Seeing with You: Perception-Reasoning Co-evolution for Multimodal Reasoning

Anonymous ACL submission

Abstract

Reinforcement learning with verifiable rewards (RLVR) has substantially enhanced the reasoning capabilities of multimodal large language models (MLLMs). However, existing RLVR approaches typically rely on outcome-driven optimization that updates both perception and reasoning using a shared reward based solely on the final answer. This coupled supervision blurs credit assignment, frequently improving reasoning patterns while failing to reliably enhance the accuracy of upstream visual evidence extraction. To address this perception bottleneck, we introduce **PRCO** (**P**erception–**R**easoning **CO**-evolution), a dual-role RLVR framework with a unified shared policy. PRCO alternates between two cooperative roles: an Observer that extracts task-relevant visual facts to generate a descriptive caption, and a Solver that predicts the final answer based on this caption. Crucially, PRCO employs role-factorized supervision: the Solver is optimized using verifiable outcome rewards on the final answer, while the Observer receives a utility reward based on the Solver's success. Extensive experiments across eight challenging multimodal reasoning benchmarks demonstrate that PRCO yields consistent improvements across model scales by over 7 points on average accuracy compared to the base model, outperforming prior open-sourced RL-tuned baselines.

1 Introduction

Reinforcement learning with verifiable rewards (RLVR), particularly online algorithms such as Group Relative Policy Optimization (GRPO), has substantially enhanced the reasoning capabilities of Large Language Models (LLMs) in text-only problem domains (Guo et al., 2025; Team et al., 2025; Yang et al., 2025). Building on these advances, recent work extends RLVR to Multimodal Large Language Models (MLLMs) for challenging multimodal reasoning tasks (Huang et al., 2025b; Shen et al., 2025a; Wang et al., 2025a; Zhu et al., 2025; Zheng et al., 2025b; Su et al., 2025; Fan et al., 2025; Cao et al., 2025; Meng et al., 2025). Most prior multimodal RLVR work focuses on data-centric curation (Liu et al., 2025b; Leng et al., 2025; Wu et al., 2025; Li et al., 2025a; Wang et al., 2025d) and reward-centric engineering (Liu et al., 2025c; Wan et al., 2025; Wang et al., 2025b,c). Effective reasoning relies on accurate perception, which provides the necessary grounding for logical deduction (Liu et al., 2025a; Yao et al., 2025b). However, RLVR is often applied in an outcome-driven manner, verifying only the final textual answer while largely neglecting the accuracy of upstream visual perception (Wang et al., 2025e; Li et al., 2025c).

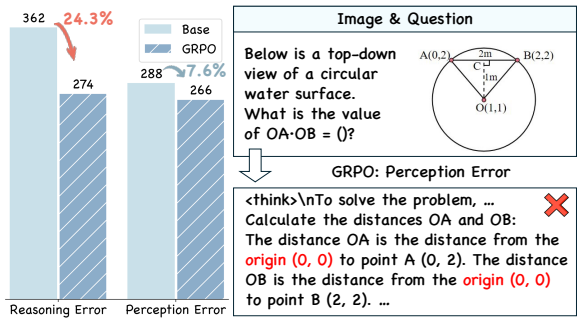


Figure 1: Diagnostic analysis of GRPO on WeMath (Qiao et al., 2025). Left: GRPO reduces reasoning errors much more than perception errors. Right: a representative failure case caused by incorrect perception.

To concretely examine this bottleneck, we conduct a diagnostic analysis using GRPO as a representative multimodal RLVR baseline. We train Qwen2.5-VL-7B (Team, 2025) with GRPO and compare its failure modes against its initialization on WeMath (Qiao et al., 2025) using fine-grained error categorization. As shown in Fig. 1, *training with GRPO substantially reduces reasoning errors, whereas perception errors improve only marginally over the base model.* We attribute this bottleneck to outcome-only RLVR with a shared reward updating

both perception and reasoning. This blurs credit assignment and improves reasoning patterns without reliably improving visual evidence extraction.

Recently, some works have recognized perception as critical and explored perception-centric RLVR for MLLMs. These efforts focus on three directions: introducing additional perception-oriented optimization objectives (Xiao et al., 2025; Wang et al., 2025e), using weighted token-level credit assignment for perception and reasoning tokens (Huang et al., 2025a), requiring an explicit image-description step before reasoning (Xing et al., 2025; Gou et al., 2025). Despite being promising, these works still distribute outcome-based rewards to both perception and reasoning tokens. Consequently, the coupled training signal may improve reasoning patterns without reliably improving visual evidence extraction. These findings naturally prompt the question:

Can we use reliable, separate learning signals for perception and reasoning to decouple them at the gradient level?

To this end, we introduce **PRCO** (**P**erception–**R**easoning **CO**-evolution), a dual-role RLVR framework that disentangles perception and reasoning. In this setup, a unified multimodal policy alternates between two roles: the Observer, which performs question-conditioned evidence extraction by grounding on the image and writing an evidence caption that records the visual facts needed to answer the question; and the Solver, which produces the final answer based on the question and the caption and uses the image when available. Crucially, they are trained with separate and reliable learning signals: the Solver is optimized with verifiable outcome rewards on the final answer, while the Observer is optimized with a utility reward defined by the Solver's verifier-validated success when conditioned on its caption. This role-factorized supervision decouples gradient signals for perception and reasoning while still allowing joint optimization under a unified policy. As a result, PRCO drives co-evolution, in which the Observer progressively improves question-grounded visual perception by extracting and articulating task-relevant evidence, while the Solver learns to reason more reliably under explicit evidence guidance.

To validate PRCO, we conduct extensive experiments on eight challenging multimodal reasoning benchmarks covering mathematics, geometry, logic, and multidisciplinary reasoning. PRCO yields consistent gains across model scales: our 7B model improves average accuracy by 7.18 points over the corresponding base model and outperforms prior open-source RL-tuned methods, while our 3B model improves average accuracy by 7.65 points. Moreover, on WeMath, PRCO reduces perception-related errors by 39.2% relative to the base model, whereas vanilla GRPO achieves only a 7.6% reduction.

To sum up, our main contributions are threefold:

- We propose **PRCO**, a dual-role RLVR framework that disentangles perception and reasoning with an Observer and a Solver under a shared policy.
- We demonstrate the effectiveness of PRCO on diverse and challenging multimodal reasoning benchmarks, showing consistent improvements over strong RLVR baselines.
- We provide extensive ablation and diagnostic analyses that validate PRCO's key design choices and characterize its effects on perception and reasoning.

2 Method

We propose a dual-role RLVR framework that trains a shared multimodal large language model (MLLM) policy in two cooperative roles. The Observer extracts image-grounded visual evidence and produces a question-conditioned evidence caption c, while the Solver answers based on the caption and, when available, the image. We first review Group Relative Policy Optimization (Sec. 2.1), then describe the dual-role interaction loop (Sec. 2.2), detail the Observer and Solver (Sec. 2.3–2.4), and finally present unified optimization with role-specific advantages (Sec. 2.5).

2.1 Preliminary: Group Relative Policy Optimization

Group Relative Policy Optimization (GRPO) (Shao et al., 2024) is a reinforcement learning algorithm for fine-tuning a policy LLM without learning a separate value function. Its key idea is to compute *relative* learning signals by normalizing rewards against other responses sampled from the same prompt. For a given prompt x, the policy generates a group of G complete responses $\{y_i\}_{i=1}^{G}$. Each response receives a scalar reward r_i. GRPO converts these rewards into response-level advantages via

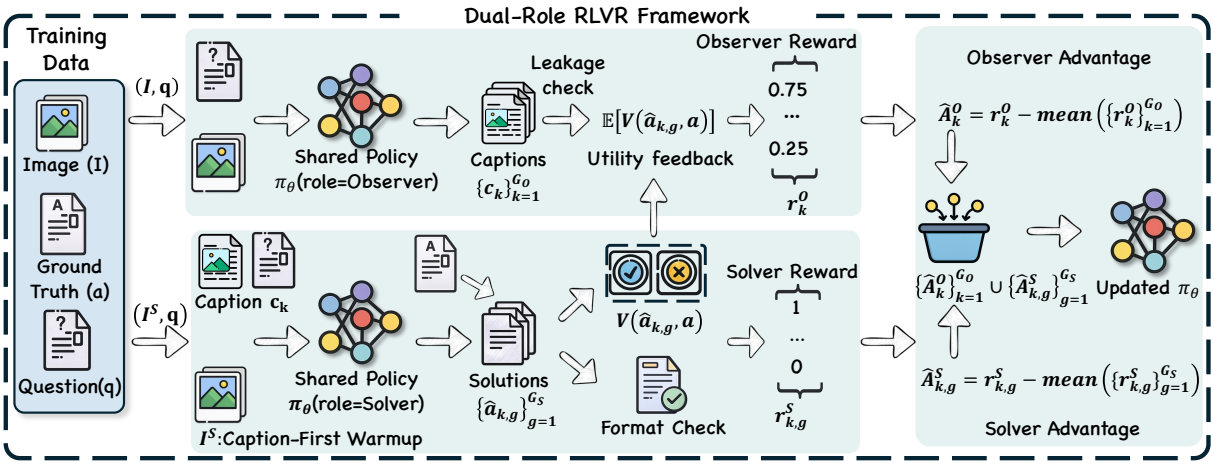


Figure 2: Overview of PRCO. A shared policy alternates between an Observer for question-conditioned evidence caption and a Solver for caption-conditioned reasoning. The two roles are jointly optimized with role-specific feedback and group-relative advantages, enabling perception–reasoning co-evolution under a shared policy.

z-score normalization:

$$\hat{A}_i = \frac{r_i - \text{mean}(\{r_j\}_{j=1}^{G})}{\text{std}(\{r_j\}_{j=1}^{G}) + \epsilon_{\text{norm}}}, \tag{1}$$

where ϵ_{norm} is a small constant for numerical stability.

