CPPO: Contrastive Perception Policy Optimization for VLM Agents

Ahmad Rezaei [† 1] **Mohsen Gholami** [1] **Saeed Ranjbar Alvar** [1] **Kevin Cannons** [1] **Mohammad Asiful Hossain** [1]
Zhou Weimin [2] **Yong Zhang** [‡ 1] **Mohammad Akbari** [‡ 1]

Code CPPO

Abstract

We introduce **CPPO**, a Contrastive Perception Policy Optimization method for finetuning vision–language models (VLMs). Reliable perception is a core requirement for VLM-based agents that must reason and act in open-ended environments: faulty visual grounding cascades directly into faulty actions, hallucinated tool calls, and unsafe decisions. While reinforcement learning (RL) has significantly improved reasoning in language models, extending these advances to multimodal agents requires improving both perception and reasoning. Prior works address this challenge mainly through explicit perception rewards, which often require extra LLM judges, ground-truth annotations, or forced separation of perception from reasoning. CPPO addresses this limitation in a self-supervised manner by extending the RL objective with a Contrastive Perception Loss (CPL) that provides a direct learning signal for visual grounding. The contrastive objective encourages the model to become more sensitive to input visual information. To apply this signal effectively, CPPO identifies perception tokens using an entropy-shift mechanism in the model's output distributions under perturbed images and applies the contrastive loss selectively to those tokens during training. Experiments show that CPPO surpasses prior methods while avoiding extra models, making training more efficient and scalable, and yielding policies that are better suited to perception-critical agentic tasks.

† Corresponding author. ‡ Co-leads. [1]Huawei Technologies Canada Co. Ltd. [2]Huawei Cloud. Correspondence to: Ahmad Rezaei <ahmad.rezaei2@huawei.com>.

Published at the Second Workshop on Agents in the Wild: Safety, Security, and Beyond (AIWILD) at ICML 2026. Copyright 2026 by the author(s).

1. Introduction

Vision–language models (VLMs) are increasingly deployed as the perceptual backbone of *multimodal agents* that browse interfaces, operate computer-use environments, and ground long-horizon plans in real visual scenes. In all such settings, the agent's behavior is only as trustworthy as its perception: a single misread chart, a misidentified UI element, or a hallucinated visual fact propagates into wrong actions and unsafe outcomes. Improving the perceptual grounding of VLM policies is therefore a core safety lever for agents in open-ended environments.

Reinforcement learning (RL) with verifiable rewards has emerged as an effective finetuning method. Notably, (DeepSeek-AI, 2025) showed the potential of language models to develop reasoning capabilities without explicit step-by-step supervision, focusing on their self-evolution through a pure RL process. In contrast, VLMs often exhibit weaker multimodal reasoning performance compared to their language-only counterparts (Yang et al., 2025c; Bi et al., 2025; Gholami et al., 2025; Cannons et al., 2025). Given the success of RL in language models, recent research has focused on extending RL-based methods to VLMs and multimodal reasoning (Xia et al., 2025; Wang et al., 2025b; Li et al., 2025a; Liu et al., 2025).

In the language-only setting, the policy model draws on its internal knowledge to generate step-by-step logical inference tokens, which we refer to as reasoning tokens. For a VLM policy, however, accurate perception is also required to generate query-relevant factual tokens from the image. We refer to these tokens that encode image information as perception tokens. PAPO (Wang et al., 2025b) shows that wrong perception tokens are a significant source of failures in multimodal reasoning. However, RL algorithms with verifiable final-answer rewards (*e.g.*, DeepSeek-AI (2025)) do not separate perception from reasoning errors. This design is problematic, since inaccurate perception tokens will lead to an incorrect final answer, even with correct reasoning steps. Therefore, achieving the optimal policy is difficult when all output tokens are penalized based on the final answer alone. This limitation raises two questions: *1) How can the*

