Reinforcing Multimodal Reasoning Against Visual Degradation

Rui Liu[1,2], **Dian Yu**[3], **Haolin Liu**[4], **Yucheng Shi**[1], **Tong Zheng**[2], **Runpeng Dai**[5], **Haitao Mi**[1], **Pratap Tokekar**[2]

[1]Tencent Hunyuan, [2]University of Maryland, College Park
[3]Meta, [4]University of Virginia, [5]University of North Carolina, Chapel Hill

Abstract

Reinforcement Learning (RL) has significantly advanced the reasoning capabilities of Multimodal Large Language Models (MLLMs), yet the resulting policies remain brittle against real-world visual degradations such as blur, compression artifacts, and low-resolution scans. Classical robustness techniques rely on static data augmentation or value-based regularization, which do not transfer to critic-free RL of autoregressive MLLMs, and naively rolling out on degraded images induces reward poisoning: occluded evidence triggers hallucinated trajectories and corrupts the reward signal. We propose ROMA, which samples trajectories and rewards only from clean images and re-evaluates the same trajectories under degraded views via teacher forcing. ROMA combines a worst-case token-level KL penalty for distributional consistency, an auxiliary policy gradient anchored to clean-image advantages that preserves a reliable learning signal, and correctness conditioning that restricts invariance enforcement to successful trajectories. On Qwen3-VL 4B/8B across seven multimodal reasoning benchmarks, ROMA improves robustness on both seen and unseen corruptions over GRPO while matching clean accuracy, with gains that hold across random seeds, RL algorithms, and backbones and transfer to naturally degraded real-world images.

1 Introduction

Reinforcement Learning (RL) [38] has driven a paradigm shift in the training of large language models, unlocking strong reasoning capabilities [3, 5, 6, 8, 13, 16, 27, 32, 52]. These advances have been extended to multimodal large language models (MLLMs) [1, 2, 12, 15, 21–23], enabling reasoning over rich visual inputs. However, such capabilities are typically developed in controlled settings with clean, well-curated data. In real-world deployment, MLLMs must contend with noisy and unstructured visual inputs, including blurry photographs, compression artifacts, and low-resolution document scans, and a model that performs reliably on a clean input (e.g., a high-quality PDF) often fails catastrophically on a degraded version of the same content. This brittleness to visual degradation poses a critical barrier to the reliable deployment of reasoning-capable MLLMs.

Visual robustness has been extensively studied in computer vision and reinforcement learning. In vision, robustness is typically pursued through data augmentation such as cropping, cutout, and flipping, often combined with contrastive objectives [31, 35, 37]. In deep RL, a parallel line of work has shown that injecting visual augmentations during training improves out-of-distribution generalization [10, 14, 28, 36, 50], transferring invariance learning from static perception to sequential decision-making.

Despite this progress, the visual robustness of reasoning-capable MLLMs remains underexplored, and enforcing robustness during RL fine-tuning introduces challenges that are absent in standard settings.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

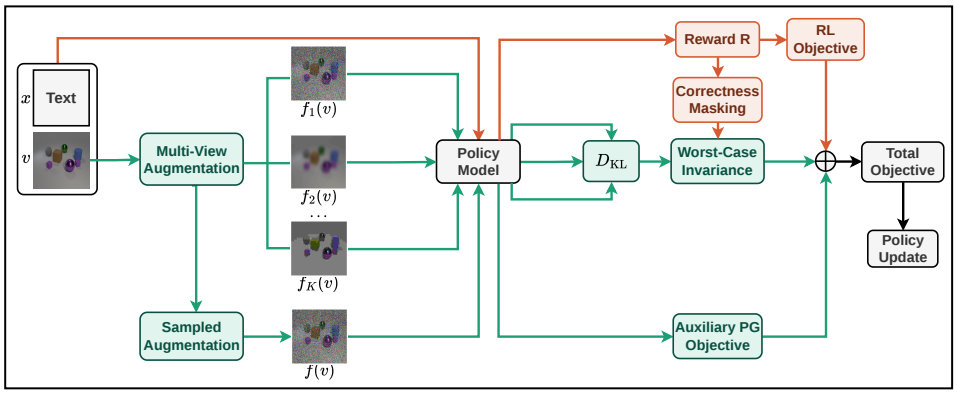


Figure 1: **Overview of ROMA**. A standard RL rollout on the clean input yields a trajectory and reward defining the main RL objective. The trajectory is then re-evaluated under perturbations via two branches: a worst-case invariance branch applying a token-level KL penalty against the most divergent of multiple degraded views, gated by a correctness mask so it fires only on successful trajectories; and an auxiliary policy gradient branch computing a clipped surrogate on a sampled degraded view, anchored to clean-image advantages. The three objectives combine into a total objective for the policy update. No rollout is sampled from a degraded input, avoiding reward poisoning.

First, architectural mismatch. Modern RL fine-tuning of autoregressive models increasingly relies on critic-free algorithms such as Group Relative Policy Optimization (GRPO) [39] to avoid the memory overhead of value networks; consequently, classical value-based robustness regularizers [36] do not apply out of the box. **Second, reward poisoning.** Naively rolling out on degraded inputs can obscure perceptual evidence and force the model to hallucinate [24], so the resulting reward signal penalizes perceptual failure rather than reasoning errors and starves many prompts of learning signal, which can destabilize optimization. Empirically, rolling out on degraded images lowers the mean reward and nearly doubles the fraction of prompt groups that receive zero advantage (Section 4.3). These challenges motivate our central question: *how can we make RL-fine-tuned MLLMs robust to visual degradation without sacrificing reasoning fidelity or destabilizing training?*

To answer this, we propose **ROMA**, a novel RL fine-tuning framework situated at the intersection of **M**ultimod**A**l reasoning and **RO**bust reinforcement learning. Unlike prior approaches that rely on static augmentation [14, 24, 49], ROMA modifies the RL optimization dynamics directly to reinforce reasoning against visual degradation while preserving clean-input performance.

At the core of our ROMA is a *dual-forward-pass* training strategy over a critic-free autoregressive MLLM, as illustrated in Figure 1. The first pass performs standard RL rollouts on the clean image, producing reasoning trajectories and their advantages. The second pass generates multiple degraded views of the same image and re-evaluates the *same* frozen trajectory via teacher forcing, computing token-level log-probabilities under each corrupted view without sampling new rollouts. This sidesteps reward poisoning by construction: trajectories are never sampled from degraded inputs, yet we still observe how the model's token distributions shift under perturbation.

On top of this scaffold, ROMA introduces three regularizers that together yield robust reasoning. (i) A token-level surrogate KL penalty enforces distributional consistency between clean and degraded views, applied in a *worst-case* fashion against the augmentation with the largest divergence. (ii) An auxiliary policy gradient loss is computed on a randomly sampled degraded view but *anchored to clean-image advantages*, preserving a reliable reward signal and preventing collapse under regularization. (iii) *Correctness-conditioned* regularization restricts invariance enforcement to successful trajectories, so the model is not pushed toward becoming consistently but systematically incorrect.

We validate ROMA by fine-tuning Qwen3-VL 4B and 8B Instruct models [1] and evaluating visual robustness across seven multimodal reasoning benchmarks: MathVista [26], WeMath [34], ChartQA [29], LogicVista [47], MMStar [4], VisualPuzzles [40], and RealWorldQA [46]. While standard GRPO reaches strong clean-input accuracy (68.9% at 8B), it degrades sharply under corruption, falling to 59.2% on seen and 54.0% on unseen perturbations. ROMA matches clean performance (68.7%) while substantially improving robustness, reaching 61.6% on seen (+2.4%) and 56.3% on unseen (+2.3%) perturbations, with consistently smaller clean-to-degraded gaps.

In summary, our key contributions are as follows:

- We propose ROMA, an RL fine-tuning approach for MLLMs that enforces robustness to visual degradation.
- ROMA combines a correctness-conditioned, token-level KL invariance penalty applied in a worst-case multi-view manner with an auxiliary policy gradient anchored to clean advantages, enabling stable robustness learning in critic-free settings.
- ROMA improves robustness on seven multimodal benchmarks empirically, achieving higher accuracy under both seen and unseen corruptions while maintaining strong clean-input performance. The gains are stable across random seeds, hold across RL algorithms (GRPO, DAPO) and backbones (Qwen3-VL, InternVL3), exceed matched augmentation baselines, and transfer to naturally degraded real-world benchmarks.