Policy update. GRPO updates the policy using a PPO-style clipped surrogate objective to improve stability. To prevent excessive policy drift, the objective is regularized with a KL-divergence penalty to the old policy:

$$\begin{aligned}\mathcal{L}_{\text{GRPO}}(\theta) = -\,\mathbb{E}_{i,t}\Big[\min\Big(\rho_{i,t}(\theta)\hat{A}_i,\\ \text{clip}\big(\rho_{i,t}(\theta), 1-\epsilon, 1+\epsilon\big)\,\hat{A}_i\Big)\Big]\\ +\,\beta\,\mathbb{E}_x\Big[\text{KL}\big(\pi_\theta(\cdot|x)\,\|\,\pi_{\theta_{\text{old}}}(\cdot|x)\big)\Big].\end{aligned} \tag{2}$$

where $\rho_{i,t}(\theta)$ is the token-level importance ratio, ϵ is the clipping threshold, and β controls the KL penalty. Optimizing this objective increases the likelihood of responses with positive relative advantages, while the KL term constrains divergence from $\pi_{\theta_{\text{old}}}$ for stable training.

2.2 Overview

RLVR setting. Following typical RLVR setups (Wang et al., 2025a; Yu et al., 2025), our training dataset $\mathcal{D}$ consists of tuples (I, q, a) where I is an image, q is a question, and a is a short ground-truth answer. We do not rely on any existing chain-of-thought data and initiate RL training directly without supervised fine-tuning. We use a lightweight rule-based verifier $V(\hat{a}, a) \in \{0, 1\}$ that checks whether the predicted answer $\hat{a}$ matches a, and a simple format checker $\text{FormatScore}(\hat{a}) \in [0, 1]$ that evaluates whether the output satisfies the required format.

Two roles under one shared policy. We instantiate a single policy π_θ in two roles via role-specific prompting. We denote by r_O and r_S the prompts for the Observer and Solver, respectively. For each sample (I, q, a), the Observer first generates an intermediate caption c summarizing question-relevant visual evidence; then the Solver outputs the final answer conditioned on c and optionally the image. We denote the Solver's visual input as $I^S \in \{\emptyset, I\}$. For each training instance, we sample G_O candidate captions $\{c_k\}_{k=1}^{G_O}$ under the Observer role; given a caption c_k, we sample G_S candidate answers $\{\hat{a}_{k,g}\}_{g=1}^{G_S}$ under the Solver role. The Observer is encouraged to externalize visual evidence into captions, while the Solver is trained to leverage captions for evidence-conditioned reasoning.

2.3 Observer: Utility-Driven Evidence Captioning

The Observer converts high-dimensional visual input into a textual evidence signal by producing a question-conditioned caption that summarizes the visual facts most relevant to q (e.g., entities, attributes, counts, and relations). Formally, given (I, q), the Observer samples $c \sim \pi_\theta(\cdot \mid I, q, r_O)$.

Utility reward with leakage suppression. Directly verifying the intrinsic quality of an evidence caption is difficult. We therefore train the Observer through the downstream utility that its caption pro-

vides to the Solver. A key failure mode is answer shortcutting, where the Observer directly places the final answer in the caption instead of extracting task-relevant visual evidence. To suppress this behavior, we use an auxiliary LLM-based leakage checker that takes the caption c and question q as input and outputs a binary leakage indicator. Let $\mathbb{I}_{\text{leak}}(q, c) \in \{0, 1\}$ be the indicator of answer leakage in c, where 1 indicates the presence of leakage. For a sampled caption c_k, we define the Observer reward as

$$r_k^O = \big(1 - \mathbb{I}_{\text{leak}}(q, c_k)\big)\, \mathbb{E}_{\hat{a} \sim \pi_\theta(\cdot \mid I^S, q, c_k, r_S)} \left[V(\hat{a}, a) \right]. \tag{3}$$

The expectation is approximated by the empirical mean of the verifier scores over G_S sampled Solver rollouts conditioned on (I^S, q, c_k, r_S). This reward favors captions that help downstream solving and suppresses captions judged as leaking the answer.

2.4 Solver: Evidence-Conditioned Multimodal Reasoning

The Solver produces a short final answer by reasoning over the question and the Observer's caption, with the image provided when available. Conditioning on c encourages explicit evidence-driven reasoning, while image input helps recover global structure or complex geometric relations that are difficult to fully convey in text. Formally, given a caption c, the Solver samples $\hat{a} \sim \pi_\theta(\cdot \mid I^S, q, c, r_S)$.

Solver reward. We define the correctness reward via the verifier as $r^{\text{acc}} = V(\hat{a}, a)$. In addition, $r^{\text{format}} \in [0, 1]$ measures whether the response strictly follows the required format. We compute it with a simple rule-based checker as $r^{\text{format}} = \text{FormatScore}(\hat{a})$. The Solver is rewarded for both correctness and basic format compliance:

$$r^S = \lambda\, r^{\text{acc}} + (1 - \lambda)\, r^{\text{format}}, \tag{4}$$

where λ balances accuracy and format compliance, with a default value of 0.9.

Caption-first warmup. In early training, we set $I^S = \emptyset$ so that the Solver must rely on (q, c). We find that if the Solver receives both the image and the caption too early, it tends to ignore the caption and solve directly from the image, which can drown out the learning signal for improving captions. Therefore, we first warm up the Solver to solve using only captions; after a short warmup, we switch to $I^S = I$ to restore full multimodal grounding while retaining the benefits of caption conditioning.

2.5 Unified Policy Optimization with Role-Specific Advantages

The Observer and the Solver share the same parameters and are optimized jointly. However, they induce different reward distributions and natural groupings: captions compete within the same (I, q) instance, whereas answers compete within groups conditioned on a particular caption. We therefore compute group-relative advantages separately for each role and perform a single GRPO update over the union of all role trajectories.

Role-wise grouping and advantage computation. For each sample (I, q, a), the Observer generates G_O candidate captions $\{c_k\}_{k=1}^{G_O}$ under $\pi_\theta(\cdot \mid I, q, r_O)$. For each caption c_k, we generate G_S Solver answers $\{\hat{a}_{k,g}\}_{g=1}^{G_S}$ under (I^S, q, c_k, r_S) and compute rewards $\{r_{k,g}^S\}_{g=1}^{G_S}$ using Eq. (4). We then compute Observer rewards $\{r_k^O\}_{k=1}^{G_O}$ via Eq. (3). Following Eq. (1), we compute group-relative advantages separately for the two roles while omitting the standard deviation normalization term. Concretely, for a reward group $\{r_i\}_{i=1}^{G}$, we use $\hat{A}_i = r_i - \text{mean}(\{r_j\}_{j=1}^{G})$. By centering rewards around the role-specific group mean without standard deviation normalization, we ensure that gradient updates are driven by within-group relative performance rather than cross-role variance differences. We compute caption advantages $\{\hat{A}_k^O\}_{k=1}^{G_O}$ from $\{r_k^O\}_{k=1}^{G_O}$ across the G_O captions. For the Solver update, we reuse these caption-conditioned answer rollouts. For each (I, q, a), we preferentially sample one caption index $\tilde{k}$ uniformly from captions with $\text{Var}(\{r_{k,g}^S\}_{g=1}^{G_S}) > 0$ to avoid degenerate relative signals, and compute Solver advantages $\{\hat{A}_{\tilde{k},g}^S\}_{g=1}^{G_S}$ from $\{r_{\tilde{k},g}^S\}_{g=1}^{G_S}$.

Unified policy update. We aggregate Observer caption trajectories (with $\hat{A}^O$) and Solver answer trajectories (with $\hat{A}^S$) into one rollout batch and optimize a dual-role GRPO objective:

$$\mathcal{L}_{\text{dual}}(\theta) = \mathcal{L}_{\text{GRPO}}\big(\theta; \hat{A}^S\big) + \mathcal{L}_{\text{GRPO}}\big(\theta; \hat{A}^O\big), \tag{5}$$

where $\mathcal{L}_{\text{GRPO}}(\theta; \hat{A})$ denotes Eq. (2) evaluated on the corresponding role trajectories using advantage $\hat{A}$. Following (Yu et al., 2025), we set $\beta = 0$ to remove the KL penalty and encourage exploration. Since both roles are instantiated by the same pol-

icy π_θ, this single update simultaneously improves captioning and evidence-conditioned solving.

3 Experiments

3.1 Experimental Setup

Models, Data, and Baselines. We perform direct RL training on Qwen2.5-VL-3B and Qwen2.5-VL-7B (Team, 2025) using ViRL39K (Wang et al., 2025a). ViRL39K contains 39K verifiable multimodal reasoning questions spanning diverse visual formats such as diagrams and charts, with many problems focused on visual mathematics, including geometry, non-geometry math, and chart-based reasoning. We benchmark our method against a suite of recent open-source reasoning MLLMs at both model scales. For the 3B setting, we compare with PAPO-G-3B and PAPO-D-3B (Wang et al., 2025e), MMR1-3B (Leng et al., 2025), and Vision-SR1-3B (Li et al., 2025c). For the 7B setting, we include R1-ShareVL-7B (Yao et al., 2025a), Perception-R1-7B (Xiao et al., 2025), Vision-Matters-7B (Li et al., 2025b), NoisyRollout-7B (Liu et al., 2025b), and VPPO-7B (Huang et al., 2025a). We also implement two strong RLVR baselines by fine-tuning the same Qwen2.5-VL backbones with GRPO (Shao et al., 2024) and DAPO (Yu et al., 2025). More details are provided in Appendix A.1.