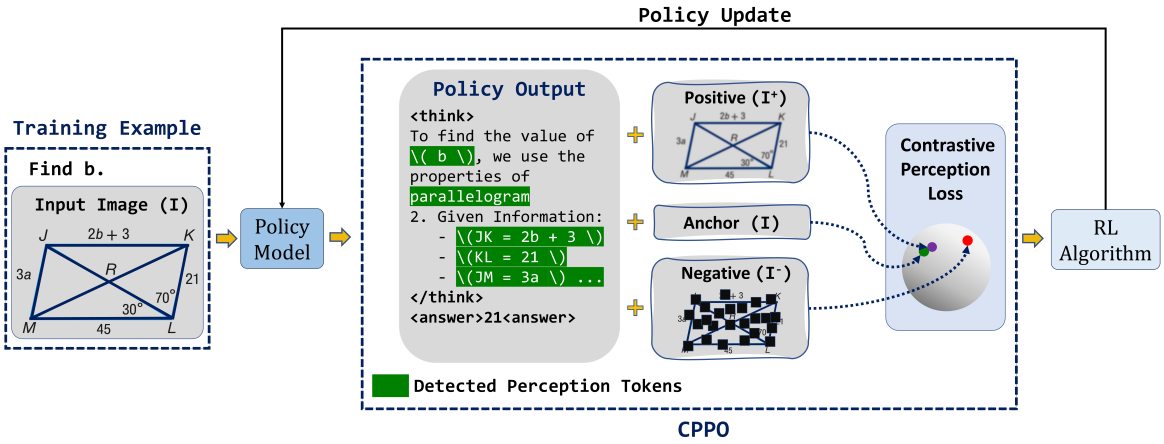


Figure 1. Illustration of the proposed **CPPO modification to the RL objective**. CPPO augments the RL objective with a perception-specific self-supervised contrastive loss and selectively applies it to tokens that are detected as perception-dependent. Improved perception is critical for vision–language agents that must *act* reliably in open-ended environments based on what they see.

output perception and reasoning tokens be disentangled for a VLM policy? 2) How to best define an explicit perception loss/reward?

To address the first question, Xia et al. (2025) and Li et al. (2025b) force the policy to separate perception from reasoning by defining specific generation tags: `<perception>` and `<think>`. However, forcing such a separation disrupts the natural reasoning process of the model, making it difficult to apply to many tasks (*e.g.*, with complex images). In addition, the process becomes vulnerable to reward hacking (where the model places the final answer in the perception section to maximize reward). Thus, we argue that perception and reasoning should be disentangled within the model's natural generation flow.

In order to address the second question, Visionary-R1 (Xia et al., 2025), Vision-SR1 (Li et al., 2025b), and Perception-R1 (Xiao et al., 2025) rely on an LLM and utilize either the policy's own perceptual outputs or ground-truth Chain-of-Thought (CoT) annotations to compute perception rewards. Such evaluation of perception outputs with LLMs still requires explicit separation of perception from reasoning, incurs computational overhead, and relies on unscalable CoT supervision. PAPO (Wang et al., 2025b) takes a different approach via a KL divergence loss between model outputs conditioned on the original and corrupted versions of the images. However, the KL divergence is unbounded, which can easily cause reward collapse and makes the method's hyperparameters sensitive. Moreover, PAPO applies the perception loss uniformly across all tokens and output rollouts, regardless of whether they correspond to perception or reasoning, or whether the outputs are correct or incorrect. Applying divergence over reasoning tokens leads to over-regularization, while maximizing divergence on wrong perception tokens effectively reinforces incorrect perception outputs.

Motivated by these observations, we propose **C**ontrastive **P**erception **P**olicy **O**ptimization (CPPO), an RL solution designed for VLMs. CPPO integrates two main components into the training process: (1) a mechanism that uses the policy's own output probability distribution to determine the tokens in a generated response that the policy most strongly considers as perception tokens in its current state, and (2) a token-level Contrastive Perception Loss (CPL) incorporated into the RL objective to enforce *differential sensitivity* to vision information. Specifically, in each training step, we compare the policy's entropy for each token within responses when the policy is conditioned on the original image as well as a perturbed image with information-removing augmentations. Tokens whose entropy increases the most under this perturbation are selected as perception tokens by the policy, since their distribution exhibits the highest mutual information with the image.