2 Related Work

Visual Robustness and Data Augmentation in RL. Data augmentation has a long history in deep RL for visual robustness. DrAC [36], RAD [14], and DrQ [50] show that augmentations such as cropping, blurring, or flipping improve OOD generalization. In these actor-critic setups, robustness comes from regularizing both the policy and the value network toward consistent representations across clean and augmented states [10, 28]. Related work addresses learning under uncertainty via conformal guidance [19], distributionally robust control [17], and multimodal learning with missing modalities [18, 20].

These regularizers, however, do not transfer directly to MLLM fine-tuning, which is critic-free and sensitive to the semantics of the visual input. ROMA reformulates visual invariance for large-scale, critic-free generative models: it replaces value-based regularization with a token-level surrogate KL penalty, and instead of uniform augmentation it targets the most adversarial corruption at each step via a worst-case multi-view strategy. Combined with an auxiliary policy gradient objective, this enables robust invariance learning while preserving the logical consistency required for multimodal reasoning.

Reinforcement Learning for Multimodal Reasoning. RL has recently emerged as a powerful paradigm for eliciting complex reasoning in MLLMs. Tan et al. [42] adapt text-based reasoning paradigms to multimodal settings, Peng et al. [33] scale mathematical reasoning and cross-modality alignment, Yang et al. [48] improve visual question answering, and Huang et al. [12] employ vision-grounded prompts to facilitate multi-step logic; other work improves the stability and reward signals of multimodal RL [15, 21, 22]. A recent line of research leverages visual perturbations: Wang et al. [45] encourage visual grounding by penalizing the policy when its outputs remain unchanged under heavy masking, Liu et al. [24] inject data augmentation into rollouts to reinforce visual exploration, and Liu et al. [23] use visual uncertainty to guide policy exploration.

Despite these advances, robustness to visual degradation in RL-based multimodal reasoning remains underexplored, and none of these methods decouples trajectory and reward generation from robustness learning. ROMA addresses this gap with a clean-rollout / degraded-view decomposition: trajectories and rewards come only from clean images, while the same trajectories are teacher-forced under degraded views, so degraded inputs shape optimization without producing unreliable trajectories or reward targets. Together with correctness-conditioned invariance, this prevents the reward poisoning common in naive augmentation.

3 Approach

In this section, we present our approach for improving the visual robustness and OOD generalization of MLLMs trained via RL. We first formalize the autoregressive fine-tuning setting, and then introduce our key components: a correctness-conditioned, token-level invariance regularization objective, and a worst-case multi-view optimization strategy combined with an auxiliary policy gradient objective to enforce robustness.

Problem Formulation. We consider a multimodal reasoning task where a MLLM produces a logical chain-of-thought to answer a visual query. Each input consists of a text question x and an associated image v. To solve the task, the MLLM acts as a stochastic policy π_θ, parameterized by θ, generating a step-by-step reasoning trajectory $y \sim \pi_\theta(\cdot \mid v, x)$. Upon generating the complete

trajectory y, a reward function evaluates its correctness and yields a scalar reward $R(v, x, y)$. The standard reinforcement learning objective seeks to maximize this expected reward:

$$J_{\text{RL}}(\theta) = \mathbb{E}_{(v,x)\sim\mathcal{D},\, y\sim\pi_\theta(\cdot|v,x)}\left[R(v, x, y)\right] \tag{1}$$

However, optimizing this objective solely on clean images v leads to policies that fail to generalize under real-world visual degradations (e.g., blur, sensor noise, and compression artifacts). Consequently, our goal is to regularize π_θ such that the generated trajectory y remains robust and logically consistent even under degraded visual inputs.

Correctness-Conditioned Token-Level Invariance. To embed visual invariance directly into the autoregressive generation process, we draw inspiration from [36]. Traditional actor-critic methods enforce invariance jointly across both policy and value networks. However, modern large-scale RL frameworks (e.g., GRPO [39]) are inherently critic-free, making value-based regularization inapplicable. We therefore isolate the policy invariance objective and reformulate it as a token-level surrogate KL divergence penalty tailored to autoregressive generation.

Let $f \in \mathcal{F}$ be a stochastic visual augmentation function, such that $f(v)$ produces a degraded view of the original input v. To enforce perceptual invariance, the token distribution under the degraded view should align with that of the clean view. Treating the clean visual state as a reference anchor, we penalize the divergence between the degraded and clean policy logits. To prevent the noisy gradients from corrupting the clean representations, we apply a stop-gradient operator ($\text{sg}[\cdot]$) to the clean policy outputs. For a given trajectory y sampled from the old policy π_{old}, the invariance penalty is defined as:

$$G_\pi(\theta, f) = \mathbb{E}_{(v,x)\sim\mathcal{D},\, y\sim\pi_{\text{old}}}\left[\sum_{t=1}^{|y|} D_{\text{KL}}\Big(\text{sg}\big[\pi_\theta(\cdot \mid v, x, y_{<t})\big] \,\|\, \pi_\theta(\cdot \mid f(v), x, y_{<t})\Big)\right], \tag{2}$$

where the per-token KL divergence is practically approximated via the standard RL surrogate: $D_{\text{KL}}^{(t)} \approx p_t \cdot (\log p_t - \log q_t)$, with $p_t = \text{sg}\big[\pi_\theta(y_t \mid v, x, y_{<t})\big]$ and $q_t = \pi_\theta(y_t \mid f(v), x, y_{<t})$. This sampled-token expression is a practical, biased surrogate for the full categorical KL rather than an exact computation over the vocabulary; we adopt it for tractability, since the exact KL requires materializing next-token distributions over the entire vocabulary for every degraded view, which is prohibitively memory-intensive during RL training. Crucially, enforcing consistency across views is actively harmful if the underlying trajectory y is hallucinated or factually incorrect. To prevent the policy from becoming robustly incorrect, we introduce a correctness mask, applying the penalty strictly to trajectories that successfully solve the task ($R > 0$).

Worst-Case Multi-View Optimization. During standard training, randomly sampled augmentations may be visually trivial, providing weak regularization signals. To enforce rigorous adversarial robustness, we depart from single-view augmentation in favor of a worst-case multi-view strategy.

At each training step, we sample a subset of K distinct augmentations, $\mathcal{F}_K \subset \mathcal{F}$, generating K degraded views. We compute the token-level invariance penalty $G_\pi(\theta, f_k)$ for all K views. Rather than averaging these penalties, we apply a minimax formulation, regularizing the policy exclusively against the augmentation that induces the maximum divergence:

$$G_\pi^{\text{worst}}(\theta) = \max_{f_k \in \mathcal{F}_K} G_\pi(\theta, f_k). \tag{3}$$

Auxiliary Policy Gradient Loss. While G_π^{worst} enforces distributional consistency, excessive KL regularization without a grounding reward signal can induce policy collapse, where the MLLM learns to output consistent but nonsensical tokens. To provide an active learning signal under degradation, we introduce an auxiliary policy gradient objective ($J_{\text{aug_pg}}$). We compute an additional clipped-surrogate objective directly on the augmented logits of a randomly sampled view. Crucially, to prevent reward poisoning, we evaluate this objective using the exact token trajectories y and advantages $A(v, x, y)$ derived from the clean rollout:

$$J_{\text{aug_pg}}(\theta) = \mathbb{E}_{f\sim\mathcal{F}_K}\left[\mathbb{E}\left[\mathbb{I}[R(v, x, y) > 0] \sum_{t=1}^{|y|} \min\left(\rho_t A, \text{clip}(\rho_t, 1-\epsilon, 1+\epsilon)A\right)\right]\right], \tag{4}$$

where $f \sim \mathcal{F}_K$ is a randomly sampled augmentation function from the augmentation pool, and the importance sampling ratio is $\rho_t = \frac{\pi_\theta(y_t|f(v),x,y_{<t})}{\pi_{\text{old}}(y_t|v,x,y_{<t})}$. By anchoring both the rollout generation and the advantage computation to the clean images, we force the model to actively maximize the expected reward under visual noise without training on structurally hallucinated exploration paths. As with the invariance penalty, this objective is gated by the correctness mask, so consistency under degradation is only encouraged for trajectories that solve the task on the clean image.