Training Details. All experiments are implemented using the EasyR1 codebase (Zheng et al., 2025a) and optimized with AdamW (Loshchilov and Hutter, 2017), with a learning rate of 1×10^{-6}. Following prior work (Yao et al., 2025a; Liu et al., 2025b; Wang et al., 2025e; Huang et al., 2025a), we set the maximum response length to 2048 and use a rollout batch size of 384 for 200 optimization steps. We set the Observer rollout group size to 4. For the Solver, we use a rollout group size of 8, in line with recent multimodal RL training practice (Huang et al., 2025a; Li et al., 2025c). We adopt a caption-first warmup for the first 40 steps, during which the Solver is trained without image inputs to encourage effective caption conditioning before restoring full multimodal inputs. More training details are provided in Appendix A.2.

Evaluation Benchmarks. We evaluate on eight multimodal reasoning benchmarks, including math-related visual reasoning on MathVista (Lu et al., 2023), MathVerse (Zhang et al., 2024), MathVision (Wang et al., 2024), WeMath (Qiao et al., 2025), and DynaMath (Zou et al., 2024), and general tasks on LogicVista (Xiao et al., 2024), MMMU-Pro (Yue et al., 2025), and MMStar (Chen et al., 2024). We use VLMEvalKit (Duan et al., 2024) with greedy decoding for all benchmarks, setting temperature to 0 and top-p to 1.0. We report acc@1 as the evaluation metric. All models are evaluated under a single fixed pipeline to ensure fair comparison and reproducibility. See Appendix A.1 for additional evaluation details.

3.2 Main Results

PRCO yields consistent improvements across model scales and task categories. As shown in Table 1, PRCO improves upon the Qwen2.5-VL backbones at both scales, with average gains of 7.65 and 7.18 points in the 3B and 7B settings, respectively. Under identical training settings, PRCO outperforms GRPO and DAPO. PRCO-3B surpasses strong RLVR baselines and recent open-source reasoning MLLMs built on the same 3B backbone. PRCO-7B further achieves the best overall performance and the strongest results across all evaluated benchmarks, outperforming the strongest baseline, VPPO. Across task categories, PRCO yields steady gains on math-related benchmarks while also improving general multimodal reasoning, indicating broad cross-task generalization. Overall, PRCO enables perception-reasoning co-evolution through role-factorized, reliable learning signals under a unified policy, setting a new performance bar among open-source MLLMs.

3.3 Ablation Study

To better understand the contribution of each component in PRCO, we conduct comprehensive ablations. We report math-related, general-task, and overall averages in Table 2. More ablation details are provided in Appendix A.4.

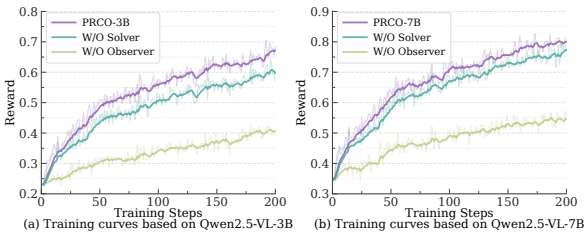
Effect of role-wise updates. We isolate PRCO's role-specific learning signals by dropping one role's trajectories during policy updates while keeping the trajectory generation procedure unchanged. PRCO w/o Solver updates only from the Observer caption trajectories, whereas PRCO w/o Observer updates only from the Solver answer trajectories. As shown in Table 2, removing Solver updates substantially reduces the overall improvement. This is expected because the base model's reasoning is limited, and outcome-driven optimization on Solver trajectories is necessary for improving end-task accuracy. Notably, PRCO w/o Solver still outper-

Model	Math-Related					General Task			Overall
	MathVerse	MathVision	MathVista	WeMath	DynaMath	LogicVista	MMMU-Pro	MMStar	Avg.
Based on Qwen2.5-VL-3B									
Base Model	34.13	22.50	65.00	23.52	12.37	38.70	26.76	56.06	34.88
GRPO	36.29	24.70	67.40	30.57	17.96	38.92	29.88	58.00	37.97
DAPO	40.98	**27.40**	70.20	35.14	20.35	43.62	31.73	60.40	41.23
PAPO-G-3B	37.56	23.61	67.60	31.05	17.96	40.26	29.01	58.86	38.24
PAPO-D-3B	40.48	26.61	69.30	32.29	19.36	**47.20**	31.27	60.66	40.90
MMR1-3B	38.57	22.17	64.50	38.29	16.36	40.49	30.34	57.20	38.49
Vision-SR1-3B	36.29	25.65	64.90	34.48	18.16	41.16	**33.12**	56.73	38.81
PRCO-3B	**42.51**	27.27	**70.30**	**40.00**	**22.36**	44.97	31.85	**61.00**	**42.53**
Based on Qwen2.5-VL-7B									
Base Model	43.02	25.46	70.20	35.43	20.35	45.41	35.49	64.26	42.45
GRPO	44.28	28.28	75.40	41.43	25.14	46.08	39.01	64.06	45.46
DAPO	48.73	29.30	74.80	45.62	26.14	47.87	41.38	65.40	47.41
PAPO-G-7B	44.79	27.20	74.30	39.62	23.55	43.17	40.11	64.93	44.71
PAPO-D-7B	47.33	24.34	76.70	39.05	25.34	48.54	41.50	66.93	46.22
R1-ShareVL-7B	48.22	29.14	73.30	45.14	24.55	48.76	38.32	65.06	46.56
Perception-R1-7B	46.70	26.74	73.40	46.48	23.75	44.07	38.20	64.33	45.46
NoisyRollout-7B	46.44	27.50	72.30	46.10	23.15	48.32	36.30	63.93	45.51
MMR1-7B	43.90	26.01	71.60	47.87	27.14	49.44	35.08	64.80	45.73
VPPO-7B	47.20	30.52	76.60	43.81	27.94	47.87	39.65	67.20	47.60
Vision-Matters-7B	47.08	27.23	72.30	41.71	24.75	48.09	37.10	62.20	45.06
Vision-SR1-7B	42.76	27.76	72.30	37.14	24.55	48.32	41.38	65.26	44.93
PRCO-7B	**49.49**	**30.86**	**77.10**	**50.29**	**29.74**	**49.66**	**42.08**	**67.80**	**49.63**

Table 1: Main results on eight multimodal reasoning benchmarks with Qwen2.5-VL-3B and Qwen2.5-VL-7B backbones. We report accuracy (%) on math-related benchmarks, general-task benchmarks, and their overall average; the best and second-best results within each backbone are highlighted in **bold** and underline, respectively.

Model	Math	Δ	General	Δ	Avg	Δ
Qwen2.5-VL-3B	31.50	-	40.51	-	34.88	-
+ PRCO w/o warmup	39.45	+7.95	45.88	+5.37	41.86	+6.98
+ PRCO w/o Observer	39.48	+7.98	45.02	+4.51	41.56	+6.68
+ PRCO w/o Solver	33.46	+1.96	41.55	+1.04	36.49	+1.61
+ PRCO	**40.49**	**+8.99**	**45.94**	**+5.43**	**42.53**	**+7.65**
Qwen2.5-VL-7B	38.89	-	48.39	-	42.45	-
+ PRCO w/o warmup	44.52	+5.63	51.57	+3.18	47.16	+4.71
+ PRCO w/o Observer	46.22	+7.33	51.47	+3.08	48.19	+5.74
+ PRCO w/o Solver	41.73	+2.84	49.16	+0.77	44.52	+2.07
+ PRCO	**47.50**	**+8.61**	**53.18**	**+4.79**	**49.63**	**+7.18**

Table 2: Ablation study of PRCO on Qwen2.5-VL-3B and Qwen2.5-VL-7B. We report average accuracy (%) on math-related benchmarks, general-task benchmarks, and all benchmarks; Δ denotes improvement over the corresponding base model.



(a) Training curves based on Qwen2.5-VL-3B (b) Training curves based on Qwen2.5-VL-7B

Figure 3: Training reward curves of PRCO and its role-ablation variants with Qwen2.5-VL-3B and Qwen2.5-VL-7B as backbones.

forms the baseline, indicating that utility-driven caption learning alone can improve final-answer accuracy and suggesting that the perception side remains a key bottleneck. In contrast, removing Observer updates yields consistent drops across model scales and benchmarks. This confirms that utility-driven evidence extraction provides complementary benefits on top of outcome-optimized reasoning by strengthening the intermediate evidence signal available to the Solver. We also report the training curves in Fig. 3, which show that PRCO achieves higher rewards throughout training.

Effect of caption-first warmup. We further ablate the caption-first warmup, where the Solver is first trained without image inputs to encourage reliance on the Observer caption before restoring full multimodal inputs. As shown in Table 2, removing warmup degrades performance on both backbones, reducing the overall average on Qwen2.5-VL-7B from 49.63 to 47.16. This suggests that warmup is important for encouraging caption usage. Without warmup, the Solver can rely on raw visual inputs too early, which weakens the learning signal for the Observer. To further diagnose this behavior, we

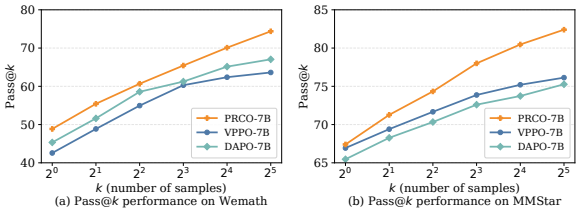


Figure 4: Pass@k comparison on WeMath and MMStar for PRCO-7B, DAPO-7B, and VPPO-7B under different inference-time sampling budgets.

