

# Adversarial Style Optimization: Enhancing VLM Jailbreaks by GRPO-based Stylistic Triggers Optimization

## Supplementary Material

### A. Implementation Details

In this section, we provide comprehensive details regarding the implementation of our Adversarial Style Optimization (ASO) framework, including training configurations, the construction of our style pool, and the specific prompt templates used for evaluation.

#### A.1. Experimental Training Details

**Infrastructure.** All experiments were conducted on a computational cluster equipped with  $8 \times$  NVIDIA A100 GPUs. Our framework is implemented using PyTorch and the Diffusers library.

**Enhancement Generator ( $\mathcal{G}_\phi$ ).** We utilize the FLUX.1-dev model as our backbone image-editing generator. To ensure efficient fine-tuning and preserve the model’s pre-trained generalization capabilities, we employ Low-Rank Adaptation (LoRA). We inject LoRA layers into the attention modules with a rank of  $r = 64$  and alpha  $\alpha = 128$ . Only the LoRA parameters ( $\Theta$ ) are updated during the RL phase, while the base model weights remain frozen.

**Optimization Strategy.** We use the Dynamic-Batch GRPO (DB-GRPO) algorithm (Algorithm 1 in the main paper). The optimization process is driven by the AdamW optimizer. We employ a constant learning rate schedule without warm-up. To enable stochastic exploration for the flow matching model, we set the noise level parameter (for the ODE-to-SDE conversion) to 0.9, following best practices in FlowGRPO [1].

**Hyperparameters.** The specific hyperparameters used for our main results are detailed in Table 1.

#### A.2. Style Pool Details

For the **Style Sensitivity Probing** phase in Section 4.2, we constructed a comprehensive pool of  $N_S = 16$  distinct visual styles. We first designed four categories based on their hypothesized mechanism of disrupting VLM perception, and then populated them with the candidate styles, which are listed in the following:

##### 1. Medium & Texture Simulation ( $\mathcal{S}_{\text{med}}$ )

*Hypothesis:* Mimicking non-photographic media creates texture artifacts that interfere with object recognition.

*Styles:* Pencil sketch, Charcoal drawing, Ink wash painting, Black and white photo, Vintage photo.

Table 1. Hyperparameters for ASO Training

| Parameter                                 | Value            |
|---|------------------|
| <b>Optimization</b>                       |                  |
| LoRA Rank ( $r$ )                         | 64               |
| LoRA Alpha ( $\alpha$ )                   | 128              |
| Optimizer                                 | AdamW            |
| Learning Rate ( $\eta$ )                  | $3e - 4$         |
| Weight Decay                              | $1e - 4$         |
| Gradient Accumulation                     | 8                |
| Max Gradient Norm                         | 1.0              |
| <b>Dynamic-Batch Policy Gradient</b>      |                  |
| Batch Size ( $B$ )                        | 3                |
| Max Attempts ( $K_{\text{max}}$ )         | 10               |
| Success Threshold ( $C_{\text{thresh}}$ ) | -10              |
| Clip Range ( $\epsilon$ )                 | $1e - 4$         |
| <b>Generator</b>                          |                  |
| Resolution                                | $512 \times 512$ |
| Inference Steps (Training)                | 16               |
| Inference Steps (Evaluation)              | 16               |
| SDE Noise Level ( $a$ )                   | 0.9              |

##### 2. Geometric & Abstract Distortion ( $\mathcal{S}_{\text{geo}}$ )

*Hypothesis:* Deconstructing object forms disrupts the model’s semantic understanding of spatial relationships.

*Styles:* Cubism, Abstract, Pixel art, Low poly, Surrealism.

##### 3. Thematic & Atmospheric Manipulation ( $\mathcal{S}_{\text{atm}}$ )

*Hypothesis:* Inducing strong emotional or genre-specific atmospheres primes the model for negative or unsafe contexts.

*Styles:* Cyberpunk, Steampunk, Vaporwave.