After identifying vision-dependent tokens in the policy's output, we compute the token-level CPL term. Unlike prior work, CPL is an *unsupervised* perception contrastive loss that does not require additional CoT supervision or proprietary models. Specifically, for each input image, we create two other variants: an information-preserving perturbation that retains query-relevant content and an information-removing perturbation that obscures such information. CPL is then implemented as an InfoNCE contrastive loss (Chen et al., 2020): the token probability distribution conditioned on the original image serves as the anchor, the distribution under the information-preserving perturbation as the positive, and the distribution under the information-removing perturbation as the negative sample. Crucially, the contrastive loss is applied only to perception tokens from *correct* rollouts, ensuring that anchors correspond to accurate

and verified perception tokens. This provides targeted perception feedback to the policy, thereby improving its visual grounding capability.

Relevance to agents in the wild. Although our experiments target standard multimodal reasoning benchmarks, the failure mode that CPPO targets—ungrounded or weakly-grounded perception tokens—is precisely the failure mode that compromises VLM agents acting on real visual inputs. A policy that has been explicitly optimized to be *differentially sensitive* to visual content is, by construction, less likely to invent visual facts when its observations are noisy, partially occluded, or adversarially perturbed—all of which are routine conditions in deployed agentic settings.

In summary, the major contributions of our work are as follows:

- We propose CPPO, an RL-based finetuning solution tailored for VLMs to disentangle perception and reasoning improvement of the policy, motivated by the perceptual reliability requirements of VLM agents.
- We propose CPL, an *unsupervised* perception-specific contrastive loss to optimize a VLM policy.
- We propose an entropy-based perception token detection method, where the VLM policy identifies its own perception tokens using its output distribution.
- We show the superiority of CPPO compared with prior perception-specific RL methods across math and visual reasoning benchmarks.

2. Related Work

In this section, we categorize the related RL methods proposed for VLMs into three directions: 1) sampling and rollout augmented methods, 2) RL combined with SFT or off-policy data, and 3) perception-aware approaches. Our approach falls into the third category, while the other directions are orthogonal to our method. We also discuss the background of using contrastive learning in RL.

Sampling and Rollout Augmented RL with VLMs. This line of work improves robustness and training efficiency by mixing trajectories from clean and moderately distorted images during RL training. NoisyRollout (Liu et al., 2025) and Vision Matters (Li et al., 2025a) use input perturbations to stabilize grounding and enhance generalization. Shuffle-R1 (Zhu et al., 2025) introduces pairwise trajectory sampling and advantage-based batch reshuffling to improve gradient signal quality and increase exposure to valuable rollouts. VL-Rethinker (Wang et al., 2025a) proposes selective sample replay to address the "vanishing advantages" problem and forced rethinking, which appends a trigger token to enforce self-reflective reasoning. This line of work is orthogonal to CPPO.

RL Combined with SFT or Off-Policy with VLMs. This line of research combines on-policy RL with off-policy CoT or SFT training. Vision-R1 (Huang et al., 2025b), Lookback (Yang et al., 2025b), OpenVLThinker (Deng et al., 2025), VisionThink (Yang et al., 2025a), and Shen et al. (2025) focus on semi-off-Policy RL with emphasis on rethinking, iterative pipelines, or off-policy data to enhance slow-thinking reasoning and overall training stability. Similar to the prior category, this line of work is also orthogonal to our work.

Perception-Aware RL with VLMs. This line of work aims to improve how VLM policies couple visual perception with reasoning. One direction adopts decoupled architectures, using a VLM for visual description and an LLM for reasoning, optimized jointly with RL (Guo et al., 2025; Gou et al., 2025). Another approach explicitly separates perception from reasoning tokens in the model output. Vision-SR1 (Li et al., 2025b) and Visionary-R1 (Xia et al., 2025) enforce special tags (e.g., <perception> and <think>) and use an LLM to evaluate perception tokens, while Perception-R1 (Xiao et al., 2025) instead leverages supervised CoT trajectories to assess perception within reasoning paths. However, these methods rely on additional models or supervision, increasing computational cost and limiting scalability. To avoid extra supervision, PAPO (Wang et al., 2025b) introduces an unsupervised KL divergence loss between outputs conditioned on original and corrupted images. However, the unbounded KL objective can cause instability and is applied uniformly across all tokens, potentially over-regularizing reasoning tokens. Recent work further studies token-level perception in multimodal RL, showing that only a subset of tokens strongly depend on visual evidence and that perception signals vary across trajectories (Huang et al., 2025a).