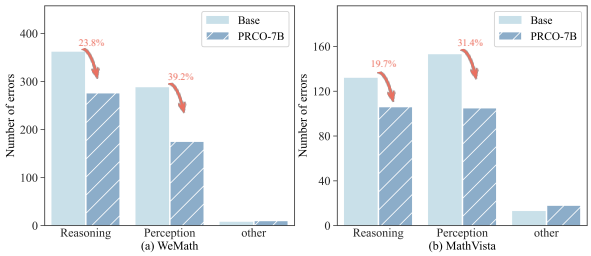


Figure 5: Error category analysis on WeMath and MathVista. Compared with Qwen2.5-VL-7B, PRCO-7B reduces both perception and reasoning errors. For presentation clarity, Knowledge and Extraction errors are grouped into the Other category.

analyze the dynamics of caption-reward variance during training in Fig. 9. Without warmup, the variance decreases rapidly. Solver outcomes then become largely insensitive to which caption is provided. Consequently, different captions induce similar downstream outcomes and yield low-contrast utility feedback to the Observer, weakening credit assignment and making perception–reasoning decoupling less effective later in training.

3.4 More Results and Analysis

Pass@k Performance. Pass@k estimates the probability that a model can solve a question within k attempts, and is commonly used as a proxy for the model's reasoning capability (Chen et al., 2021). We compare PRCO-7B with two competitive Qwen2.5-VL-7B baselines, DAPO-7B and VPPO-7B, by estimating pass@k with $k \in \{1, 2, 4, 8, 16, 32\}$ sampled solutions per question. We report the pass@k on WeMath and MMStar in Fig. 4. As k increases, PRCO-7B exhibits larger gains over the baselines on both benchmarks. On WeMath, the gap over DAPO-7B grows from 3.53 at pass@1 to 7.33 at pass@32. On MMStar, PRCO-7B is comparable to VPPO-7B at pass@1, while the margin increases from 0.47 at pass@1 to 6.27 at pass@32. This trend suggests that PRCO-7B scales better with the sampling budget, indicating more robust reasoning capability.

Error category analysis. We conduct an error-category analysis of Qwen2.5-VL-7B and PRCO-7B on WeMath and MathVista. Using the prompt in Fig. 11, we use OpenAI's GPT-5.1 Chat model to categorize each incorrect prediction into three types: perception-related errors, reasoning-related errors, and other errors (including knowledge and extraction errors). Compared with Qwen2.5-VL-7B, PRCO reduces both perception-related and reasoning-related errors, as shown in Fig. 5. On WeMath, PRCO reduces perception-related errors by 39.2% and reasoning-related errors by 23.8%. Notably, PRCO achieves a larger reduction in perception-related errors than GRPO, consistent with the results in Fig. 1. The same qualitative trend is observed on MathVista, where PRCO also reduces both perception-related and reasoning-related errors. These results suggest that separate and reliable learning signals improve question-grounded visual perception and enable more robust reasoning under explicit evidence guidance.

Effect of rollout group size. Rollout group size is a key hyperparameter in online RL, as it controls both the number of within-prompt samples and the training-time rollout budget. We further study PRCO under different rollout budgets by first fixing the Observer group size and varying the Solver rollout group size G_S. As shown in Fig. 6, increasing G_S consistently improves both PRCO and DAPO, with a larger gain from 4 to 8 than from 8 to 12. Notably, PRCO with only G_S=4 already outperforms DAPO with G=12, underscoring the effectiveness of PRCO even with a much smaller Solver-side rollout group. This indicates that PRCO benefits from separate learning signals for perception and reasoning, which decouple the two roles at the gradient level and improve optimization efficiency under a fixed rollout budget. We also study the effect of the Observer rollout group size G_O in Appendix A.4.

3.5 Case Study

To better understand how PRCO improves question-grounded visual perception, Fig. 7 presents two representative qualitative cases, showing the Observer outputs and the corresponding attention overlays obtained by averaging attention to image tokens across all layers. In case (a), the Observer accurately extracts only the task-relevant visual facts from the *Rec* column, rather than transcrib-

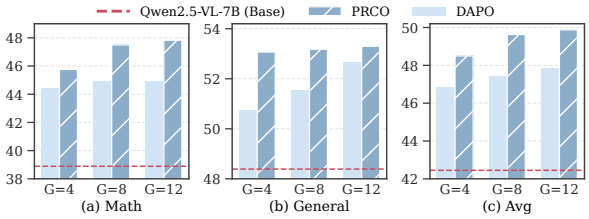


Figure 6: Effect of solver rollout group size G_S on Qwen2.5-VL-7B. We vary G_S among {4, 8, 12} and compare PRCO with DAPO on Math, General, and Avg. The dashed line denotes the base model performance.

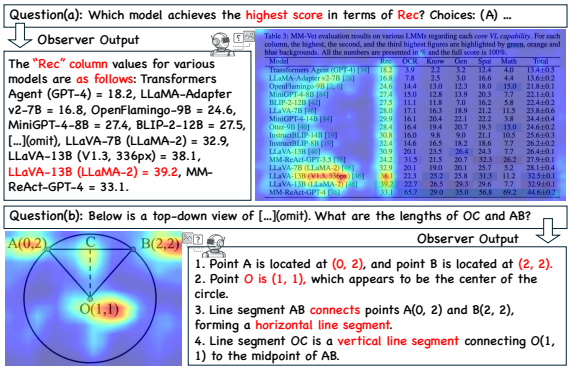


Figure 7: Qualitative analysis of PRCO-7B on two representative cases. For each case, we show the Observer output together with an attention overlay obtained by averaging attention to image tokens across all layers.

ing the full table, indicating that it preserves the evidence necessary for table-based option selection while avoiding unnecessary visual details. In case (b), the Observer localizes points, coordinates, and segment relations from the diagram, providing the full set of visual evidence required for solving the geometry question. In both cases, the attention overlays are concentrated on the corresponding question-relevant regions. More complete case studies are provided in Appendix A.5.

4 Related Work

RL with verifiable rewards for multimodal reasoning. Reinforcement learning with verifiable rewards (RLVR) defines rewards via automatic outcome verification. It is often optimized with group-based PPO variants such as GRPO and DAPO (Shao et al., 2024; Yu et al., 2025). Recent work has begun to explore RLVR for multimodal large language models (MLLMs), with improvements in data construction, curricula, and rollout strategies. Vision-R1 bootstraps multimodal chain-of-thought with staged RL schedules (Huang et al., 2025b), NoisyRollout perturbs images during rollouts to improve exploration and robustness (Liu et al., 2025b), and VL-Rethinker stabilizes training via selective replay and forced rethinking (Wang et al., 2025a). RLVR has also been paired with explicit visual operations, e.g., Active-O3 (Zhu et al., 2025), DeepEyes (Zheng et al., 2025b), Pixel Reasoner (Wang et al., 2025b), and OpenThinkIMG (Su et al., 2025).

Perception-aware RL for multimodal reasoning. Beyond outcome rewards, recent work incorporates perception-aware signals and objectives to improve visual perception in multimodal reasoning (Wang et al., 2025e; Xiao et al., 2025; Zhang et al., 2025). Perception-R1 introduces an explicit perception reward to score the fidelity of visual evidence in trajectories (Xiao et al., 2025). CapRL defines caption rewards by their question-answering utility for a vision-free LLM (Xing et al., 2025), while SOPHIA adopts semi-off-policy RL that propagates outcome rewards from external slow-thinking traces back to the model's visual understanding (Shen et al., 2025b). Other caption-centric or consistency objectives similarly optimize descriptions for downstream solvability (Gou et al., 2025; Chen et al.; Tu et al., 2025). PAPO integrates perception signals into policy optimization via objective-level regularization (Wang et al., 2025e). Reward designs based on verifiable perception proxies or perception gates provide complementary supervision (Wang et al., 2025c; Zhang et al., 2025). Credit assignment is further refined by reweighting updates toward visually dependent tokens (Huang et al., 2025a; Hong et al.; Lu et al.; Huang et al., 2026).

5 Conclusion

In this paper, we presented PRCO, a dual-role RLVR framework for multimodal reasoning that disentangles perception and reasoning under a shared policy. By assigning separate and reliable learning signals to an Observer for question-grounded evidence extraction and a Solver for evidence-conditioned reasoning, PRCO enables perception–reasoning co-evolution during RLVR training. Extensive experiments on eight challenging benchmarks showed consistent gains over strong RLVR baselines across model scales, while ablation and diagnostic analyses further validated the effectiveness of its key design choices. Overall, these results suggest that role-factorized supervision is a promising direction for improving multimodal reasoning under verifiable rewards.

Limitations

Our current study focuses on multimodal reasoning benchmarks with concise and verifiable answers. Further evaluation is needed to determine how well PRCO generalizes to more open-ended generation settings. Extending the framework to broader multimodal generation tasks is an important direction for future work, since reward signals in these settings are often less well defined. In addition, the Observer is trained with auxiliary signals for leakage detection and answer verification. Although these signals provide useful supervision in our setting, they may also introduce additional noise and computational overhead. Finally, representing visual evidence as short captions is inherently lossy. Important aspects of the input, such as global structure (e.g., layout and texture), fine-grained spatial relations, and geometric details that are difficult to compress faithfully into text, may be only partially preserved. Future work could therefore explore richer intermediate representations for visual inputs that cannot be adequately captured by caption.

Ethical Considerations

This work aims to improve multimodal reasoning by explicitly separating perception and reasoning during reinforcement learning. All experiments are conducted on publicly available datasets and benchmarks. As in prior work, these data sources may contain social biases, annotation artifacts, or other imperfections that can affect model behavior and evaluation outcomes. We do not identify additional ethical risks introduced specifically by our method beyond those already associated with multimodal model training and evaluation on existing public data. We encourage continued attention to data quality, transparent evaluation, and responsible reporting of model capabilities and limitations.

References

Meng Cao, Haoze Zhao, Can Zhang, Xiaojun Chang, Ian Reid, and Xiaodan Liang. 2025. Ground-r1: Incentivizing grounded visual reasoning via reinforcement learning. *arXiv preprint arXiv:2505.20272*.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and 1 others. 2024. Are we on the right way for evaluating large vision-language models? *Advances in Neural Information Processing Systems*, 37:27056–27087.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, and 1 others. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Zihao Chen, Beini ZHANG, Haonan Li, Miren Tian, Xiaowu Liu, Yi Bai, Dandan Tu, Changzheng Zhang, and 1 others. Caption as reward: Enhancing vision-language reasoning through dense visual description.

Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, and 1 others. 2024. Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In *Proceedings of the 32nd ACM international conference on multimedia*, pages 11198–11201.

Yue Fan, Xuehai He, Diji Yang, Kaizhi Zheng, Ching-Chen Kuo, Yuting Zheng, Sravana Jyothi Narayanaraju, Xinze Guan, and Xin Eric Wang. 2025. Grit: Teaching mllms to think with images. *arXiv preprint arXiv:2505.15879*.

Yunhao Gou, Kai Chen, Zhili Liu, Lanqing Hong, Xin Jin, Zhenguo Li, James T Kwok, and Yu Zhang. 2025. Perceptual decoupling for scalable multi-modal reasoning via reward-optimized captioning. *arXiv e-prints*, pages arXiv–2506.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Jiale Hong, Dongjie Yang, Xuecai Hu, Yong Wang, Xiangxiang Chu, and 1 others. Don't lose sight: Visually-grounded credit assignment for multimodal reasoning.

Muye Huang, Lingling Zhang, Yifei Li, Yaqiang Wu, and Jun Liu. 2026. Sketchvl: Policy optimization via fine-grained credit assignment for chart understanding and more. *arXiv preprint arXiv:2601.05688*.

Siyuan Huang, Xiaoye Qu, Yafu Li, Yun Luo, Zefeng He, Daizong Liu, and Yu Cheng. 2025a. Spotlight on token perception for multimodal reinforcement learning. *arXiv preprint arXiv:2510.09285*.

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

Sicong Leng, Jing Wang, Jiaxi Li, Hao Zhang, Zhiqiang Hu, Boqiang Zhang, Yuming Jiang, Hang Zhang, Xin Li, Lidong Bing, and 1 others. 2025. Mmr1: Enhancing multimodal reasoning with variance-aware sampling and open resources. *arXiv preprint arXiv:2509.21268*.

Shenshen Li, Kaiyuan Deng, Lei Wang, Hao Yang, Chong Peng, Peng Yan, Fumin Shen, Heng Tao Shen, and Xing Xu. 2025a. Truth in the few: High-value data selection for efficient multi-modal reasoning. *arXiv preprint arXiv:2506.04755*.

Yuting Li, Lai Wei, Kaipeng Zheng, Jingyuan Huang, Guilin Li, Bo Wang, Linghe Kong, Lichao Sun, and Weiran Huang. 2025b. Revisiting visual understanding in multimodal reasoning through a lens of image perturbation. *arXiv preprint arXiv:2506.09736*.

Zongxia Li, Wenhao Yu, Chengsong Huang, Rui Liu, Zhenwen Liang, Fuxiao Liu, Jingxi Che, Dian Yu, Jordan Boyd-Graber, Haitao Mi, and 1 others. 2025c. Self-rewarding vision-language model via reasoning decomposition. *arXiv preprint arXiv:2508.19652*.

Chengzhi Liu, Zhongxing Xu, Qingyue Wei, Juncheng Wu, James Zou, Xin Eric Wang, Yuyin Zhou, and Sheng Liu. 2025a. More thinking, less seeing? assessing amplified hallucination in multimodal reasoning models. *arXiv preprint arXiv:2505.21523*.

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

Ziyu Liu, Zeyi Sun, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Haodong Duan, Dahua Lin, and Jiaqi Wang. 2025c. Visual-rft: Visual reinforcement fine-tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2034–2044.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Jinda Lu, Junkang Wu, Jinghan Li, Kexin Huang, Guoyin Wang, Jiancan Wu, Xiang Wang, and Xiangnan He. Bridging perception and reasoning: Token reweighting for rlvr in multimodal llms.

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

Fanqing Meng, Lingxiao Du, Zongkai Liu, Zhixiang Zhou, Quanfeng Lu, Daocheng Fu, Botian Shi, Wenhai Wang, Junjun He, Kaipeng Zhang, and 1 others. 2025. Mm-eureka: Exploring visual aha moment with rule-based large-scale reinforcement learning. *arXiv preprint arXiv:2503.07365*.

Runqi Qiao, Qiuna Tan, Guanting Dong, MinhuiWu MinhuiWu, Chong Sun, Xiaoshuai Song, Jiapeng Wang, Zhuoma Gongque, Shanglin Lei, Yifan Zhang, and 1 others. 2025. We-math: Does your large multimodal model achieve human-like mathematical reasoning? In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 20023–20070.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Haozhan Shen, Peng Liu, Jingcheng Li, Chunxin Fang, Yibo Ma, Jiajia Liao, Qiaoli Shen, Zilun Zhang, Kangjia Zhao, Qianqian Zhang, and 1 others. 2025a. Vlm-r1: A stable and generalizable r1-style large vision-language model. *arXiv preprint arXiv:2504.07615*.

Junhao Shen, Haiteng Zhao, Yuzhe Gu, Songyang Gao, Kuikun Liu, Haian Huang, Jianfei Gao, Dahua Lin, Wenwei Zhang, and Kai Chen. 2025b. Semi-off-policy reinforcement learning for vision-language slow-thinking reasoning. *arXiv preprint arXiv:2507.16814*.

Zhaochen Su, Linjie Li, Mingyang Song, Yunzhuo Hao, Zhengyuan Yang, Jun Zhang, Guanjie Chen, Jiawei Gu, Juntao Li, Xiaoye Qu, and 1 others. 2025. Openthinkimg: Learning to think with images via visual tool reinforcement learning. *arXiv preprint arXiv:2505.08617*.

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, and 1 others. 2025. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*.

Qwen Team. 2025. Qwen2.5-vl.

Songjun Tu, Qichao Zhang, Jingbo Sun, Yuqian Fu, Linjing Li, Xiangyuan Lan, Dongmei Jiang, Yaowei Wang, and Dongbin Zhao. 2025. Perception-consistency multimodal large language models reasoning via caption-regularized policy optimization. *arXiv preprint arXiv:2509.21854*.

Zhongwei Wan, Zhihao Dou, Che Liu, Yu Zhang, Dongfei Cui, Qinjian Zhao, Hui Shen, Jing Xiong, Yi Xin, Yifan Jiang, and 1 others. 2025. Srpo: Enhancing multimodal llm reasoning via reflection-aware reinforcement learning. *arXiv preprint arXiv:2506.01713*.

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

Haozhe Wang, Alex Su, Weiming Ren, Fangzhen Lin, and Wenhu Chen. 2025b. Pixel reasoner: Incentivizing pixel-space reasoning with curiosity-driven reinforcement learning. *arXiv preprint arXiv:2505.15966*.

Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Houxing Ren, Aojun Zhou, Mingjie Zhan, and Hongsheng Li. 2024. Measuring multimodal mathematical reasoning with math-vision dataset. *Advances in Neural Information Processing Systems*, 37:95095–95169.

Xiyao Wang, Zhengyuan Yang, Chao Feng, Yongyuan Liang, Yuhang Zhou, Xiaoyu Liu, Ziyi Zang, Ming Li, Chung-Ching Lin, Kevin Lin, and 1 others. 2025c. Vicrit: A verifiable reinforcement learning proxy task for visual perception in vlms. *arXiv preprint arXiv:2506.10128*.

Xiyao Wang, Zhengyuan Yang, Chao Feng, Hongjin Lu, Linjie Li, Chung-Ching Lin, Kevin Lin, Furong Huang, and Lijuan Wang. 2025d. Sota with less: Mcts-guided sample selection for data-efficient visual reasoning self-improvement. *arXiv preprint arXiv:2504.07934*.

Zhenhailong Wang, Xuehang Guo, Sofia Stoica, Haiyang Xu, Hongru Wang, Hyeonjeong Ha, Xiusi Chen, Yangyi Chen, Ming Yan, Fei Huang, and 1 others. 2025e. Perception-aware policy optimization for multimodal reasoning. *arXiv preprint arXiv:2507.06448*.

Zijian Wu, Jinjie Ni, Xiangyan Liu, Zichen Liu, Hang Yan, and Michael Qizhe Shieh. 2025. Synthrl: Scaling visual reasoning with verifiable data synthesis. *arXiv preprint arXiv:2506.02096*.

Tong Xiao, Xin Xu, Zhenya Huang, Hongyu Gao, Quan Liu, Qi Liu, and Enhong Chen. 2025. Perception-r1: Advancing multimodal reasoning capabilities of mllms via visual perception reward. *arXiv preprint arXiv:2506.07218*.

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

Long Xing, Xiaoyi Dong, Yuhang Zang, Yuhang Cao, Jianze Liang, Qidong Huang, Jiaqi Wang, Feng Wu, and Dahua Lin. 2025. Caprl: Stimulating dense image caption capabilities via reinforcement learning. *arXiv preprint arXiv:2509.22647*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Huanjin Yao, Qixiang Yin, Jingyi Zhang, Min Yang, Yibo Wang, Wenhao Wu, Fei Su, Li Shen, Minghui Qiu, Dacheng Tao, and 1 others. 2025a. R1-sharevl: Incentivizing reasoning capability of multimodal large language models via share-grpo. *arXiv preprint arXiv:2505.16673*.

Zijun Yao, Yantao Liu, Yanxu Chen, Jianhui Chen, Junfeng Fang, Lei Hou, Juanzi Li, and Tat-Seng Chua. 2025b. Are reasoning models more prone to hallucination? *arXiv preprint arXiv:2505.23646*.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, and 1 others. 2025. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*.