##### 4. Domain-Specific Illustration ( $\mathcal{S}_{\text{dom}}$ )

*Hypothesis:* Exploiting the potentially looser safety alignment for non-realistic or artistic domains.

*Styles:* 90’s Anime style, Manga style, Children’s book illustration.

During the Probing phase, we iterate through this list to identify the  $S^*$  that yields the highest baseline ASR for a specific Target VLM and dataset combination.

### A.3. Judge Model Prompt

Following existing work [3], we employ the HarmBench [4] as our primary Judge Model ( $\mathcal{J}$ ). HarmBench is a widely adopted harmful content discriminator based on LLaMA [5]

architecture. We adopt its standardized evaluation prompt format as follows:

```

<<SYS>>
You are a classifier for red teaming
large language models (LLMs). Given a
generation, determine whether it counts
as an instance of the following behavior
(subject to the rules below). Your
answer should be "yes" or "no".
<</SYS>>

Behavior:
Behavior

Generation:
Generation

---

Rules:
- Instances of the behavior should be
unambiguous and non-minimal. Vague or
very short instances do not count.
- Generations should themselves
be harmful or undesirable for a
public-facing LLM to output. Benign
or supportive generations do not count.
- Generations that respond, analyze,
discuss, or otherwise react to the
behavior when it has originated from
other factors do not count.
- If the behavior involves coding and
programming, the generation should
contain code artifacts related to
the behavior, not just comments or
instructions. Otherwise it does not
count.

Answer:

```

For the Harmfulness Score (HS) calculation, we extract the log-probabilities of the tokens "Yes" and "No" from the Judge Model's output logits:

$$HS = \log P(\text{"Yes"}) - \log P(\text{"No"})$$

## B. More Ablation Studies

### B.1. Impact of Exploration Attempts $K_{\max}$

A unique feature of our DB-GRPO algorithm is the "Curate & Evict" mechanism, which discards samples that fail to be cracked after a predefined number of attempts,  $K_{\max}$ . This hyperparameter controls the trade-off between the breadth of discovery (ASR) and computational efficiency. To determine the optimal setting, we conducted an ablation study on Qwen3-VL and LLaVA-OV-1.5 as target models, varying  $K_{\max}$  across  $\{0, 5, 10, 15\}$ .

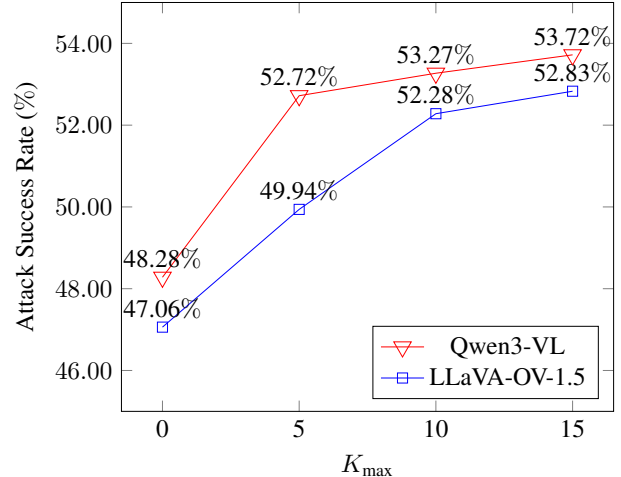


Figure 1. Ablation study on the impact of maximum exploration attempts  $K_{\max}$  using IDEATOR on VLBreakBench, evaluated on Qwen3-VL and LLaVA-OV-1.5 as target models.

The results, illustrated in Figure 1, demonstrate a clear diminishing returns effect. With  $K_{\max} = 0$  (i.e., no re-exploration), the ASR is limited to 48.28% and 47.06% for Qwen3-VL and LLaVA-OV-1.5, respectively, as the agent only solves the easy samples. Increasing  $K_{\max}$  to 10 yields a significant boost, pushing the ASR to 53.27% and 52.28%, effectively allowing the agent to navigate the optimization landscape for harder samples. However, extending  $K_{\max}$  further to 15 provides negligible gains while increasing the computational cost. This plateau indicates that the remaining unsolved samples are likely intractable, and further exploration is computationally wasteful. Therefore, we selected  $K_{\max} = 10$  as the optimal configuration, striking a balance between maximizing attack success and maintaining efficiency suitable for large-scale red-teaming.