Contrastive Learning in RL. Contrastive learning has been explored in general RL literature to improve representation quality and sample efficiency, e.g., CURL (Laskin et al., 2020), SPR (Schwarzer et al., 2021), SODA (Hansen & Wang, 2021), and TACO (Zheng et al., 2023). Recently, contrastive methods have been adopted for preference alignment in LLMs, e.g., Contrastive Preference Learning (Hejna et al., 2024) and Contrastive Preference Optimization (CPO) (Xu et al., 2024). However, VLM policy optimization remains unexplored. Our approach introduces a token-level contrastive loss as a self-supervised approach to improve policy's perception quality.

Augmenting Input Image. Adding perturbation to image inputs is used in prior works to reduce object-hallucination (VCD by Leng et al. (2024)) and improve visual grounding (SeVa by Zhu et al. (2024)) at inference/alignment time. Other approaches improve vision encoder of VLMs

(Epistemic-Uncertainty Masking by Seo et al. (2025)) or inject image-grounded signals at decode time (MARINE by Zhao et al. (2024)). Unlike these methods, which are primarily training-free, post-hoc, or operate at the vision encoder level, CPPO embeds image perturbations into RL training and applies a token-level contrastive loss only to entropy-identified perception tokens to shape the policy's visual sensitivity during optimization rather than correcting it post hoc.

3. Method

In this section, we review RL with verifiable rewards and then elaborate our proposed unsupervised contrastive perception policy optimization and how it is selectively applied to perception-dependent tokens.

3.1. Preliminaries

Group Relative Policy Optimization (GRPO). GRPO (DeepSeek-AI, 2025) includes RL fine-tuning of the policy VLM π_θ with parameters θ on verifiable tasks. Given an input set $x = \{q, I\}$ including query q and image I, a group of G output trajectories (responses) $\{\mathbf{o}_1, \ldots, \mathbf{o}_G\} \sim \pi_\theta(\cdot \mid x)$ are sampled. Each output $\mathbf{o}_i$ consists of T tokens $\{o_{i,1}, \ldots, o_{i,t}, \ldots, o_{i,T}\}$ and receives a scalar reward $R(\mathbf{o}_i)$, typically reflecting correctness. Relative advantages are computed as:

$$A_i = \frac{R(\mathbf{o}_i) - mean\big(R(\mathbf{o}^{1:G})\big)}{std\big(R(\mathbf{o}^{1:G})\big)}, \tag{1}$$

where $i \in [1, G]$. The GRPO objective is then defined as:

$$\begin{aligned}\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{\mathbf{o}_i \sim \pi_{\theta_{old}}} \tfrac{1}{G} \sum_{i=1}^{G} \tfrac{1}{|\mathbf{o}_i|} \sum_{t=1}^{|\mathbf{o}_i|} \Big\{ \\ \min\big(r_{i,t}(\theta) A_i,\ \text{clip}(r_{i,t}(\theta), 1-\epsilon, 1+\epsilon)\, A_i\big) \\ - \beta\, \text{KL}[\pi_\theta \,\|\, \pi_{ref}] \Big\},\end{aligned} \tag{2}$$

where $r_{i,t}(\theta) = \pi_\theta(o_{i,t} \mid x, \mathbf{o}_{i,<t}) / \pi_{\theta_{old}}(o_{i,t} \mid x, \mathbf{o}_{i,<t})$ is the importance ratio. The KL penalty controls the deviation from the frozen reference policy π_{ref} with weight β. Output trajectories are generated by the rollout policy $\pi_{\theta_{old}}$, and the hyperparameter ϵ controls clipping large policy updates. In this setting, the correctness reward alone provides no explicit signal to enhance the policy model's perceptual sensitivity. Our CPL loss aims to address this gap.

3.2. CPPO: Contrastive Perception Policy Optimization

While prior work has explored guiding the policy model toward improved perceptual understanding by providing explicit vision rewards, our approach instead augments the RL objective function with a perception-dependent contrastive loss. Figure 2 and Algorithm 1 illustrate the overall framework. Inspired by contrastive representation learning (Chen et al., 2020), the central idea of CPPO is to encourage the policy to be *differentially sensitive* to visual perturbations in the input image at the *token level* by selectively applying a contrastive perception loss to perception-dependent tokens.

