	59.6	59.2
λ_{un} −50%	70.4	56.9	67.0	64.8	69.7	57.6	56.0	48.4	57.9	61.6	51.5	59.6	57.6

C.4 Defense and Perceptual Quality Evaluation

Complete Defense Evaluation. We provide the complete defense results corresponding to Figure 6. Table 24 compares DarkLLM with X-Transfer (Huang et al., 2025) on ImageNet zero-shot classification using four CLIP backbones, including ViT-B/16 and ViT-B/32 in addition to the two backbones shown in the main paper. Table 25 reports comparisons with UAD (Lu et al., 2024) on ADE20K-val using SAM-Base and SAM-Huge. We evaluate one-pass and two-pass Feature Distillation (Liu et al., 2019), Randomized Smoothing (Cohen et al., 2019) with $\sigma = 0.05$ and 32 samples for CLIP and 9 for SAM, and JPEG compression (Guo et al., 2017) at quality factors 90 and 75. The clean SAM results are references without attacks or defenses. Across the five defense settings, DarkLLM achieves higher ASR than X-Transfer in 17 of the 20 classification comparisons and lower mIoU than UAD in all 10 segmentation comparisons.

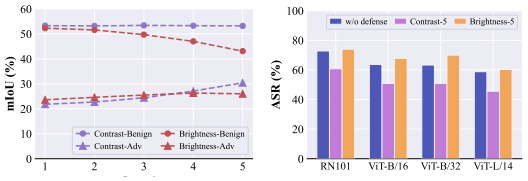
Table 24: Evaluation against defenses on ImageNet zero-shot classification. Results are ASR (%), where higher values indicate stronger attacks. X-T denotes X-Transfer. Bold numbers indicate the stronger attack within each paired comparison.

Backbone	No Defense		FD-1P		FD-2P		RS		JPEG Q90		JPEG Q75	
	X-T	Ours	X-T	Ours	X-T	Ours	X-T	Ours	X-T	Ours	X-T	Ours
RN101	**75.23**	72.7	62.07	**62.5**	61.10	**64.1**	33.97	**35.7**	**72.02**	71.2	63.56	**65.5**
ViT-B/16	**65.57**	63.7	55.21	**56.6**	53.64	**57.2**	38.49	**40.5**	**61.74**	59.5	51.88	**54.6**
ViT-B/32	59.80	**63.3**	55.09	**59.2**	55.04	**59.7**	43.00	**48.7**	57.15	**60.6**	54.90	**59.3**
ViT-L/14	53.03	**58.8**	39.18	**45.9**	37.26	**45.0**	**32.87**	32.0	44.20	**49.7**	34.54	**42.9**

Table 25: Evaluation against defenses on ADE20K-val segmentation with point prompts. Results are mIoU (%), where lower values indicate stronger attacks. Ours denotes DarkLLM. Bold numbers indicate the stronger attack within each paired comparison. Clean denotes evaluation without attacks or defenses.

Model	Clean	No Defense		FD-1P		FD-2P		RS		JPEG Q90		JPEG Q75	
		UAD	Ours	UAD	Ours	UAD	Ours	UAD	Ours	UAD	Ours	UAD	Ours
SAM-Base	50.3	42.74	**2.9**	43.65	**24.4**	44.90	**24.6**	46.19	**38.6**	43.63	**22.7**	44.37	**28.4**
SAM-Huge	53.6	48.11	**21.5**	48.75	**30.1**	49.59	**31.1**	50.07	**38.9**	48.54	**27.9**	49.01	**33.9**

Evaluation under Input Corruptions. We additionally evaluate DarkLLM under brightness and contrast transformations. For SAM-Huge, we use point prompts on ADE20K and vary corruption severity from 1 to 5. For CLIP, we evaluate ImageNet zero-shot classification at severity 5. Figure 7 reports the results. DarkLLM maintains satisfactory attack effectiveness across the different corruption types and severity levels.



(a) SAM (b) CLIP

Figure 7: Attack effectiveness under corruptions.

Perceptual Quality Evaluation We evaluate perceptual quality on 1,000 clean–adversarial image pairs using normalized L1 and L2 to measure pixel-level distortion, LPIPS-Alex (Zhang et al., 2018) to assess perceptual distance, and SSIM (Wang et al., 2004) to measure structural similarity. As shown in Table 26, DarkLLM achieves lower normalized L1 and L2 distortion than AnyAttack (0.0528 vs. 0.0540 and 0.0555 vs. 0.0568, respectively), together with a lower LPIPS-Alex distance (0.3292 vs. 0.3370), while AnyAttack obtains a higher SSIM (0.6481 vs. 0.6270). Overall, the two methods exhibit comparable metric values with different trade-offs: DarkLLM produces slightly lower pixel-level distortion and perceptual distance, whereas AnyAttack better preserves structural similarity according to SSIM.