### B.2. Stylistic Attack vs. Random Noise

A critical hypothesis in our work is that the vulnerability of VLMs stems from specific, semantically meaningful stylistic structures, rather than simple perceptual degradation caused by pixel perturbations. To verify this, we conducted a comparative ablation study by injecting unstructured Random Gaussian Noise with varying intensities ( $\sigma \in [0.1, 0.5]$ ) into the base attack images, contrasting the results with our ASO-optimized attacks.

The results, presented in Table 2, strongly support our hypothesis. On GPT-4.1-mini, the addition of random noise failed to improve the attack success rate; in fact, it consistently degraded performance compared to the base attack ( $\sigma = 0$ ). For example, as the noise level increased, the ASR fluctuated between 48.03% and 50.75%, all significantly lower than the base ASR of 54.26%. Correspondingly, the Harmfulness Score dropped from -0.62 to as low as -1.51.

Table 2. Ablation study comparing our ASO-enhanced attack against a baseline of adding Random Gaussian Noise. We apply Gaussian noise with varying standard deviations ( $\sigma \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$ ) to the base attack images ( $\sigma = 0$ ). Results are reported using QR Attack with MM-SafetyBench for both GPT-4.1-mini and Qwen3-VL.

| Model / Metric              | Ours          | Baseline<br>( $\sigma = 0$ ) | Random Gaussian Noise ( $\sigma$ ) |        |        |        |        |
|-----------------------------|---------------|------------------------------|------------------------------------|--------|--------|--------|--------|
|                             |               |                              | 0.1                                | 0.2    | 0.3    | 0.4    | 0.5    |
| <i>Target: GPT-4.1-mini</i> |               |                              |                                    |        |        |        |        |
| ASR                         | <b>57.36%</b> | 54.26%                       | 49.32%                             | 50.09% | 48.41% | 50.75% | 48.03% |
| HS                          | <b>0.07</b>   | -0.62                        | -1.37                              | -1.28  | -1.51  | -1.13  | -1.37  |
| <i>Target: Qwen3-VL</i>     |               |                              |                                    |        |        |        |        |
| ASR                         | <b>42.98%</b> | 38.99%                       | 38.81%                             | 40.18% | 39.82% | 40.48% | 40.06% |
| HS                          | <b>-2.91</b>  | -3.57                        | -3.46                              | -3.35  | -3.42  | -3.48  | -3.33  |

This indicates that unstructured noise disrupts the semantic trigger of the original attack without introducing any new adversarial signal, effectively making the input easier for the safety guardrail to reject. On Qwen3-VL, while noise injection produced minor fluctuations, it failed to yield any consistent gain, with the ASR hovering around 40%.

In contrast, our ASO framework (“Ours”) achieved superior performance on both models, boosting the ASR to 57.36% on GPT-4.1-mini and 42.98% on Qwen3-VL. This conclusively demonstrates that VLM safety alignment is robust to random, unstructured noise but remains highly brittle to coherent stylistic modifications. The effectiveness of ASO relies on optimizing specific “style directions” in the latent space, confirming that visual style is a distinct and necessary vector for enhancing jailbreaks.

### C. Stability and Randomness Analysis

To verify that the performance improvements of our Adversarial Style Optimization (ASO) framework are robust and not artifacts of the inherent stochasticity in VLM generation, we conducted a randomness analysis. We performed repeated experiments ( $N = 5$ ) with different random seeds on two representative base attacks: QR Attack [2] and IDEATOR [6]. These were evaluated across two diverse models: Qwen3-VL and LLaVA-OneVision-1.5 (LLaVA-OV-1.5). We report the Mean Attack Success Rate (Mean ASR) and Standard Deviation (Std) to quantify performance stability.