Xiang Yue, Tianyu Zheng, Yuansheng Ni, Yubo Wang, Kai Zhang, Shengbang Tong, Yuxuan Sun, Botao Yu, Ge Zhang, Huan Sun, and 1 others. 2025. Mmmu-pro: A more robust multi-discipline multimodal understanding benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15134–15186.

Chi Zhang, Haibo Qiu, Qiming Zhang, Yufei Xu, Zhixiong Zeng, Siqi Yang, Peng Shi, Lin Ma, and Jing Zhang. 2025. Perceptual-evidence anchored reinforced learning for multimodal reasoning. *arXiv preprint arXiv:2511.18437*.

Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Yu Qiao, and 1 others. 2024. Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? In *European Conference on Computer Vision*, pages 169–186. Springer.

Yaowei Zheng, Junting Lu, Shenzhi Wang, Zhangchi Feng, Dongdong Kuang, and Yuwen Xiong. 2025a. Easyr1: An efficient, scalable, multi-modality rl training framework. *arXiv preprint arXiv:2501.12345*.

Ziwei Zheng, Michael Yang, Jack Hong, Chenxiao Zhao, Guohai Xu, Le Yang, Chao Shen, and Xing Yu. 2025b. Deepeyes: Incentivizing" thinking with images" via reinforcement learning. *arXiv preprint arXiv:2505.14362*.

Muzhi Zhu, Hao Zhong, Canyu Zhao, Zongze Du, Zheng Huang, Mingyu Liu, Hao Chen, Cheng Zou, Jingdong Chen, Ming Yang, and 1 others. 2025. Active-o3: Empowering multimodal large language models with active perception via grpo. *arXiv preprint arXiv:2505.21457*.

Chengke Zou, Xingang Guo, Rui Yang, Junyu Zhang, Bin Hu, and Huan Zhang. 2024. Dynamath: A dynamic visual benchmark for evaluating mathematical reasoning robustness of vision language models. *arXiv preprint arXiv:2411.00836*.

A Appendix

A.1 Evaluation Details

We evaluate our method on a diverse set of benchmarks spanning both math-related reasoning tasks and general multimodal tasks. Table 3 summarizes the benchmarks used in our evaluation, where the evaluation splits and reported metrics follow the settings in VLMEvalKit (Duan et al., 2024).

Math-Related Reasoning Tasks. This category evaluates mathematical reasoning abilities.

- **MathVerse** (Zhang et al., 2024) is a benchmark for multimodal mathematical reasoning that examines whether MLLMs truly understand diagrams. By presenting each problem

in multiple versions with different distributions of textual and visual information, it enables fine-grained analysis of a model's reliance on visual versus textual cues.

- **MathVision** (Wang et al., 2024) focuses on competition-level multimodal math reasoning. Its problems are drawn from real mathematics competitions and cover multiple disciplines and difficulty levels, providing a challenging testbed for advanced reasoning over diagrams and symbolic content.

- **MathVista** (Lu et al., 2023) is a comprehensive benchmark for visual mathematical reasoning. It covers diverse task types such as geometry, charts, tables, and scientific figures, making it a broad benchmark for evaluating mathematical reasoning in visually grounded settings.

- **WeMath** (Qiao et al., 2025) introduces a diagnostic evaluation paradigm for multimodal math reasoning. By decomposing problems into sub-problems based on knowledge concepts, it supports fine-grained analysis of a model's strengths and weaknesses.

- **DynaMath** (Zou et al., 2024) is designed to evaluate the robustness and generalization of multimodal mathematical reasoning. It generates dynamic variations of seed problems, allowing evaluation of whether a model can maintain consistent reasoning under controlled changes.

General Multimodal Tasks. This category evaluates broader multimodal understanding abilities.

- **LogicVista** (Xiao et al., 2024) focuses on logical reasoning in visual contexts. Although not limited to mathematics, it is useful for evaluating whether models can perform structured reasoning grounded in diagrams and other visual inputs.

- **MMMU-Pro** (Yue et al., 2025) is an enhanced benchmark for multidisciplinary multimodal understanding and reasoning. It is designed to reduce shortcuts from textual clues and provide a more rigorous evaluation of genuine visual understanding across subjects.

- **MMStar** (Chen et al., 2024) is a curated benchmark for core multimodal reasoning abilities. Its samples are designed to require genuine visual understanding, making it a concise but challenging benchmark for multimodal reasoning evaluation.

Evaluation parameters. Unless otherwise specified, we use greedy decoding for acc@1 evaluation, with temperature set to 0.0, top-p to 1.0, top-k to -1, and the maximum number of generated tokens to 2048. For pass@k evaluation, we instead use temperature 0.6, top-p 0.95, and top-k -1.

A.2 Training Details

In this section, we describe the training details of the different methods. Our implementation is based on the EasyR1 framework (Zheng et al., 2025a). We train all models on the ViRL39K training set for 200 steps, and use MMK12 (Meng et al., 2025) as the validation set. Following DAPO (Yu et al., 2025), we use dynamic sampling, clip-higher, and token-level policy gradient loss. The clip ratio low and high are set to 0.2 and 0.28, respectively, and no KL-divergence penalty is applied. We also remove the standard-deviation normalization term when computing the grouped advantage. We find this design more suitable for role-specific optimization, as it preserves the original relative reward differences within each role and leads to more faithful advantage updates for both the Observer and the Solver. All training is conducted on 8 NVIDIA H200 GPUs. Table 4 summarizes the main hyperparameters used in our experiments. For PRCO, the maximum rollout length is set to 1024 tokens for the Observer and 2048 tokens for the Solver. We use Qwen3-VL-8B-Instruct (Yang et al., 2025) as the auxiliary model for answer leakage checking. We also adopt a caption-first warmup, during which the Solver is trained without image inputs to encourage caption conditioning before restoring full multimodal inputs.

A.3 Prompt Template

In this section, we present the prompts used in our experiments. For the RLVR baselines, including GRPO and DAPO, we follow the prompt setting used in EasyR1 (Zheng et al., 2025a), where the model is asked to first reason through the problem and then provide the final answer in a boxed format. For PRCO, we use role-specific prompts for the Observer and Solver. The Observer is prompted to produce a question-conditioned and visually grounded caption that captures the key visual evidence with-

Benchmark	Evaluation Split	Num of Sample	Metric
MathVerse (Zhang et al., 2024)	MathVerse_MINI_Vision_Only	788	Overall
MathVista (Lu et al., 2023)	MathVista_MINI	1000	acc
MathVision (Wang et al., 2024)	MathVision	3040	acc
WeMath (Qiao et al., 2025)	WeMath	1740	Score (Strict)
DynaMath (Zou et al., 2024)	DynaMath	5010	Overall (Worst Case)
LogicVista (Xiao et al., 2024)	LogicVista	447	Overall
MMMU-Pro (Yue et al., 2025)	MMMU_Pro_V	1730	Overall
MMStar (Chen et al., 2024)	MMStar	1500	Overall

Table 3: Details of the benchmarks we evaluate. The evaluation splits and reported metrics follow the settings in VLMEvalKit (Duan et al., 2024). We report single-sample greedy scores using each benchmark's official VLMEvalKit metric.

Method	lr	Max Len.	Steps	Warmup	Opt.	Rollout BS	Freeze VT	Temp.	top-p	top-k
GRPO	1e-6	2048	200	–	AdamW	384	False	1	1.0	-1
DAPO	1e-6	2048	200	–	AdamW	384	False	1	1.0	-1
PRCO	1e-6	2048	200	40	AdamW	384	False	1	1.0	-1

Table 4: Training hyperparameters used in our experiments.

out revealing the final answer, while the Solver is prompted to answer the question based primarily on the caption and consult the image only when necessary. Figure 10 shows the prompts used for PRCO and the RLVR baselines. In addition to the training and inference prompts, we also use auxiliary prompts for analysis. Specifically, we use a leakage-checking prompt to verify that the Observer caption does not directly reveal the answer, and an error-type classification prompt to categorize model failures in the error analysis. Figure 11 shows these auxiliary prompts.

A.4 More Results and Analysis

We additionally evaluate three PRCO-7B variants to study the roles of Solver-side visual grounding, leakage suppression, and co-evolving utility feedback: (*i*) PRCO w/ $I^S = \emptyset$, which keeps the Solver image input empty throughout the RL stage; (*ii*) PRCO w/o Leakage Checker, which removes the leakage checker from the Observer utility reward; and (*iii*) PRCO w/ Fixed Utility Estimator, which replaces the co-evolving Solver with a fixed Qwen2.5-VL-7B for caption utility estimation.

Table 5 shows that all three variants underperform the full PRCO-7B, confirming that PRCO's gains arise from the combination of evidence-conditioned solving, utility reward with leakage checking, and Observer–Solver co-evolution. The largest drop is observed for PRCO w/ $I^S = \emptyset$, where the Solver never regains access to the image after the caption-first warmup. This suggests that while restricting the Solver to caption-based evidence is beneficial in early training, access to the image remains important in later RL optimization. In PRCO, the caption is the primary evidence channel, but restored image access still helps recover global structure, fine-grained spatial relations, and geometric details that are difficult to fully compress into text. Removing the leakage checker also degrades the overall average, indicating that suppressing answer leakage is important for learning useful intermediate evidence. Without leakage checking, the Observer is more likely to exploit answer shortcutting by placing the final answer directly in the caption, rather than extracting task-relevant visual evidence. This weakens the utility reward as a supervision signal for evidence extraction and blurs the credit assignment between perception and reasoning. PRCO w/ Fixed Utility Estimator further underperforms standard PRCO, suggesting that Observer learning benefits more from utility feedback that co-evolves with the Solver and remains better aligned with its changing information needs.