Table 26: Perceptual quality on 1,000 clean–adversarial image pairs. Lower normalized L1/L2 and LPIPS-Alex and higher SSIM indicate better image preservation.

Method	Norm. L1 ↓	Norm. L2 ↓	LPIPS-Alex ↓	SSIM ↑
AnyAttack	0.0540	0.0568	0.3370	**0.6481**
DarkLLM (Ours)	**0.0528**	**0.0555**	**0.3292**	0.6270

D Qualitative Analysis

Success Cases. We provide further qualitative results to visually demonstrate the efficacy of DarkLLM. As shown in Figure 8, DarkLLM can generate diverse and potent targeted perturbations

against frontier commercial MLLMs from simple natural language instructions, successfully forcing them to output the specified target content. Figures 9 to 11 illustrate its impact on the SAM family. The adversarial noise substantially degrades the segmentation capability of the surrogate model, SAM-Base, causing it to fail across various prompts. More importantly, this same perturbation exhibits excellent transferability, successfully misleading the larger and unseen SAM-Large and SAM-Huge backbones.

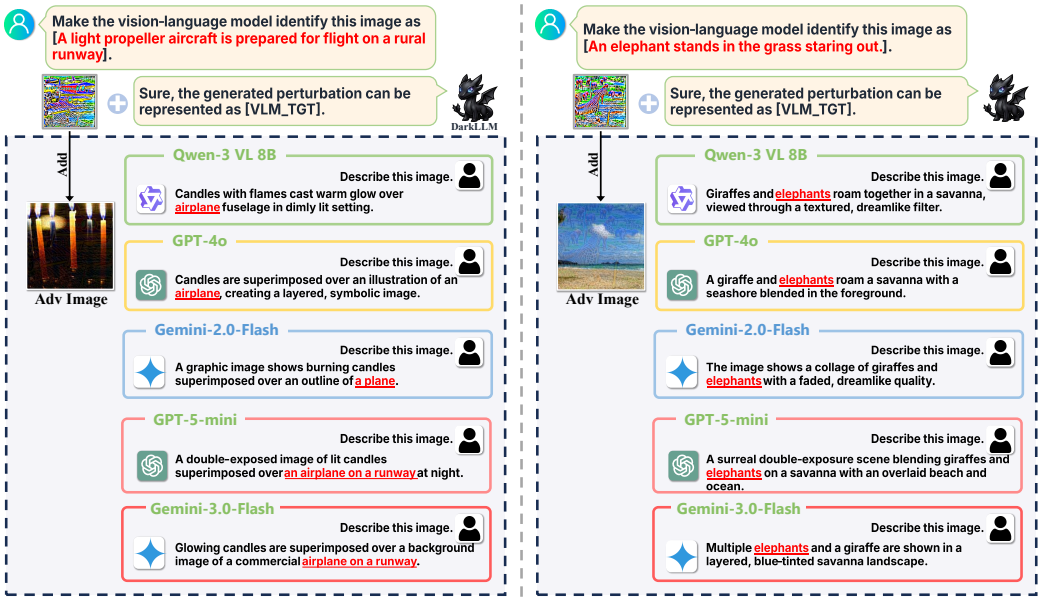


Figure 8: Visualization of DarkLLM for instruction-guided attacks on Commercial MLLMs.