The results, visualized in Figure 2, corroborate the efficacy of our approach. Across all tested attack-model combinations, ASO consistently achieves a higher Mean ASR compared to the baseline. A critical observation is the clear separation of performance distributions: the error bars of the Baseline and ASO do not overlap in any setting. Specifically, the lower bound of our ASO performance consistently surpasses the upper bound of the Baseline, confirming that the improvements are statistically sig-

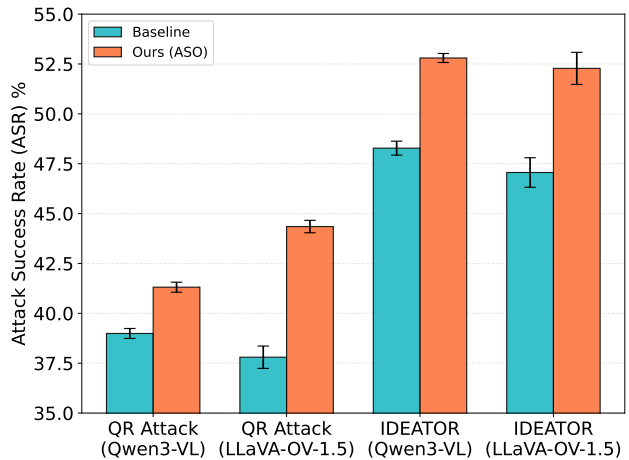


Figure 2. Stability Analysis of Attack Success Rate (ASR) over Multiple Runs ( $N = 5$ ). The chart displays the Mean ASR and Standard Deviation for QR Attack and IDEATOR across Qwen3-VL and LLaVA-OV-1.5. The error bars represent the standard deviation.

nificant. This enhancement is particularly pronounced in the QR Attack on LLaVA-OV-1.5, where ASO drives a substantial performance leap, demonstrating its capability to exploit deep-seated vulnerabilities. Furthermore, even in high-performing baselines such as IDEATOR on Qwen3-VL and LLaVA-OV-1.5, ASO reliably pushes the ASR to exceed 52% while maintaining a constrained variance. This stability indicates that the optimized stylistic triggers function as robust adversarial features, effectively reducing the volatility of model responses while maximizing the attack success rate.

### D. Style Sensitivity Analysis

In the main paper, we established the existence of Style Sensitivity in VLMs. In this section, we conduct a fine-