The final consolidated optimization objective for our robustness training is formulated as follows:

$$J_{\text{total}}(\theta) = J_{\text{RL}}(\theta) + \alpha \cdot J_{\text{aug_pg}}(\theta) - \beta \cdot \mathbb{E}\left[G_\pi^{\text{worst}}(\theta) \cdot \mathbb{I}[R(v,x,y) > 0]\right] \tag{5}$$

where $J_{\text{RL}}(\theta)$ represents the main reinforcement learning objective (e.g., GRPO), α and β are coefficients controlling the strength of the auxiliary optimization and the worst-case invariance penalty, respectively. Ultimately, we update the policy parameters θ to maximize $J_{\text{total}}(\theta)$. This unified objective simultaneously drives the MLLM to maximize logical reasoning performance on clean inputs (J_{RL}), actively learn robust feature representations under visual degradation ($J_{\text{aug_pg}}$), and minimize the worst-case distributional divergence between the clean and degraded reasoning paths ($-G_\pi^{\text{worst}}$).

4 Experiments

To evaluate the effectiveness of our proposed framework, we design experiments to answer the following questions: (1) Does our approach improve the robustness of MLLMs against visual degradation? (2) Does the framework generalize to out-of-distribution (OOD) visual corruptions not seen during training? (3) How do individual components, such as worst-case optimization, auxiliary policy gradients, and correctness-conditioning, contribute to the overall performance?

4.1 Experimental Setup

Implementation Details. We conduct direct RL training on the Qwen3-VL-4B and 8B Instruct [1] models, using GRPO as the underlying RL algorithm. The models are trained to generate responses in a structured format, where the reasoning process is enclosed within `<thinking></thinking>` tags and the final answer is presented in `\boxed{}`. For our robustness framework, we set the multi-view sample size to $K = 3$ augmentations per step. The auxiliary augmented policy gradient coefficient α is set to 0.10, and the worst-case invariance regularization coefficient β is set to 0.10. Please see a series of sensitivity analysis for these values in Section 4.5. The implementation is built on the EasyR1 framework [53]. More implementation details can be found in Appendix A.1.

Dataset and Evaluation. We train all models on the MMRL30k dataset [55], which contains around 30K samples. We evaluate on seven multimodal reasoning benchmarks, including MathVista [26], WeMath [34], ChartQA [29], LogicVista [47], MMStar [4], VisualPuzzles [40], and RealWorldQA [46]. These benchmarks cover a diverse range of multimodal reasoning, including mathematical problem solving, chart understanding, general visual reasoning, and logical inference. For evaluation, final answers are extracted from the `\boxed{}` span by rule, and Qwen2.5-72B-Instruct [43] is used only as a fallback extractor for the minority of responses that do not follow this format, following prior work [21, 23, 41, 55]. Correctness is then determined by rule-based exact match against the reference answer, so no model-based judging is involved.

Baselines. We evaluate our approach against controlled baselines: (1) Base model: the pre-trained, instruction-tuned model prior to any RL fine-tuning. (2) GRPO: a model fine-tuned via standard GRPO on clean data. (3) NoisyRollout [24] and a reward-filtered variant under the same setting (Section 4.3). In addition, for broader context, we include evaluated results from several external models, including NoisyRollout-7B [24], PAPO-7B [45], Vision-R1-7B [12], VL-Rethinker-7B [44], and OpenVLThinker-7B [7]. Vision-R1-7B used WeMath as training data, its performance on that benchmark is omitted.

Degradation Protocols. We systematically evaluate our approach across three settings: (1) Clean, (2) Seen degradations, and (3) Unseen degradations. The seen setting addresses Question 1 by measuring robustness against the types of visual degradations experienced during training. Inspired by the ImageNet-C framework [11], this pool simulates common image capture and transmission artifacts: Gaussian noise, Gaussian blur, JPEG compression, and resolution downscaling. Conversely, the unseen setting addresses Question 2 by assessing OOD generalization across novel corruption types. This pool subjects the model to corruptions strictly held out during training: motion blur,

Table 1: **Robustness performance evaluation with the 4B model**. To present a consolidated view of visual robustness, we report performance on degraded inputs as macro-averages across both seen and unseen degradation types. Our approach demonstrates superior robustness, achieving the highest average accuracy across both seen and unseen degradations.

Method	MathVista	WeMath	ChartQA	LogicVista	MMStar	VisPuzzles	RealWorldQA	Avg
Clean								
Qwen3-VL-4B Instruct [1]	76.6	64.7	79.2	59.6	66.3	41.9	69.1	65.3
+ GRPO	78.0	76.6	81.3	58.0	69.1	40.8	70.1	67.7
+ ROMA	78.4	76.4	80.6	60.0	69.9	42.3	69.5	68.2
Seen Degradations								
Qwen3-VL-4B Instruct [1]	70.3	63.9	46.3	52.6	64.6	39.9	65.2	57.5
+ GRPO	70.7	74.3	48.0	52.4	64.0	39.2	64.6	59.0
+ **ROMA**	**73.1**	**75.0**	**48.4**	**55.6**	**66.0**	**41.5**	**65.5**	**60.7**
Unseen Degradations								
Qwen3-VL-4B Instruct [1]	61.8	55.7	41.2	48.5	59.1	35.3	60.6	51.7
+ GRPO	62.9	65.6	42.4	48.5	59.6	36.6	61.2	53.8
+ **ROMA**	**64.5**	**67.0**	**43.4**	**49.6**	**61.1**	**37.2**	**62.6**	**55.1**

salt-and-pepper noise, speckle noise, posterization, and pixelation. Detailed degradation parameters and visual examples are provided in Appendix A.2 and Figure 3. Crucially, for the main results, we evaluate performance at a severe magnitude (Level 3) that strictly exceeds the parameter bounds used during training, thereby testing the model's ability to extrapolate to unseen severity distributions.

4.2 Main Results

Tables 1 and 2 present the main evaluation results for the Qwen3-VL 4B and 8B Instruct models, respectively. To provide a consolidated view of visual robustness, results under degradation are reported as macro-averages across all specific perturbation types within the seen and unseen pools for each dataset.

We first establish the baseline performance on clean data. As shown, standard GRPO yields solid improvements over the base model on clean data, achieving an average score of 67.7% (compared to the 4B base model's 65.3%) and 68.9% (compared to the 8B base model's 66.8%). Our approach performs comparably to GRPO on these clean inputs for both the 4B (68.2%) and 8B (68.7%) models. This demonstrates that our anchored optimization framework successfully preserves foundational reasoning capabilities without compromising baseline performance.

Robustness to Visual Degradations. We next evaluate the models under the Seen degradation setting to measure visual robustness. As detailed in the Tables 1 and 2, GRPO suffers a larger performance drop when transitioning from clean to degraded inputs, decreasing by 8.7% (from 67.7% to 59.0%) for the 4B model, and by 9.7% (from 68.9% to 59.2%) for the 8B model. Standard GRPO struggles to maintain performance under visual perturbations. In contrast, our approach consistently outperforms GRPO across all benchmarks under degraded conditions. The performance gap between clean and degraded inputs for our 8B model is reduced to a drop of 7.1%, compared to the 9.7% drop observed in GRPO. By anchoring the advantage computation to clean inputs and penalizing structural deviation via the token-level invariance penalty, our framework successfully mitigates the impact of perceptual artifacts encountered during training.