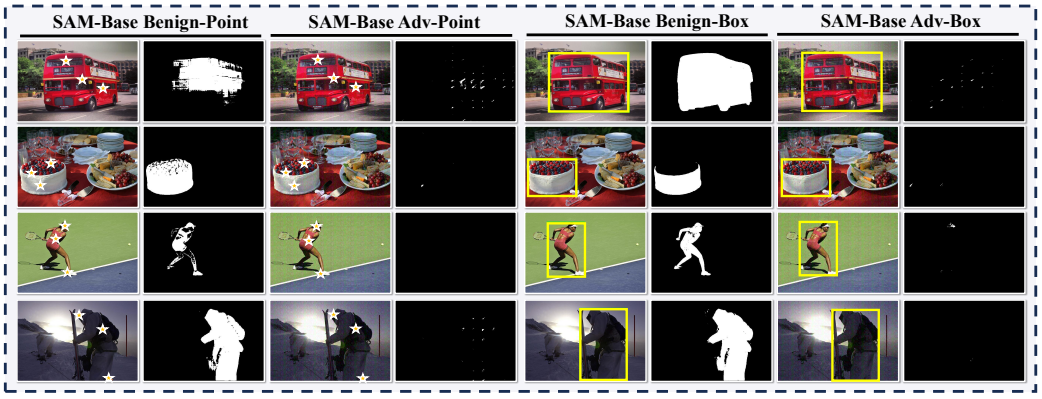


Figure 9: Visualization of DarkLLM for instruction-guided attacks on SAM-Base.

Failure Cases. Despite its strong overall performance, we also present failure cases to provide a transparent benchmark for future research. As shown in Figure 12, the generated perturbation can be less effective on images with highly complex backgrounds; we hypothesize that the adversarial noise may be masked by the image's inherent high-frequency details. Similarly, Figure 13 illustrates cases where a box prompt on a monochromatic object (an orange or a car) prevents the attack from successfully misleading the SAM-Huge model. We conjecture that the strong spatial prior provided by the box prompt, combined with the object's simple texture, makes the model more resilient to the perturbation in this specific scenario. Nonetheless, as the first language-driven adversarial framework of its kind, we believe DarkLLM possesses substantial potential. These identified limitations do not detract from its core contribution but rather provide valuable insights that will inform the future development of both more sophisticated language-driven attacks and their corresponding defenses.

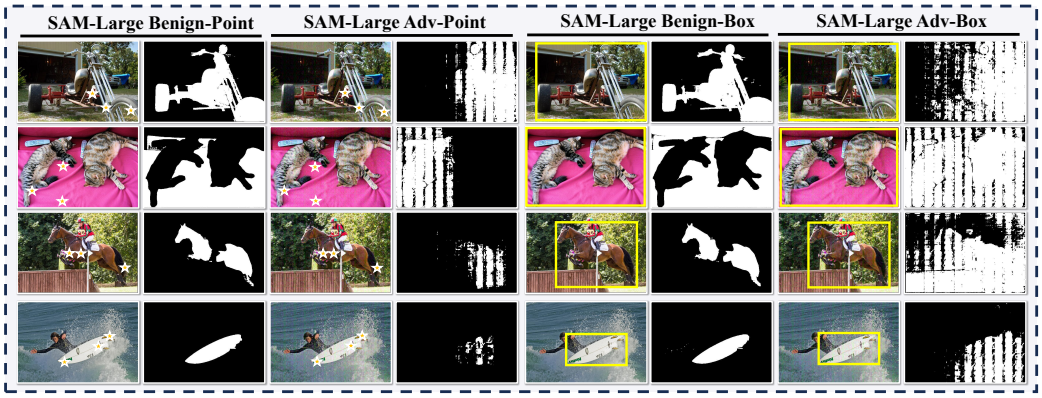


Figure 10: Visualization of DarkLLM for instruction-guided attacks on SAM-Large.

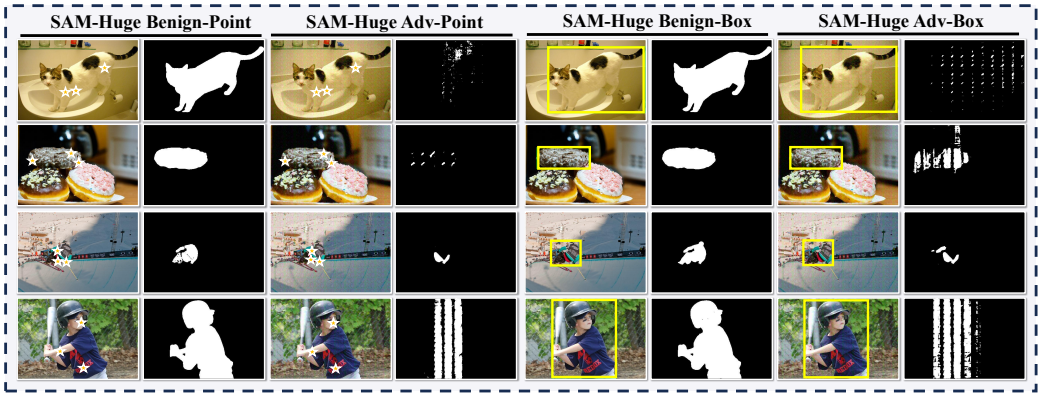


Figure 11: Visualization of DarkLLM for instruction-guided attacks on SAM-Huge.