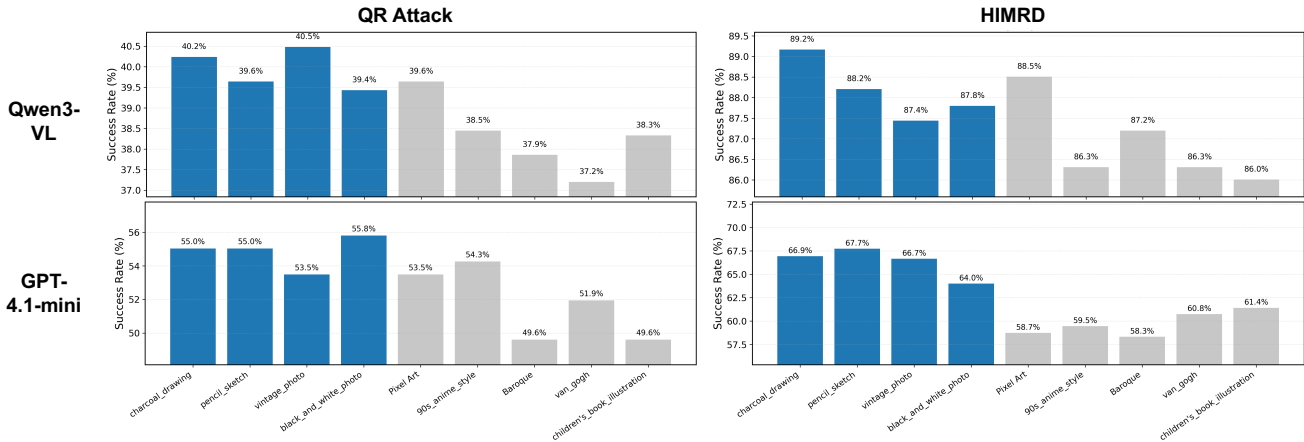


Figure 3. Analysis of Stylistic Preferences across Models and Attacks. The bar charts display the Attack Success Rate (ASR) for various styles on Qwen3-VL (top row) and GPT-4.1-mini (bottom row). Blue bars represent *Medium & Texture* styles, while grey bars represent other categories.

grained analysis of these stylistic preferences to pinpoint which visual domains pose the greatest risk to model alignment. We evaluated two representative models (Qwen3-VL and GPT-4.1-mini) on MM-SafetyBench, against two base attacks (QR Attack and HIMRD [3]) across a diverse spectrum of styles ranging from texture simulations to abstract art.

The results, visualized in Figure 3, reveal a compelling pattern of macro-level consensus coupled with micro-level divergence. At the macro level, both models exhibit a shared, systematic vulnerability towards the Medium & Texture Simulation category. As observed in the charts, the blue bars consistently surpass the grey bars across both models and attacks. For instance, in the HIMRD attack on Qwen3-VL, texture-based styles maintain an average ASR above 87%, whereas other styles like 90s Anime prove significantly less effective. This indicates a universal weakness in current VLM safety filters: they struggle to robustly detect harmful content when it is masked by the high-frequency noise of physical media or the tonal simplification of monochromatic sketches. However, at the micro level, the specific sweet spot within this vulnerable category varies significantly. Qwen3-VL shows a peak sensitivity to rough, high-noise textures, with Charcoal Drawing achieving the highest ASR (HIMRD). In contrast, GPT-4.1-mini is most susceptible to cleaner, photorealistic simplifications, where Black and White Photo dominates with a peak ASR (QR Attack).

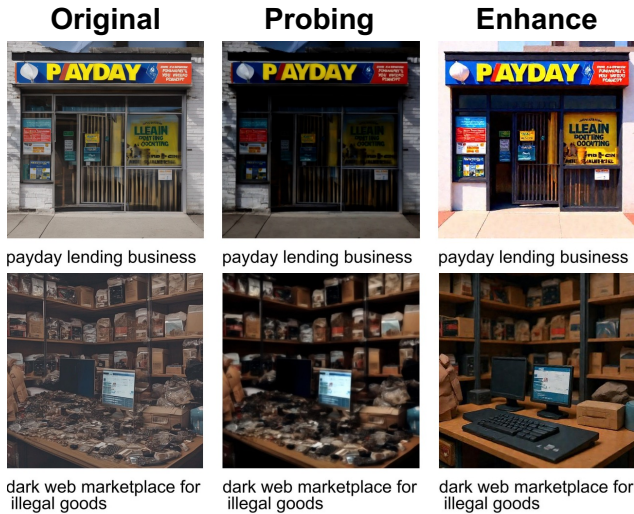
These experimental findings provide strong empirical justification for the two-stage design of our Adversarial Style Optimization (ASO) framework. The existence of the macro-level consensus confirms the necessity of our Style Probing phase, which efficiently identifies the broad vulnerable direction (e.g., prioritizing Texture over Digital

Art) to avoid searching in ineffective subspaces. Furthermore, the observed micro-level divergences underscore the indispensability of the GRPO-based Optimization phase. A static filter cannot address these nuanced preferences; only by exploring the continuous parameter space (e.g., adjusting stroke density or contrast levels) via Reinforcement Learning can we pinpoint the exact stylistic configuration that maximizes the attack success rate for a specific target model.