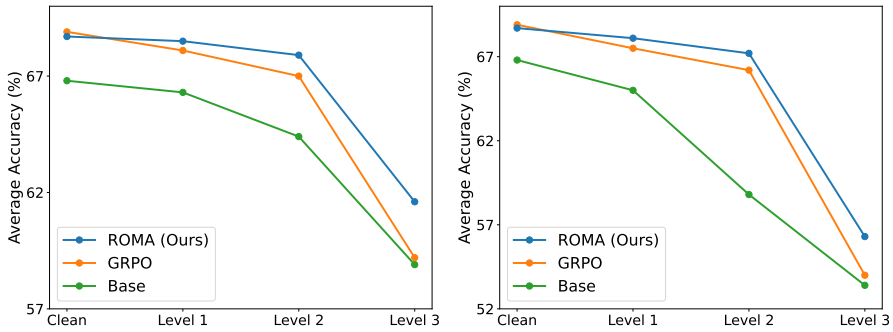
Generalization to OOD Degradations. Furthermore, we evaluate the OOD generalization of our approach on degradation types completely unseen during training. As shown in Table 1 and Table 2, our approach exhibits stronger zero-shot generalization to these unseen corruptions for both model sizes. Under OOD conditions, the 8B GRPO performance drops to 54.0%. However, our framework sustains an average score of 56.3%, outperforming the standard RL baseline. Additionally, the performance drop from clean to OOD evaluation is 12.4% for our 8B method, which is smaller than the 14.9% decrease observed in GRPO. This confirms that the robustness acquired on seen degradations transfers effectively to unseen domains, validating that our token-level constraint encourages generalized resilience without overfitting to the training distribution.

Performance Across Degradation Levels. We further measure accuracy under progressively stronger corruptions, from Clean to Level 3 (severe), in Figure 2. While all methods degrade as severity increases, our method shows the smallest drops and the highest accuracy at Level 3,

Table 2: **Robustness performance evaluation with the 8B model**. To present a consolidated view of visual robustness, we report performance on degraded inputs as macro-averages across both seen and unseen degradation types. Our approach demonstrates superior robustness, achieving the highest average accuracy across both seen and unseen degradations.

Method	MathVista	WeMath	ChartQA	LogicVista	MMStar	VisPuzzles	RealWorldQA	Avg
Clean								
OpenVLThinker-7B [7]	67.0	60.6	78.7	48.0	60.1	32.0	58.6	57.9
Vision-R1-7B [12]	72.4	–	81.6	48.7	62.7	36.1	66.1	–
NoisyRollout-7B [24]	72.7	69.3	79.8	50.0	63.2	37.7	67.1	62.8
PAPO-7B [45]	75.5	71.0	82.0	53.3	63.2	35.8	67.3	64.0
VL-Rethinker-7B [44]	72.7	67.5	79.9	46.9	61.9	34.8	68.5	61.7
Qwen3-VL-8B Instruct [1]	76.6	69.4	79.4	60.7	68.7	43.5	69.4	66.8
+ GRPO	78.4	77.6	81.5	60.8	70.1	43.5	70.6	68.9
+ ROMA	78.5	77.9	80.8	62.1	69.5	42.5	69.9	68.7
Seen Degradations								
OpenVLThinker-7B [7]	60.6	60.5	45.1	44.4	56.3	32.8	54.8	50.6
Vision-R1-7B [12]	63.9	–	47.4	46.0	58.9	34.8	60.6	–
NoisyRollout-7B [24]	66.9	67.4	46.1	46.5	60.4	34.6	62.1	54.9
PAPO-7B [45]	68.2	70.0	47.3	47.2	59.8	34.2	60.9	55.4
VL-Rethinker-7B [44]	64.4	67.1	45.9	46.0	58.1	33.8	63.1	54.1
Qwen3-VL-8B Instruct [1]	70.7	69.4	46.9	54.6	64.8	40.1	65.7	58.9
+ GRPO	71.2	75.0	48.0	52.7	64.0	38.9	64.8	59.2
+ **ROMA**	**73.3**	**77.3**	**49.1**	**57.5**	**66.3**	**41.7**	**66.0**	**61.6**
Unseen Degradations								
OpenVLThinker-7B [7]	54.4	54.7	41.6	41.8	51.6	30.1	50.9	46.4
Vision-R1-7B [12]	58.3	–	44.8	42.0	54.3	31.6	57.1	–
NoisyRollout-7B [24]	60.5	62.1	43.0	42.1	55.4	30.8	56.7	50.1
PAPO-7B [45]	60.8	63.6	42.6	43.3	55.5	31.4	57.4	50.7
VL-Rethinker-7B [44]	57.6	61.6	43.1	41.8	53.6	30.2	59.0	49.6
Qwen3-VL-8B Instruct [1]	63.1	60.4	42.1	50.0	59.5	37.3	61.4	53.4
+ GRPO	63.0	66.8	43.0	47.1	59.8	36.8	61.4	54.0
+ **ROMA**	**64.8**	**70.4**	**44.1**	**50.8**	**62.1**	**38.0**	**63.6**	**56.3**

outperforming GRPO by +2.4% and +2.3%, respectively, indicating improved robustness to both seen and unseen perturbations.



(a) Evaluation on seen degradations. (b) Evaluation on unseen degradations.

Figure 2: **Robustness under increasing visual corruption severity**. We report accuracy from Clean to Level 3 (severe) for seen and unseen degradations. ROMA consistently achieves higher accuracy at severe corruption levels and exhibits smaller performance degradation compared to the base model and GRPO.

4.3 Further Analysis

Matched Augmentation Baselines and Multiple Seeds. We retrain NoisyRollout [24] with the same backbone, data, and budget as ROMA, and strengthen it with reward filtering, which discards all-wrong groups before each update. Both variants match ROMA on clean inputs but trail it under seen and unseen degradations (Table 3), confirming the advantage of our decomposition over direct augmentation. Over three seeds, ROMA improves seen and unseen accuracy over GRPO by +2.5% and +2.7% with small standard deviations while matching clean accuracy.

Generalization Across RL Algorithms and Backbones. ROMA is not tied to GRPO or Qwen3-VL (Table 3). With DAPO [51], it improves seen and unseen accuracy by +2.4% and +3.4%; on InternVL3-2B [54], by +2.3% and +2.1%, in both cases matching clean accuracy.

Table 3: **Matched baselines, multiple seeds, and generalization.** All methods are trained on MMRL30k with the same budget. Three-seed results report mean±std. Unless noted, the backbone is Qwen3-VL-8B.

Setting	Method	Clean	Seen	Unseen
Matched baselines	GRPO	68.9	59.2	54.0
	NoisyRollout	68.3	59.5	54.5
	NoisyRollout + reward filter	69.0	60.0	55.1
	ROMA	68.7	**61.6**	**56.3**
Three seeds	GRPO	68.8±0.1	59.0±0.3	53.8±0.2
	ROMA	68.6±0.2	**61.5**±0.2	**56.5**±0.3
RL algorithm: DAPO	DAPO	70.1	61.1	55.2
	DAPO + ROMA	69.9	**63.5**	**58.6**
Backbone: InternVL3-2B	Base	50.8	43.6	41.9
	GRPO	53.1	44.6	42.7
	ROMA	53.0	**46.9**	**44.8**

Table 4: **Real-world degraded benchmarks and training cost.** Benchmarks use their official metrics. Training cost is measured on the 8B model for 120 steps.

	Real-world degraded benchmarks			Training cost	
Method	**VizWiz**	**DocVQA**	**OCRBench**	**s/step**	**GPU-h**
Base	56.0	88.8	74.5	–	–
GRPO	57.3	91.2	75.2	130.3	34.7
ROMA	**58.8**	**93.5**	**77.4**	149.8	39.9

Evidence of Reward Poisoning. To isolate the reward channel, we run GRPO with each image replaced by a degraded copy before rollout generation, keeping the policy update conditioned on the clean image. Degraded rollouts lower the mean reward from 0.742 to 0.631 and nearly double the fraction of all-wrong groups, which yield zero GRPO advantage, from 13.4% to 24.8% (Appendix A.3). These prompts are lost from the learning signal, which ROMA's clean-rollout / degraded-view decomposition avoids.

Real-World Degradations and Training Cost. On benchmarks with naturally degraded inputs, VizWiz [9] (mobile-phone photos), DocVQA [30] (scanned documents), and OCRBench [25] (OCR-heavy images), ROMA outperforms GRPO by +1.5, +2.3, and +2.2 points (Table 4). Since ROMA samples no extra rollouts, generation, which dominates RL training cost, is unchanged; the teacher-forced passes add about 15% training time (Table 4). ROMA also retains the corruption awareness of GRPO (84.9% vs. 85.1% detection accuracy; Appendix A.4), suggesting that it reasons through degradation rather than ignoring it.