Caption rollout group size. We further vary the caption rollout group size G_O on Qwen2.5-VL-7B while keeping the Solver rollout group size fixed. As shown in Fig. 8, the effect of G_O is not monotonic: performance improves from $G_O = 2$ to $G_O = 4$, but slightly declines at $G_O = 8$. A possible explanation is that perception saturates earlier

Setting	MathVerse	MathVision	MathVista	WeMath	DynaMath	LogicVista	MMMU-Pro	MMStar	Avg.
Qwen2.5-VL-7B (Base)	43.02	25.46	70.20	35.43	20.35	45.41	35.49	64.26	42.45
PRCO-7B (Ours)	49.49	30.86	77.10	50.29	29.74	49.66	42.08	67.80	49.63
PRCO w/ $I^S = \emptyset$	47.34	28.82	75.20	43.81	28.94	48.77	42.89	66.13	47.74 ↓1.89
PRCO w/ Fixed Utility Estimator	49.24	29.67	76.70	45.52	29.74	48.55	40.17	66.20	48.22 ↓1.41
PRCO w/o Leakage Checker	50.00	29.31	76.00	44.19	29.54	51.45	41.33	67.60	48.68 ↓0.95

Table 5: Additional ablations of PRCO-7B. We report accuracy (%) on eight benchmarks. Red downward arrows in the Avg. column indicate the drop relative to PRCO-7B (ours).

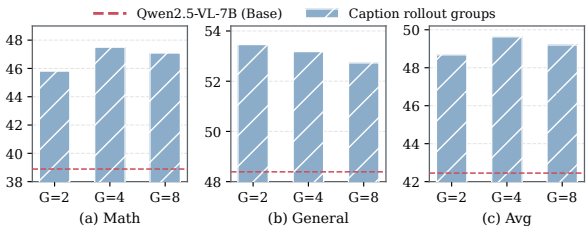


Figure 8: Ablation of the caption rollout group size G_O in PRCO on Qwen2.5-VL-7B. Bars show different G_O settings $(2, 4, 8)$ on Math, General, and Avg, and the dashed line indicates the base model performance.

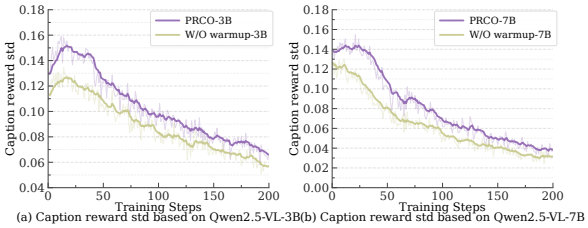


Figure 9: Training caption reward standard deviation curves of PRCO and its W/O warmup variant with Qwen2.5-VL-3B and Qwen2.5-VL-7B as backbones.

than reasoning, which is also consistent with Fig. 3, where the variant without Solver updates reaches its plateau relatively early. Once the Observer already provides sufficiently informative captions, further increasing G_O yields diminishing returns and may reduce the relative benefit of allocating more rollouts to the Solver. Under a fixed compute budget, allocating additional rollouts to the Solver appears more effective than further increasing the Observer rollout group size. This trend is also reflected in Fig. 8 and Fig. 6: the setting with $G_O = 8, G_S = 8$ performs worse than $G_O = 4, G_S = 12$, even though the former uses a larger Observer rollout group. We use $G_O = 4$, which provides a good trade-off between performance and training cost.

A.5 Case Study

We present four representative qualitative cases in Figures 12, 13, 14, and 15, covering synthetic object filtering, bar-chart reasoning, table-based option selection, and diagram-grounded geometry reasoning. These examples span several visual formats that frequently appear in our evaluation suite, including rendered scenes, charts, tables, and geometric diagrams. Across all cases, PRCO exhibits the intended division of labor between its two roles. The Observer first converts the image into a question-conditioned evidence caption that externalizes the entities, attributes, values, and relations most relevant to the question, while the Solver performs the downstream counting, comparison, or derivation over this intermediate evidence. Qualitatively, the Observer tends to preserve the attributes, numeric values, and spatial relations most relevant to the question, while the Solver performs the required filtering, comparison, counting, or geometric deduction on top of the extracted evidence. These examples complement the main quantitative results by showing that PRCO not only improves final-answer accuracy, but also yields cleaner and more task-aligned intermediate evidence.

Prompt for Observer

You are the Observer.
You are given an image and a relevant question. Your task is to write a caption that extracts only the visually grounded details most useful for answering the question later.
Question: {question}
Caption guidelines:
- Focus on the question's target(s): describe relevant objects/regions/values with attributes and spatial relations (left/right/top/bottom, near/far, inside/overlapping), including counts and comparisons when relevant.
- Include all visually grounded details necessary and sufficient to answer the question later. You may omit visual clutter unrelated to the question.
- Write the caption with enough visual evidence that a later LLM can answer Question as if it had seen the image (using only the caption + the question).
- Don't provide the final answer. Only describe what is directly observable in the image, and avoid any additional reasoning or calculations.
- Write a thorough caption that preserves enough visual details to reconstruct the scene later; avoid overly short summaries.
- When the answer depends on specific labels, numbers, or option mappings, transcribe them explicitly rather than summarizing.

Please format the caption as a Markdown bullet list rather than a paragraph.

Prompt for Solver

You are the Solver.
Use the caption as the PRIMARY and DEFAULT input and solve directly from it.
Only check the image when the caption does not provide enough information to complete the task.
The detailed caption of the provided image: <caption>
Question: {question}
Now perform your reasoning inside <think>...</think>, then output the final answer in \boxed{}.

Prompt for GRPO and DAPO

{Question}

You first think through the reasoning process as an internal monologue, enclosed within <think> </think> tags. Then, provide your final answer enclosed within \boxed{}.

Figure 10: Training and inference prompt templates of the PRCO Observer and Solver, GRPO, and DAPO.

Prompt for Leakage Checker

You are the Leakage Checker.
You will be given: - The question. - The Observer caption describing an image.
Your task: decide whether the caption contains answer leakage.
Question: `{question}`
Observer caption: `<caption>`
Reminder: 1 = LEAK, 0 = SAFE.
[Leakage Evaluation Rules...]
Output format (MUST follow exactly):
Return a valid JSON object with exactly two keys:
- `"label"`: 0 or 1
- `"reason"`: a short reason (1–5 sentences)

Example:

```
{
  "reason": "The caption does not explicitly state a final answer.",
  "label": 0
}
```

Now output the final answer as the JSON object (and nothing else).

Prompt for Error Type Classification

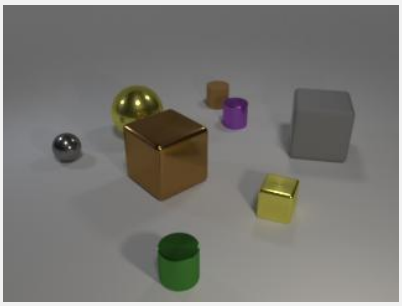
You will be given the following inputs:
- **Image**
- **Question:** `{question}`
- **Model response:** `{model_response}`
- **Gold answer:** `{gold_answer}`

Your task: Decide which error type best describes why the model response differs from the gold answer, and output exactly one primary label plus a 1–3 sentence rationale.
Choose exactly one PRIMARY type from this set:
- **Perception**: Perception error (misread options/tables, failed to extract info; if downstream mistakes are caused by misreading/missing visual info, label as Perception even if reasoning also fails).
- **Reasoning**: Reasoning error (missing/contradictory steps, wrong logical chain, arithmetic/calculation mistakes, algebraic manipulation errors, etc.).
- **Knowledge**: Factual / domain knowledge error.
- **Extraction**: Final-answer extraction/formatting error.
- **Other**: Other / cannot categorize.

Output requirements:
- Return a single minified JSON object only.
- Do not output markdown, code fences, or any extra explanation.
- Use exactly this format: `{"rationale":"1-3 sentences why","category":"<label>"}`

Figure 11: Prompt templates for leakage checking and error-type classification.

Case Study 1:



Problem: Subtract all big gray matte things. Subtract all small metallic cylinders. How many objects are left?

Observer Output.

- A large golden metallic cube is present in the center-left.
- A large yellow metallic sphere is located towards the back-left.
- A small metallic gray sphere is on the far left.
- A small yellow metallic cube is near the center-right.
- A large gray matte cube is on the far right.
- A small green metallic cylinder is in the foreground.
- A small purple metallic cylinder is near the center.
- A small brown matte cylinder is in the background.

Solver Output.

<think>

To determine the number of objects left after the specified subtractions, we start by identifying all objects in the image. First, we subtract all big gray matte things. There is one big gray matte cube, so we remove it.

Remaining objects:

- Large yellow metallic sphere.
- Large metallic gold cube.
- Small metallic gray sphere.
- Small brown matte cylinder.
- Small purple metallic cylinder.
- Small green metallic cylinder.
- Small yellow metallic cube.

Next, we subtract all small metallic cylinders. There are two small metallic cylinders: the purple one and the green one. Removing these:

Remaining objects:

- Large yellow metallic sphere.
- Large metallic gold cube.
- Small metallic gray sphere.
- Small brown matte cylinder.
- Small yellow metallic cube.

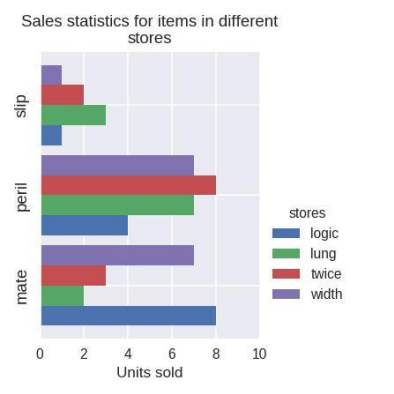
Thus, the number of objects left is 5.

</think>

$\boxed{5}$ ← Correct

Figure 12: Qualitative example of PRCO on synthetic object filtering. The Observer enumerates the relevant objects together with size, color, material, shape, and coarse spatial cues, converting the scene into a question-conditioned object inventory. PRCO performs discrete filtering and counting from explicit visual evidence.