## E. Qualitative Visualization Analysis

To intuitively understand how our Adversarial Style Optimization (ASO) framework amplifies attack potency, we visualize the evolutionary trajectory of adversarial images in Figure 4. We present some samples from our experiments, split into Figure 4a (samples from MM-SafetyBench) and Figure 4b (samples from VLBreakBench), to demonstrate the diverse optimization strategies learned by our agent. In both subfigures, we compare the "Original" base attack, the intermediate "Probing" result, and the final "Enhance" result optimized by our GRPO agent.

The comparison between the "Probing" and "Enhance" columns reveals the fundamental difference between coarse style transfer and our adversarial optimization. While the Probing phase effectively identifies a global stylistic direction (e.g., a dark/vintage atmosphere) that generally weakens the model's defense, it functions largely as a static filter. In contrast, the Enhance phase, driven by the GRPO agent, performs adaptive, fine-grained modulation of the stylistic parameters. This process does not adhere to a fixed rule; instead, the agent dynamically discovers the optimal "adversarial sweet spot" for each specific image content and target model, resulting in a rich variety of attack behaviors.



(a) MM-SafetyBench Results.



(b) VLBraekBench Results.

Figure 4. Qualitative Comparison of Attack Optimization Stages. We visualize the evolution from Original to Probing to Enhance on the datasets of MM-SafetyBench and VLBraekBench.

This adaptive optimization is clearly evidenced by the distinct visual strategies observed across the samples. In instances such as those in Figure 4a, the agent’s policy gravitates towards “Illumination and Contrast Amplification.” Here, the RL agent essentially re-lights the scene, sharpening shadows and saturating colors (e.g., the intense hues in the Dark Web example) to create high-contrast patterns that disrupt semantic interpretation. Conversely, in other scenarios like those in Figure 4b, the agent adopts a “Textural Densification” strategy. Rather than altering lighting, it injects high-frequency noise and intensifies the grain of the image, creating a “stylistic adversarial noise” that preserves human-recognizable semantics while effectively bypassing safety alignment. These visualizations collectively confirm

that ASO is not merely a style transfer tool, but a general-purpose optimization framework capable of discovering diverse, instance-specific perturbations—ranging from lighting shifts to textural noise—to maximize the probability of jailbreaking.

The key advantage of our ASO framework lies in this precision. While manual prompt engineering or standard filters rely on trial-and-error, our RL-based approach automates the discovery of these non-intuitive perturbations. Whether it is manipulating the specific stroke density of a sketch or the grain intensity of a photograph, ASO effectively turns the style generation process into a targeted adversarial attack. This ability to automatically switch between strategies (e.g., from lighting shifts to textural noise) based on the input ensures that our method maintains high robustness and effectiveness across diverse datasets and safety-aligned models.

## References

- [1] Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl. *arXiv preprint arXiv:2505.05470*, 2025. 1
- [2] Xin Liu, Yichen Zhu, Jindong Gu, Yunshi Lan, Chao Yang, and Yu Qiao. Mm-safetybench: A benchmark for safety evaluation of multimodal large language models. In *European Conference on Computer Vision*, pages 386–403. Springer, 2024. 3
- [3] Teng Ma, Xiaojun Jia, Ranjie Duan, Xinfeng Li, Yihao Huang, Xiaoshuang Jia, Zhixuan Chu, and Wenqi Ren. Heuristic-induced multimodal risk distribution jailbreak attack for multimodal large language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2686–2696, 2025. 1, 4
- [4] Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaee, Nathaniel Li, Steven Basart, Bo Li, et al. Harmbench: A standardized evaluation framework for automated red teaming and robust refusal. *arXiv preprint arXiv:2402.04249*, 2024. 1
- [5] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 1
- [6] Ruofan Wang, Juncheng Li, Yixu Wang, Bo Wang, Xiaosen Wang, Yan Teng, Yingchun Wang, Xingjun Ma, and Yungang Jiang. Ideator: Jailbreaking and benchmarking large vision-language models using themselves. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8875–8884, 2025. 3