4.4 Ablation Studies

To address Question 3 and validate our design configurations, we conduct a series of ablation studies. Specifically, we analyze the choice of multi-view optimization strategy, the effect of the auxiliary policy gradient loss, and the necessity of correctness conditioning.

Choice of Multi-View Optimization. We first conduct an ablation study evaluating the worst-case formulation (Eq. 3) for handling multi-view augmentations. Using the 8B model, we ablate this objective by replacing the worst-case penalty with a mean penalty across the augmented views. As detailed in Table 5, adopting this mean strategy incurs an average performance drop of 1.6% on seen degradations and 1.8% on unseen degradations compared to the worst-case formula. This demonstrates that averaging the invariance penalty is insufficient for securing robustness. Instead, by actively penalizing the hardest adversarial view during each optimization step, the worst-case formulation effectively forces the model to learn more robust reasoning capacity.

Ablation on Auxiliary Policy Gradient. Next, we evaluate the contribution of the auxiliary policy gradient (PG) loss by removing it from the overall objective, with the 8B model. As shown in Table 6, omitting this component reduces average accuracy by 1.6% on seen degradations and 1.8% on unseen

Table 5: **Ablation on multi-view optimization strategy.** We evaluate the worst-case formulation by comparing it against averaging the invariance penalty across augmented views. Adopting the mean strategy incurs a performance drop for both seen and unseen degradations, demonstrating that actively penalizing the hardest adversarial view is better for robust reasoning.

Method	MathVista	WeMath	ChartQA	LogicVista	MMStar	VisPuzzles	RealWorldQA	Avg
Seen Degradations								
Mean Penalty	72.1	75.1	49.0	53.5	64.6	40.9	65.1	60.0
Worst-Case Penalty	73.3	77.3	49.1	57.5	66.3	41.7	66.0	61.6
Unseen Degradations								
Mean Penalty	63.2	67.2	44.0	47.5	60.1	37.8	62.0	54.5
Worst-Case Penalty	64.8	70.4	44.1	50.8	62.1	38.0	63.6	56.3

degradations. This indicates that relying solely on the token-level invariance penalty is restrictive. While the invariance penalty successfully anchors the degraded output to the clean reference, it does not provide a sufficient learning signal to actively solve the reasoning task under visual occlusion. The auxiliary PG loss is therefore beneficial to provide a direct gradient signal that guides the policy toward correct reasoning steps despite the noise.

Table 6: **Ablation on the auxiliary policy gradient loss.** We evaluate the framework's performance after removing the auxiliary PG component. Results indicate that integrating this loss improves reasoning accuracy across both seen and unseen degradations by providing a direct learning signal under visual occlusion.

Method	MathVista	WeMath	ChartQA	LogicVista	MMStar	VisPuzzles	RealWorldQA	Avg
Seen Degradations								
w/o Auxiliary PG	72.0	76.9	48.5	55.1	65.3	40.2	65.5	60.5
Full Approach	73.3	77.3	49.1	57.5	66.3	41.7	66.0	61.6
Unseen Degradations								
w/o Auxiliary PG	63.9	69.5	43.6	49.4	60.8	37.9	62.6	55.4
Full Approach	64.8	70.4	44.1	50.8	62.1	38.0	63.6	56.3

Effect of Correctness Conditioning. Finally, we investigate the role of correctness conditioning within the token-level KL penalty using the 8B model. As detailed in Table 8, enforcing the invariance penalty unconditionally, forcing the degraded reasoning trajectory to match the clean trajectory regardless of whether the clean rationale is correct, causes an average performance drop of 2.2% on both seen and unseen degradations. By conditioning the penalty on the objective correctness of the clean rollout, our approach ensures that the policy learns to protect valid reasoning paths, effectively preventing the propagation of erroneous logic during the optimization process.

Table 7: Sensitivity analysis of the auxiliary coefficient α on seen and unseen degradations. The best performance is achieved at $\alpha = 0.10$.

Auxiliary Coefficient	Seen	Unseen
$\alpha = 0.05$	60.5	55.2
$\alpha = 0.10$	61.6	56.3
$\alpha = 0.15$	60.0	54.8

Table 8: **Ablation on correctness conditioning.** We evaluate the necessity of conditioning the invariance penalty on the objective correctness of the clean trajectory. Results demonstrate that strictly applying the penalty to valid reasoning paths prevents the propagation of erroneous logic and improves overall performance.

Method	MathVista	WeMath	ChartQA	LogicVista	MMStar	VisPuzzles	RealWorld	Avg
Seen Degradations								
Unconditional Penalty	69.8	74.7	47.9	52.6	65.0	40.0	65.5	59.4
Correctness-Conditioned	73.3	77.3	49.1	57.5	66.3	41.7	66.0	61.6
Unseen Degradations								
Unconditional Penalty	61.8	67.0	43.2	48.2	59.8	36.8	62.1	54.1
Correctness-Conditioned	64.8	70.4	44.1	50.8	62.1	38.0	63.6	56.3

4.5 Sensitivity Analysis

We conduct a series of sensitivity analyses on the key hyperparameters with the 8B model, including the auxiliary policy gradient coefficient α, the number of augmented views K, and the invariance penalty weight β.

Auxiliary Coefficient. We evaluate the framework's sensitivity to the auxiliary policy gradient loss by varying the coefficient $\alpha \in \{0.05, 0.10, 0.15\}$. As presented in Table 7, the model achieves best performance at $\alpha = 0.10$, yielding 61.6% and 56.3% accuracy on seen and unseen degradations, respectively. Decreasing the coefficient to $\alpha = 0.05$ provides insufficient auxiliary guidance, resulting in a performance drop. Conversely, increasing α to 0.15 also leads to a reduction, as the excessively weighted auxiliary loss begins to over-regularize and interfere with the primary optimization objective.

Number of Augmented Views. The parameter K dictates the diversity of perturbations evaluated during the optimization step. As shown in Table 9, increasing K from 1 to 3 yields steady improvements in both seen and unseen robustness, as the policy is penalized against a broader distribution of visual noise. However, increasing K beyond 3 (e.g., $K = 4$) slightly degrades performance. Because the penalty is β times the largest divergence among the K views, adding a view can only raise (never lower) the selected divergence; a larger K thus implicitly strengthens the regularization, much like increasing β, which over-constrains the policy (Table 10). Therefore, we select $K = 3$ as the default setting to maintain a computationally efficient training pipeline without sacrificing robust generalization.

Table 9: Sensitivity analysis of the number of augmented views K on seen and unseen degradations. We select $K = 3$ as the default setting.

Number of Augmented Views	Seen	Unseen
$K = 1$	59.5	54.4
$K = 2$	60.7	55.2
$K = 3$	61.6	56.3
$K = 4$	61.3	56.0

Invariance Penalty Weight. We investigate the trade-off between baseline reasoning capacity and visual robustness by varying $\beta \in \{0.05, 0.10, 0.15\}$ (Table 10). A high penalty weight (e.g., $\beta = 0.15$) overly constrains the policy, forcing it to prioritize structural matching over exploratory problem-solving, which leads to a performance drop. Conversely, a small weight fails to enforce sufficient noise resilience. We find that $\beta = 0.10$ establishes an optimal balance, maximizing robustness without degrading foundational reasoning capabilities.

Table 10: Sensitivity analysis of the penalty weight β on seen and unseen degradations. The best performance is achieved at $\beta = 0.10$.

Penalty Weight	Seen	Unseen
$\beta = 0.05$	59.4	55.3
$\beta = 0.10$	61.6	56.3
$\beta = 0.15$	56.8	54.6

5 Conclusions

We presented ROMA, an RL fine-tuning framework that reinforces multimodal reasoning against visual degradation without sampling rollouts from degraded inputs. Clean rollouts supply trajectories and rewards, while teacher-forced degraded views drive a correctness-conditioned, worst-case token-level invariance penalty and an auxiliary policy gradient anchored to clean advantages. ROMA consistently improves robustness to seen and unseen corruptions while preserving clean accuracy, and its gains hold across seeds, RL algorithms, backbones, and real-world degraded images at a modest training overhead. Please refer to a discussion on future work in Appendix A.6.