Algorithm 1 CPPO

Require: policy π_θ, rollout policy $\pi_{\theta_{\text{old}}}$, dataset $\mathcal{D}$, RL objective $\mathcal{J}$, contrastive perception loss weight λ, top-k ratio k
for each training step **do**
  Sample input $(q, I) \sim \mathcal{D}$
  Generate rollout $o_i \sim \pi_{\theta_{\text{old}}}(\cdot \mid q, I)$
  Compute reward $R(o_i)$ and advantage A_i
  for each rollout o_i **do**
    Create information-removing image I^-
    Compute entropy increase $\Delta H_{i,t}$ for all tokens
    Select top-k positive $\Delta H_{i,t}$ tokens: $\mathcal{S}_{\text{perception}}(o_i)$
  end for
  for each rollout o_i **do**
    Create information-preserving image I^+
    For all $t \in \mathcal{S}_{\text{perception}}(o_i)$:
      Compute contrastive loss $\mathcal{L}_{i,t}^{\text{InfoNCE}}$
    $\mathcal{L}_{\text{CPL}}(o_i) = \frac{1}{|\mathcal{S}_{\text{perception}}(o_i)|} \sum_t \mathcal{L}_{i,t}^{\text{InfoNCE}}$
  end for
  $\mathcal{J}(\theta) = \mathcal{J}(\theta) - \lambda \cdot \mathbf{1}[A_i > 0]\, \mathcal{L}_{\text{CPL}}(o_i)$
  Update policy $\theta \leftarrow \theta + \nabla_\theta \mathcal{J}(\theta)$
  Update rollout policy $\theta_{\text{old}} \leftarrow \theta$
end for

Contrastive Perception Loss (CPL). Given the generated response $\mathbf{o}_i$ for the input $x = \{q, I\}$, CPL operates by contrasting the policy's probability distribution of each token under perturbed views of the input image I. For each token $o_{i,t}$, CPL recomputes the probability distributions under two variants of the anchor image I:

- *Information-removing perturbations* I^-, obtained from transformations such as region masking or deletion of critical visual elements that obscure query-relevant information. The policy's output distribution under I^- should diverge from that of I.
- *Information-preserving perturbations* I^+, obtained from transformations such as mild Gaussian noise or small brightness shifts that do not remove query-relevant content. The output distribution under I^+ should remain consistent with that of I.

Formally, for a token $o_{i,t}$ in $\mathbf{o}_i$, we denote the policy probability distribution under the original image as the anchor

$$\pi_\theta(o_{i,t}) = \pi_\theta(o_{i,t} \mid q, I, \mathbf{o}_{i,<t}), \tag{3}$$

the distribution under I^+ as the positive sample

$$\pi_\theta^+(o_{i,t}) = \pi_\theta(o_{i,t} \mid q, I^+, \mathbf{o}_{i,<t}), \tag{4}$$