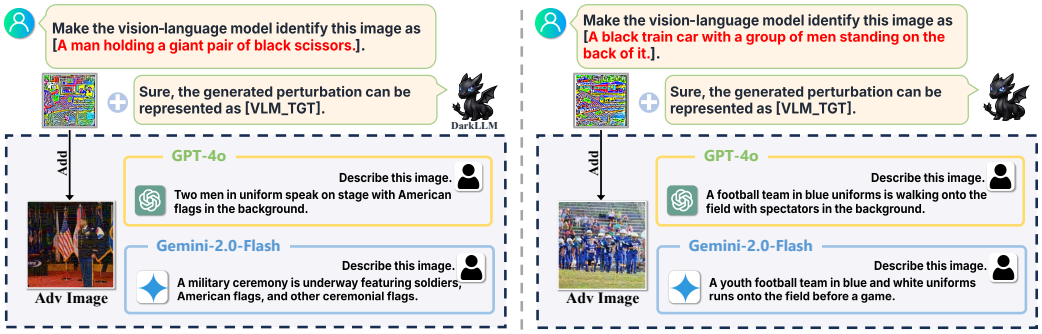


Figure 12: Failed cases of DarkLLM for instruction-guided attacks on Commercial MLLMs.

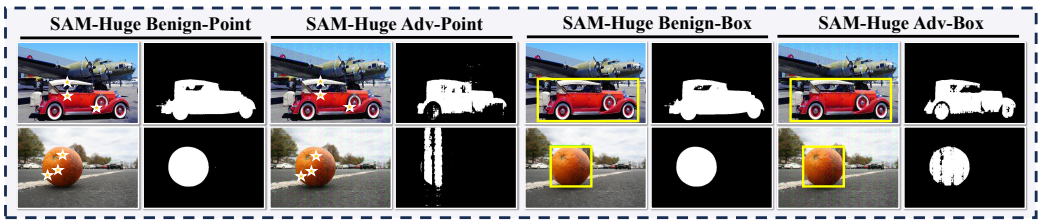


Figure 13: Failed cases of DarkLLM for instruction-guided attacks on SAM-Huge.

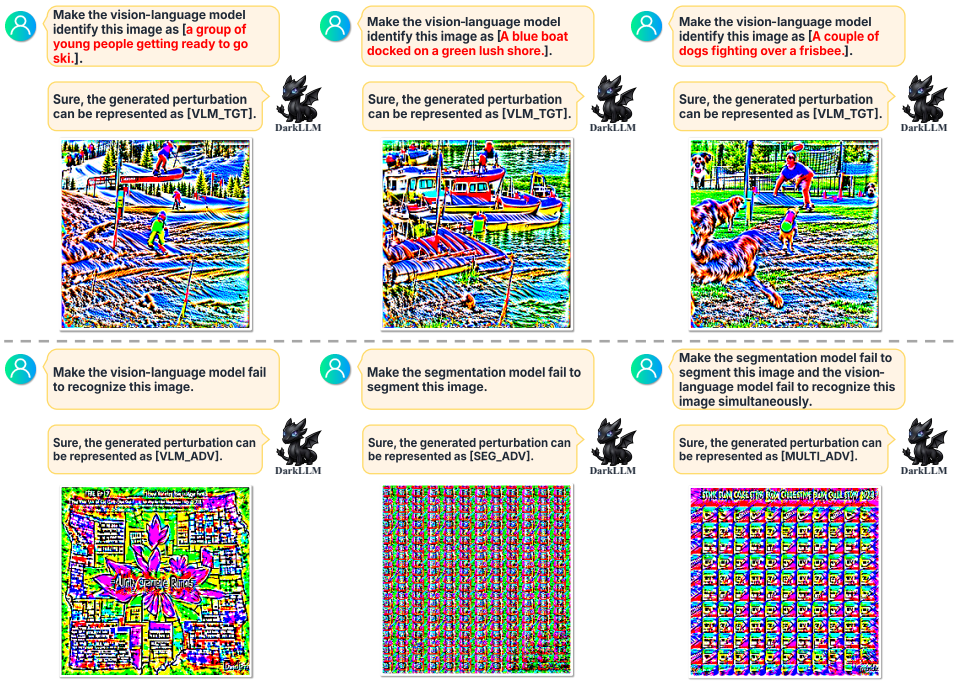


Figure 14: Details of perturbations generated by DarkLLM from user instructions.