References

[1] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

[2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[3] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

[4] Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and Feng Zhao. Are we on the right way for evaluating large vision-language models?, 2024. URL https://arxiv.org/abs/2403.20330.

[5] Runpeng Dai, Linfeng Song, Haolin Liu, Zhenwen Liang, Dian Yu, Haitao Mi, Zhaopeng Tu, Rui Liu, Tong Zheng, Hongtu Zhu, and Dong Yu. CDE: Curiosity-driven exploration for efficient reinforcement learning in large language models. In *International Conference on Learning Representations (ICLR)*, 2026.

[6] Runpeng Dai, Tong Zheng, Rui Liu, Chengsong Huang, and Hongtu Zhu. Small RL controller, large language model: RL-guided adaptive sampling for test-time scaling. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2026.

[7] Yihe Deng, Hritik Bansal, Fan Yin, Nanyun Peng, Wei Wang, and Kai-Wei Chang. Openvlthinker: Complex vision-language reasoning via iterative sft-rl cycles, 2025. URL https://arxiv.org/abs/2503.17352.

[8] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[9] Danna Gurari, Qing Li, Abigale J. Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P. Bigham. VizWiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3608–3617, 2018.

[10] Nicklas Hansen and Xiaolong Wang. Generalization in reinforcement learning by soft data augmentation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13611–13617. IEEE, 2021.

[11] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

[12] Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaosheng Cao, Zheyu Ye, Fei Zhao, Zhe Xu, Yao Hu, and Shaohui Lin. Vision-r1: Incentivizing reasoning capability in multimodal large language models. *arXiv preprint arXiv:2503.06749*, 2025.

[13] Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, et al. T\" ulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*, 2024.

[14] Misha Laskin, Kimin Lee, Adam Stooke, Lerrel Pinto, Pieter Abbeel, and Aravind Srinivas. Reinforcement learning with augmented data. *Advances in neural information processing systems*, 33:19884–19895, 2020.

[15] Zongxia Li, Wenhao Yu, Chengsong Huang, Rui Liu, Zhenwen Liang, Fuxiao Liu, Jingxi Chen, Dian Yu, Jordan Boyd-Graber, Haitao Mi, and Dong Yu. Self-rewarding vision-language model via reasoning decomposition. In *International Conference on Learning Representations (ICLR)*, 2026.

[16] Haolin Liu, Dian Yu, Sidi Lu, Yujun Zhou, Rui Liu, Zhenwen Liang, Haitao Mi, Chen-Yu Wei, and Dong Yu. Save the good prefix: Precise error penalization via process-supervised RL to enhance LLM reasoning. In *Findings of the Association for Computational Linguistics: ACL 2026*, 2026.

[17] Rui Liu, Guangyao Shi, and Pratap Tokekar. Data-driven distributionally robust optimal control with state-dependent noise. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023.

[18] Rui Liu, Yu Shen, Peng Gao, Pratap Tokekar, and Ming Lin. CAML: Collaborative auxiliary modality learning for multi-agent systems. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

[19] Rui Liu, Peng Gao, Yu Shen, Ming Lin, and Pratap Tokekar. Adaptive conformal guidance for learning under uncertainty. In *International Conference on Learning Representations (ICLR)*, 2026.

[20] Rui Liu, Pratap Tokekar, and Ming Lin. Active asymmetric multi-agent multimodal learning under uncertainty. *arXiv preprint arXiv:2602.04763*, 2026.

[21] Rui Liu, Dian Yu, Lei Ke, Haolin Liu, Yujun Zhou, Zhenwen Liang, Haitao Mi, Pratap Tokekar, and Dong Yu. Stable and efficient single-rollout RL for multimodal reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

[22] Rui Liu, Dian Yu, Zhenwen Liang, Yucheng Shi, Tong Zheng, Runpeng Dai, Haitao Mi, Pratap Tokekar, and Leoweiliang. DeltaRubric: Generative multimodal reward modeling via joint planning and verification. *arXiv preprint arXiv:2605.09269*, 2026.

[23] Rui Liu, Dian Yu, Tong Zheng, Runpeng Dai, Zongxia Li, Wenhao Yu, Zhenwen Liang, Linfeng Song, Haitao Mi, Pratap Tokekar, and Dong Yu. Dual-uncertainty guided policy learning for multimodal reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2026*, 2026.

[24] Xiangyan Liu, Jinjie Ni, Zijian Wu, Chao Du, Longxu Dou, Haonan Wang, Tianyu Pang, and Michael Qizhe Shieh. Noisyrollout: Reinforcing visual reasoning with data augmentation. *arXiv preprint arXiv:2504.13055*, 2025.

[25] Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. OCRBench: On the hidden mystery of OCR in large multimodal models. *Science China Information Sciences*, 67(12):220102, 2024.

[26] Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*, 2023.

[27] Trung Quoc Luong, Xinbo Zhang, Zhanming Jie, Peng Sun, Xiaoran Jin, and Hang Li. Reft: Reasoning with reinforced fine-tuning. *arXiv preprint arXiv:2401.08967*, 2024.

[28] Guozheng Ma, Zhen Wang, Zhecheng Yuan, Xueqian Wang, Bo Yuan, and Dacheng Tao. A comprehensive survey of data augmentation in visual reinforcement learning. *International Journal of Computer Vision*, 133(10):7368–7405, 2025.

[29] Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. *arXiv preprint arXiv:2203.10244*, 2022.

[30] Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. DocVQA: A dataset for VQA on document images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2200–2209, 2021.

[31] Alhassan Mumuni, Fuseini Mumuni, and Nana Kobina Gerrar. A survey of synthetic data augmentation methods in machine vision. *Machine Intelligence Research*, 21(5):831–869, 2024.

[32] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

[33] Yingzhe Peng, Gongrui Zhang, Miaosen Zhang, Zhiyuan You, Jie Liu, Qipeng Zhu, Kai Yang, Xingzhong Xu, Xin Geng, and Xu Yang. Lmm-r1: Empowering 3b lmms with strong reasoning abilities through two-stage rule-based rl. *arXiv preprint arXiv:2503.07536*, 2025.

[34] Runqi Qiao, Qiuna Tan, Guanting Dong, Minhui Wu, Chong Sun, Xiaoshuai Song, Zhuoma GongQue, Shanglin Lei, Zhe Wei, Miaoxuan Zhang, et al. We-math: Does your large multimodal model achieve human-like mathematical reasoning? *arXiv preprint arXiv:2407.01284*, 2024.

[35] 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*, pages 8748–8763. PmLR, 2021.

[36] Roberta Raileanu, Max Goldstein, Denis Yarats, Ilya Kostrikov, and Rob Fergus. Automatic data augmentation for generalization in deep reinforcement learning. *arXiv preprint arXiv:2006.12862*, 2020.

[37] Fawaz Sammani, Boris Joukovsky, and Nikos Deligiannis. Visualizing and understanding contrastive learning. *IEEE Transactions on Image Processing*, 33:541–555, 2023.

[38] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[39] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[40] Yueqi Song, Tianyue Ou, Yibo Kong, Zecheng Li, Graham Neubig, and Xiang Yue. Visualpuzzles: Decoupling multimodal reasoning evaluation from domain knowledge. *arXiv preprint arXiv:2504.10342*, 2025.

[41] Yi Su, Dian Yu, Linfeng Song, Juntao Li, Haitao Mi, Zhaopeng Tu, Min Zhang, and Dong Yu. Crossing the reward bridge: Expanding rl with verifiable rewards across diverse domains. *arXiv preprint arXiv:2503.23829*, 2025.

[42] Huajie Tan, Yuheng Ji, Xiaoshuai Hao, Minglan Lin, Pengwei Wang, Zhongyuan Wang, and Shanghang Zhang. Reason-rft: Reinforcement fine-tuning for visual reasoning. *arXiv preprint arXiv:2503.20752*, 2025.

[43] Qwen Team. Qwen2.5: A party of foundation models, September 2024. URL `https://qwenlm.github.io/blog/qwen2.5/`.

[44] Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhu Chen. Vl-rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. *arXiv preprint arXiv:2504.08837*, 2025.