Case Study 2:



Problem: How many items sold less than 1 units in at least one store?

Observer Output.

- For the item "slip", the sales for "logic" are 1 unit, for "lung" are 3 units, for "twice" are 2 units, and for "width" are 1 unit.
- For the item "peril", the sales for "logic" are 4 units, for "lung" are 7 units, for "twice" are 8 units, and for "width" are 7 units.
- For the item "mate", the sales for "logic" are 8 units, for "lung" are 2 units, for "twice" are 3 units, and for "width" are 7 units.

Solver Output.

<think>

To determine how many items sold less than 1 unit in at least one store, we need to carefully examine the bar lengths for each item in the given chart.

1. **Item "slip":**
   - Logic: 1 unit (not less than 1).
   - Lung: 3 units (not less than 1).
   - Twice: 2 units (not less than 1).
   - Width: 1 unit (not less than 1).

   Since all bars for "slip" are either 1 or greater, none of them are less than 1.
2. **Item "peril":**
   - Logic: 4 units (not less than 1).
   - Lung: 7 units (not less than 1).
   - Twice: 8 units (not less than 1).
   - Width: 7 units (not less than 1).

   All bars for "peril" are greater than or equal to 1, so none are less than 1.
3. **Item "mate":**
   - Logic: 8 units (not less than 1).
   - Lung: 2 units (not less than 1).
   - Twice: 3 units (not less than 1).
   - Width: 7 units (not less than 1).

   All bars for "mate" are greater than or equal to 1, so none are less than 1.

Upon reviewing all items, none of the bars for any item show values less than 1.
Therefore, the number of items sold less than 1 unit in at least one store is 0.
</think>

$\boxed{0}$ ← Correct

Figure 13: Qualitative example of PRCO on bar-chart threshold counting. The Observer transcribes item-wise values across stores into explicit textual evidence, and the Solver then checks the threshold condition and aggregates over items. PRCO supports accurate chart value extraction and evidence-conditioned counting through a compact intermediate representation.

Case Study 3:

Table 3: MM-Vet evaluation results on various LMMs regarding each *core VL capability*. For each column, the highest, the second, and the third highest figures are highlighted by green, orange and blue backgrounds. All the numbers are presented in % and the full score is 100%.

Model	Rec	OCR	Know	Gen	Spat	Math	Total
Transformers Agent (GPT-4) [34]	18.2	3.9	2.2	3.2	12.4	4.0	13.4±0.5
LLaMA-Adapter v2-7B [28]	16.8	7.8	2.5	3.0	16.6	4.4	13.6±0.2
OpenFlamingo-9B [2, 6]	24.6	14.4	13.0	12.3	18.0	15.0	21.8±0.1
MiniGPT-4-8B [84]	27.4	15.0	12.8	13.9	20.3	7.7	22.1±0.1
BLIP-2-12B [42]	27.5	11.1	11.8	7.0	16.2	5.8	22.4±0.2
LLaVA-7B [46]	28.0	17.1	16.3	18.9	21.2	11.5	23.8±0.6
MiniGPT-4-14B [84]	29.9	16.1	20.4	22.1	22.2	3.8	24.4±0.4
Otter-9B [40]	28.4	16.4	19.4	20.7	19.3	15.0	24.6±0.2
InstructBLIP-14B [19]	30.8	16.0	9.8	9.0	21.1	10.5	25.6±0.3
InstructBLIP-8B [19]	32.4	14.6	16.5	18.2	18.6	7.7	26.2±0.2
LLaVA-13B [46]	30.9	20.1	23.5	26.4	24.3	7.7	26.4±0.1
MM-ReAct-GPT-3.5 [76]	24.2	31.5	21.5	20.7	32.3	26.2	27.9±0.1
LLaVA-7B (LLaMA-2) [46]	32.9	20.1	19.0	20.1	25.7	5.2	28.1±0.4
LLaVA-13B (V1.3, 336px) [46]	38.1	22.3	25.2	25.8	31.3	11.2	32.5±0.1
LLaVA-13B (LLaMA-2) [46]	39.2	22.7	26.5	29.3	29.6	7.7	32.9±0.1
MM-ReAct-GPT-4 [76]	33.1	65.7	29.0	35.0	56.8	69.2	44.6±0.2

Problem: Which model achieves the highest score in terms of Rec? Choices: (A) Transformers Agent (GPT-4), (B) LLaMA-Adapter v2-7B, (C) LLaVA-7B, (D) Otter-9B, (E) MM-ReAct-GPT-3.5, (F) LLaVA-13B (LLaMA-2), (G) MM-ReAct-GPT-4

Observer Output.

- The "Rec" column values for various models are as follows: Transformers Agent (GPT-4) = 18.2, LLaMA-Adapter v2-7B = 16.8, OpenFlamingo-9B = 24.6, MiniGPT-4-8B = 27.4, BLIP-2-12B = 27.5, LLaVA-7B = 28.0, MiniGPT-4-14B = 29.9, Otter-9B = 28.4, InstructBLIP-14B = 30.8, InstructBLIP-8B = 32.4, LLaVA-13B = 30.9, MM-ReAct-GPT-3.5 = 24.2, LLaVA-7B (LLaMA-2) = 32.9, LLaVA-13B (V1.3, 336px) = 38.1, LLaVA-13B (LLaMA-2) = 39.2, MM-ReAct-GPT-4 = 33.1.

Solver Output.

`<think>`

To determine which model achieves the highest score in terms of "Rec" (Recognition), we need to identify the highest value in the "Rec" column of the table.

The "Rec" scores for the models are as follows:

- Transformers Agent (GPT-4): 18.2
- LLaMA-Adapter v2-7B: 16.8
- OpenFlamingo-9B: 24.6
- MiniGPT-4-8B: 27.4
- BLIP-2-12B: 27.5
- LLaVA-7B: 28.0
- MiniGPT-4-14B: 29.9
- Otter-9B: 28.4
- InstructBLIP-14B: 30.8
- InstructBLIP-8B: 32.4
- LLaVA-13B: 30.9
- MM-ReAct-GPT-3.5: 24.2
- LLaVA-7B (LLaMA-2): 32.9
- LLaVA-13B (V1.3, 336px): 38.1
- LLaVA-13B (LLaMA-2): 39.2
- MM-ReAct-GPT-4: 33.1

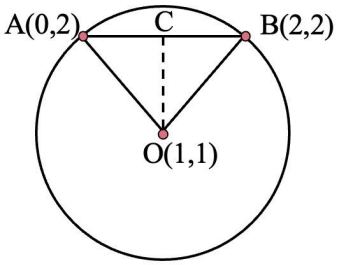
Among these, the highest score in the "Rec" column is 39.2, which belongs to the model "LLaVA-13B (LLaMA-2)". Therefore, the correct choice is (F).

`</think>`

$\boxed{F}$ ← `Correct`

Figure 14: Qualitative example of PRCO on table-based option selection. The Observer extracts only the question-relevant entries in the `Rec` column, rather than transcribing the full table, after which the Solver identifies the maximum and maps it to the correct answer option. PRCO focuses on task-relevant evidence while avoiding unnecessary visual details.

Case Study 4:



Problem: Below is a top-down view of a circular water surface. In the diagram, one unit length represents 1 meter. What is the degree measure of $\angle BOA$? What are the lengths of OC and AB?
Options: A. 30°, the lengths of OC and AB are 1m and 2m respectively;
B. 45°, the lengths of OC and AB are 2m and 2m respectively;
C. 90°, the lengths of OC and AB are 1m and 2m respectively;
D. Cannot be determined; E. No correct answer

Observer Output.

- Point A is located at $(0, 2)$, and point B is located at $(2, 2)$.
- Point O is at $(1, 1)$, which appears to be the center of the circle.
- Line segment AB connects points A$(0, 2)$ and B$(2, 2)$, forming a horizontal line segment.
- Line segment OC is a vertical line segment connecting O$(1, 1)$ to the midpoint of AB.

Solver Output.

<think>

To determine the degree measure of $\angle BOA$ and the lengths of OC and AB, we start by finding the radius of the circle. Given points $A(0, 2)$ and $B(2, 2)$, the center $O(1, 1)$ of the circle can be used to find the radius.
First, calculate the length OA:

$$OA = \sqrt{(1-0)^2 + (1-2)^2} = \sqrt{1^2 + (-1)^2} = \sqrt{1+1} = \sqrt{2}.$$

Similarly, calculate OB:

$$OB = \sqrt{(1-2)^2 + (1-2)^2} = \sqrt{(-1)^2 + (-1)^2} = \sqrt{1+1} = \sqrt{2}.$$

Since $OA = OB = \sqrt{2}$, the radius r of the circle is $\sqrt{2}$.
Next, we find OC, the distance from the center $O(1, 1)$ to the midpoint C of AB. Since $A(0, 2)$ and $B(2, 2)$, the midpoint C of AB is:

$$C = \left(\frac{0+2}{2}, \frac{2+2}{2}\right) = (1, 2).$$

Calculate OC:

$$OC = \sqrt{(1-1)^2 + (2-1)^2} = \sqrt{0+1} = 1.$$

Next, we calculate the length AB:

$$AB = \sqrt{(2-0)^2 + (2-2)^2} = \sqrt{2^2 + 0^2} = \sqrt{4} = 2.$$

Since $OA = OB = \sqrt{2}$ and $AB = 2$, triangle AOB is a right triangle at O, so

$$\angle BOA = 90^\circ.$$

Therefore, the length $OC = 1$, and $AB = 2$.
So, the correct option is:

$$\boxed{C}.$$

</think>

$\boxed{C}$ ← Correct

Figure 15: Qualitative example of PRCO on diagram-grounded geometry reasoning. The Observer localizes points, coordinates, and segment relations from the diagram, and the Solver then uses this structured evidence to derive $\angle BOA$, OC, and AB.