[45] Zhenhailong Wang, Xuehang Guo, Sofia Stoica, Haiyang Xu, Hongru Wang, Hyeonjeong Ha, Xiusi Chen, Yangyi Chen, Ming Yan, Fei Huang, et al. Perception-aware policy optimization for multimodal reasoning. *arXiv preprint arXiv:2507.06448*, 2025.

[46] xAI. Grok-1.5 Vision Preview, 2024. URL `https://x.ai/news/grok-1.5v`.

[47] Yijia Xiao, Edward Sun, Tianyu Liu, and Wei Wang. Logicvista: Multimodal llm logical reasoning benchmark in visual contexts. *arXiv preprint arXiv:2407.04973*, 2024.

[48] Yi Yang, Xiaoxuan He, Hongkun Pan, Xiyan Jiang, Yan Deng, Xingtao Yang, Haoyu Lu, Dacheng Yin, Fengyun Rao, Minfeng Zhu, et al. R1-onevision: Advancing generalized multimodal reasoning through cross-modal formalization. *arXiv preprint arXiv:2503.10615*, 2025.

[49] Huanjin Yao, Qixiang Yin, Jingyi Zhang, Min Yang, Yibo Wang, Wenhao Wu, Fei Su, Li Shen, Minghui Qiu, Dacheng Tao, et al. R1-sharevl: Incentivizing reasoning capability of multimodal large language models via share-grpo. *arXiv preprint arXiv:2505.16673*, 2025.

[50] Denis Yarats, Ilya Kostrikov, and Rob Fergus. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. In *International conference on learning representations*, 2021.

[51] Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, et al. DAPO: An open-source LLM reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

[52] Tong Zheng, Hongming Zhang, Wenhao Yu, Xiaoyang Wang, Xinyu Yang, Runpeng Dai, Rui Liu, Huiwen Bao, Chengsong Huang, Heng Huang, et al. Parallel-R1: Towards parallel thinking via reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2026.

[53] Yaowei Zheng, Junting Lu, Shenzhi Wang, Zhangchi Feng, Dongdong Kuang, and Yuwen Xiong. Easyr1: An efficient, scalable, multi-modality rl training framework, 2025. URL `https://github.com/hiyouga/EasyR1`.

[54] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, et al. InternVL3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

[55] Linghao Zhu, Yiran Guan, Dingkang Liang, Jianzhong Ju, Zhenbo Luo, Bin Qin, Jian Luan, Yuliang Liu, and Xiang Bai. Shuffle-r1: Efficient rl framework for multimodal large language models via data-centric dynamic shuffle. *arXiv preprint arXiv:2508.05612*, 2025.

A Appendix

A.1 Implementation Details

We train all models on the MMRL30k dataset [55], which contains around 30K samples. The models are trained to generate responses in a structured format, where the reasoning process is enclosed within `<thinking></thinking>` tags and the final answer is presented in `\boxed{}`. The training is performed for 120 steps with a learning rate of $1e-6$ and a weight decay of 0.01. We adopt a global batch size of 128, a rollout batch size of 256, and generate 8 rollouts per input with a rollout temperature 1.0. The implementation builds on the EasyR1 framework [53].

A.2 Degradation Details and Severity Levels

To rigorously evaluate the robustness of our approach, we apply a diverse set of visual corruptions. During training, parameters are sampled continuously according to corresponding distributions. During evaluation, we utilize three severity levels (Level 1 to Level 3) for benchmarking, following the ImageNet-C framework [11]. Crucially, Level 3 is designed to evaluate OOD magnitude generalization. For every degradation type, Level 3 applies a severity that strictly exceeds the bounds of the parameter distribution encountered by the model during training. The parameter configurations for the degradations are detailed below in Table 11 and qualitative examples are provided in Figure 3.

Table 11: Parameter configurations for visual degradations during training and evaluation. For JPEG Quality, Resolution Scale, Posterization, and Pixelation, a lower value indicates higher severity. For the seen pool, Level 3 consistently falls outside the bounds of the training distribution (magnitude OOD). The unseen pool is strictly OOD across all levels.

Degradation Type	**Parameter**	**Training Distribution**	**Eval Level 1**	**Eval Level 2**	**Eval Level 3**
		Seen Degradations			
Gaussian Noise	Std. Dev. (σ)	0.05	0.03	0.06	$\mathbf{0.12}$
Gaussian Blur	Radius (r)	$\mathcal{U}(0.5, 2.0)$	1.0	2.0	$\mathbf{3.5}$
JPEG Compression	Quality (q) $\downarrow$	$\mathcal{U}\{30, 85\}$	65	40	$\mathbf{15}$
Resolution Scale	Scale Factor (f) $\downarrow$	$\mathcal{U}(0.3, 0.7)$	0.6	0.4	$\mathbf{0.2}$
		Unseen Degradations			
Motion Blur	Kernel Size (k)	(Held Out)	5	15	25
Salt & Pepper Noise	Probability (p)	(Held Out)	0.02	0.05	0.10
Speckle Noise	Std. Dev. (σ)	(Held Out)	0.06	0.12	0.25
Posterization	Bit Depth (b) $\downarrow$	(Held Out)	6	4	2
Pixelation	Scale Factor (f) $\downarrow$	(Held Out)	0.5	0.25	0.12

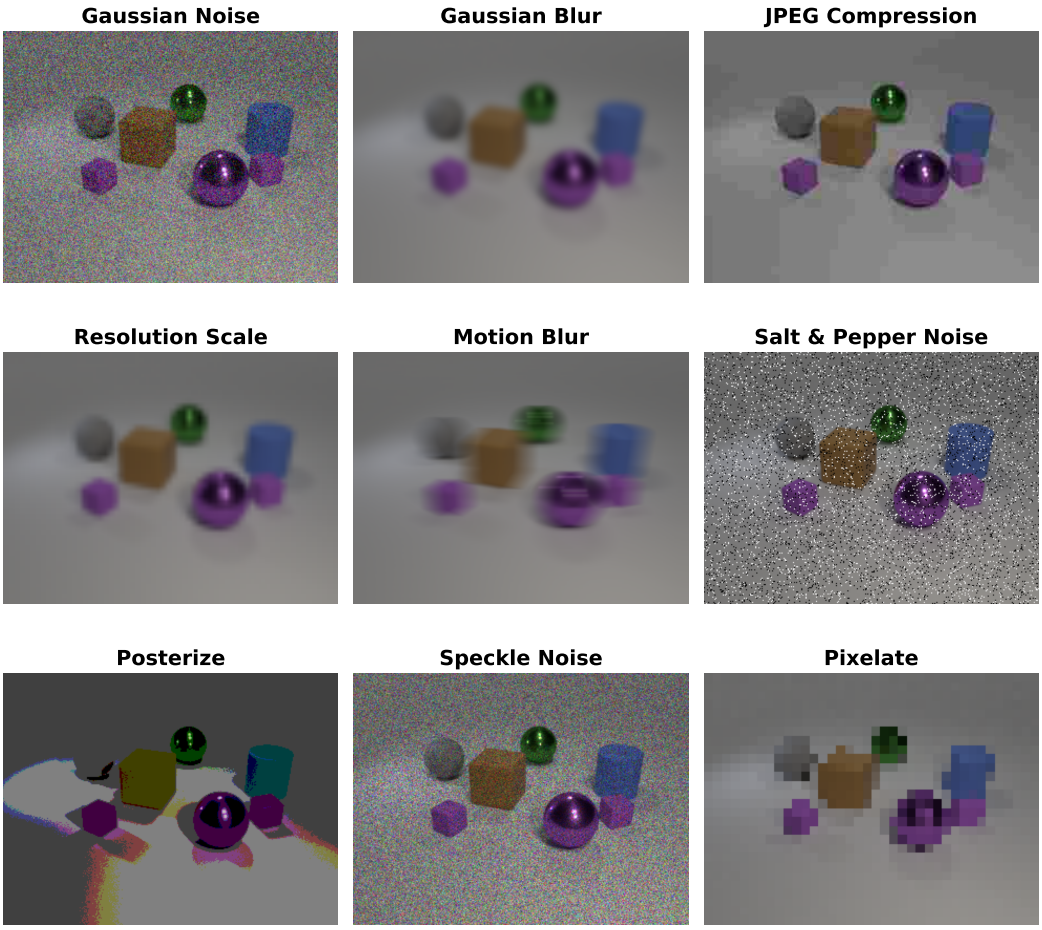


Figure 3: **Qualitative examples of visual degradations.** The perturbations are categorized into seen degradations (e.g., Gaussian noise, Gaussian blur, JPEG compression, and Resolution downscaling), which simulate common transmission artifacts encountered during training, and unseen degradations (e.g., motion blur, salt-and-pepper noise, speckle noise, posterization, and pixelation), which test out-of-distribution (OOD) generalization against held-out noise structures.

A.3 Reward-Channel Analysis

To isolate the reward channel, we run a degraded-rollout control that is identical to the GRPO baseline in every respect but one: before rollout generation, each image is replaced by a degraded copy, so all rollouts of a prompt are sampled under the same degraded image and the group's rewards and advantages come entirely from the degraded input. The policy update is still conditioned on the clean image, as in GRPO, so the reward signal is the only thing that changes. Table 12 reports two quantities averaged over 120 training steps: the mean verifier reward per rollout, and the fraction of prompt groups whose rollouts all fail (reward 0) and therefore yield zero advantage under GRPO. Computing rewards from degraded images lowers the mean reward and nearly doubles the all-wrong rate, removing these prompts from the learning signal. ROMA's clean-rollout / degraded-view decomposition avoids this by construction.

Table 12: GRPO reward statistics when rollouts are sampled from clean vs. degraded images, averaged over 120 training steps.

Rollout Input	Mean Reward	All-Wrong Groups (%)
Clean (GRPO)	0.742	13.4
Degraded	0.631	24.8

A.4 Corruption Awareness

Invariance is not unconditionally the right target: when degradation destroys the evidence needed to answer, a model should not pretend the image is clean. ROMA is designed to avoid enforcing invariance indiscriminately. Both the invariance penalty and the auxiliary objective apply only to trajectories that are correct on the clean input, and the invariance term is a soft regularizer, so the main RL objective remains free to adapt the reasoning to degraded evidence. To test whether ROMA nevertheless becomes blind to corruption, we prompt the Base 8B, GRPO, and ROMA models to inspect image quality before answering, using the following instruction:

```
You FIRST inspect the image quality, then answer the question.  Step
1:  State whether the image is corrupted.  Write exactly 'IMAGE:
corrupted' or 'IMAGE: clean'.  Step 2:  If corrupted, name the
most likely corruption type as 'TYPE: <type>'.  If clean, write
'TYPE: none'.  Step 3:  Answer the question.  Put the final answer in
\boxed{}.
```

As shown in Table 13, ROMA achieves corruption-detection and corruption-type accuracy comparable to GRPO, so its improved robustness does not come from ignoring the corruption. Table 14 shows a representative example. All three models report the blur, but Base and GRPO attribute the haze to the blur and revise their answer from gray to blue-gray, letting the corruption move the answer. ROMA detects the same blur and recognizes the resulting desaturation, yet still infers the underlying color correctly, suggesting that it implicitly reasons through the corruption rather than suppressing it.

Table 13: Corruption awareness of the 8B models when prompted to inspect image quality before answering.

Method	Corruption Detection Acc. (%)	Corruption-Type Acc. (%)
Base	84.0	61.0
GRPO	85.1	62.3
ROMA	84.9	62.0

A.5 Qualitative Examples

Table 15 compares GRPO and ROMA on clean and blurred versions of the same images. Both models answer correctly on the clean images, but GRPO fails under blur while ROMA remains correct. In Case 1, both models reject the ski-resort option on the clean image because no ski lifts are visible;

Table 14: Corruption-aware example. *Question:* What is the dominant color of the ocean and sky in the image? (A) Bright blue (B) Blue-gray (C) Gray (D) Green. *Ground truth:* C. All models answer C on the clean image and report `IMAGE: clean, TYPE: none`. Responses on the blurred image are abridged.

Model	**Blurred input**	**Response excerpt (blurred input)**
Base	corrupted / blur; B (×)	"The image is blurred, making it difficult to discern fine details. However, the dominant color of the ocean and sky appears to be a muted, hazy tone, likely gray or blue-gray, due to the atmospheric conditions or the blur effect. [...] Blue-gray (B) is the most plausible description."
GRPO	corrupted / blur; B (×)	"The image is blurred, making fine details indistinct. However, the dominant color of the ocean and sky appears to be a muted, hazy tone, likely gray or blue-gray, due to atmospheric conditions or the blur effect. [...] Blue-gray best matches the overall color impression."
ROMA	corrupted / blur; C (✓)	"The image is blurred, making it difficult to discern fine details. However, the overall color palette appears to be dominated by muted, warm tones. The ocean and sky blend into a hazy, uniform color that is best described as gray or a very pale, desaturated blue-gray, rather than bright blue or green."

under blur, GRPO reverses this judgment and reads the mountains and lights as evidence of a ski resort, whereas ROMA still asserts that the ski lifts are absent. In Case 2, the decisive cue is the dancers themselves: under blur, GRPO no longer perceives them and settles on a concert, the option that requires the least visual detail to justify, while ROMA still resolves the figures as dancers.

A.6 Discussions and Future Work

While our proposed approach establishes a robust foundation for multimodal reasoning against degradations, it also opens several promising avenues for future research. A natural progression is to extend this worst-case multi-view optimization paradigm to temporal modalities, such as video-based reasoning. Furthermore, future work could investigate adaptive mechanisms to dynamically weight both the auxiliary policy gradient objective and the invariance penalty based on the inferred severity of the visual degradation, thereby allocating stronger defensive penalties specifically to highly adversarial inputs.

Table 15: Qualitative comparison of GRPO and ROMA on clean and blurred images. Responses are abridged.

Case 1. What is the overall scene depicted in this image? (A) people ice skating on a frozen lake (B) a small town during winter snowfall (C) a busy city street at Christmas (D) a ski resort town with ski lifts visible. *Ground truth:* B.		
GRPO, clean	B (✓)	"There are decorative lights on trees and buildings [. . .] but no visible ski lifts or large open ski areas. [. . .] B: Small town during winter snowfall — this matches the snowy buildings, trees, and quiet street scene."
GRPO, blur	D (×)	"The image is blurry, but I can still make out key elements [. . .] the background has hills or mountains, and the overall ambiance with lights and snow-covered structures is consistent with a ski resort town [. . .] Option D best fits the scene."
ROMA, clean	B (✓)	"There are no visible ice skaters, no frozen lake, and no ski lifts. [. . .] There are snowy mountains, but no ski lifts are visible. [. . .] Option B is the most accurate description of the scene."
ROMA, blur	B (✓)	"There are buildings with lights, decorated trees [. . .] In the background, there are mountains. [. . .] D: There are no visible ski lifts in the image, so this option is less accurate."
Case 2. What is the main event in the image? (A) a theater play (B) a sporting event (C) a concert (D) a dance competition. *Ground truth:* D.		
GRPO, clean	D (✓)	"The performers are dressed in coordinated costumes [. . .] others are standing in expressive dance positions. [. . .] A dance competition fits best because the performers appear to be showcasing choreography."
GRPO, blur	C (×)	"The image is blurry [. . .] There are no visible sports equipment, audience seating typical of a theater, or judging panels that would indicate a dance competition. [. . .] the most fitting option is C: A concert."
ROMA, clean	D (✓)	"They are dressed in coordinated outfits, mostly white, and are posing in dance-like positions. [. . .] The coordinated poses and stage setup strongly suggest a dance competition."
ROMA, blur	D (✓)	"Concerts often feature musicians and singers, but the image shows a group of performers who appear to be dancers or a dance troupe. [. . .] D: A dance competition — This fits well."

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Please see in Section 3 and Section 4.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Please see in Section A.6.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: Please see in Section 4.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Please see in Section 4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The dataset used in this paper is publicly available. The paper has sufficient instructions to reproduce the main experimental results.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Please see in Section 4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report the mean and standard deviation over three random seeds for GRPO and ROMA in Section 4.3 (Table 3).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Please see in Section 4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in the paper conform in every respect with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Please see in Section 1 and Section 5.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Please see in Section 4.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not need IRB approvals.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: We use LLM only for writing, editing, or formatting purposes.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.