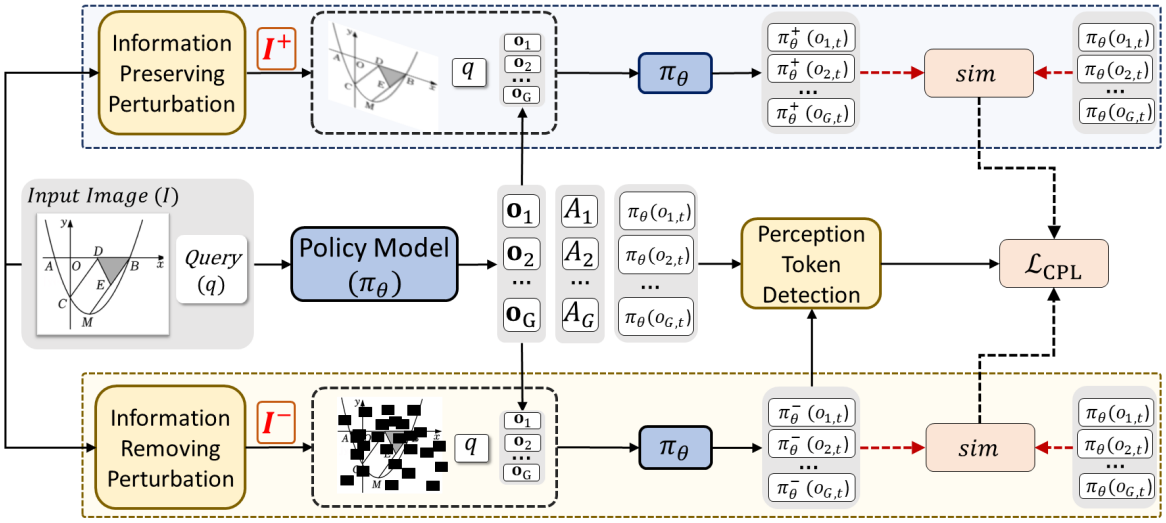


Figure 2. **Detailed overview of CPPO.** For each rollout $\mathbf{o}_i$, perception tokens are identified and their probability distributions are computed under three conditions: the original image I (anchor sample: $\pi_\theta(o_{i,t})$), an information-preserving perturbation I^+ (positive sample: $\pi_\theta^+(o_{i,t})$), and an information-removing perturbation I^- (negative sample: $\pi_\theta^-(o_{i,t})$). Similarities $sim\big(\pi_\theta(o_{i,t}), \pi_\theta^+(o_{i,t})\big)$ and $sim\big(\pi_\theta(o_{i,t}), \pi_\theta^-(o_{i,t})\big)$ are computed and incorporated into the Contrastive Perception Loss (CPL), which serves as an additional perception-specific term in the RL objective. Notations are simplified for brevity.

and the distribution under I^- as the negative sample

$$\pi_\theta^-(o_{i,t}) = \pi_\theta(o_{i,t} \mid q, I^-, \mathbf{o}_{i,<t}). \tag{5}$$

Let $sim(p, p^*) = -\mathrm{KL}(p \,\|\, p^*)$ denote the negative KL divergence as a similarity measure between token probability distributions, and let $s_{i,t}^{\pm} = sim\big(\pi_\theta(o_{i,t}), \pi_\theta^{\pm}(o_{i,t})\big)$. We adopt the InfoNCE loss (Chen et al., 2020) to define the contrastive objective:

$$\mathcal{L}_{o_{i,t}}^{\text{InfoNCE}} = -\log \frac{\exp(s_{i,t}^+/\tau)}{\exp(s_{i,t}^+/\tau) + \exp(s_{i,t}^-/\tau)}, \tag{6}$$

where $\tau > 0$ is a temperature hyperparameter. Minimizing this loss encourages the anchor distribution to remain close to the positive view while being pushed away from the negative view, thereby explicitly enforcing perceptual improvement in a fully unsupervised manner without relying on CoT annotations.

Perception Token Detection. Not all tokens in an output are equally dependent on perceptual input. For example, interpreting "the base is 10 cm" relies on visual information, whereas solving "$x^2 + 2x + 1 = 0$" or recalling that "the angles of a triangle sum to 180°" can be performed independently of the image. Applying CPL uniformly across all tokens may lead to excessive regularization and destabilize training. Therefore, we introduce a mechanism to selectively identify perception-dependent tokens using the model's own output distribution, and apply CPL only to these tokens.

Proposition 3.1 (Entropy increase as a proxy for perception dependence). *(Proof in Appendix A) Let I denote the original image, I^- a perturbed variant that removes query-relevant perceptual information, and $\mathbf{o}_i$ a sequence of tokens generated by the policy when conditioned on I. The increase in entropy of a token $o_{i,t} \in \mathbf{o}_i$, when the policy is conditioned on I^- rather than I, serves as a proxy for the degree to which the policy associates $o_{i,t}$ with the query-relevant visual content of I. The increase is calculated as:*

$$\Delta H_{i,t} = H(o_{i,t}|q, I^-, \mathbf{o}_{i,<t}) - H(o_{i,t}|q, I, \mathbf{o}_{i,<t}). \tag{7}$$

For token $o_{i,t}$, the predictive entropy is defined as:

$$H(o_{i,t}|x, \mathbf{o}_{i,<t}) = -\sum_{o_{i,t} \in \mathcal{V}} \pi_\theta(o_{i,t}|x, \mathbf{o}_{i,<t}) \log \pi_\theta(o_{i,t}|x, \mathbf{o}_{i,<t}), \tag{8}$$

where $\mathcal{V}$ denotes the vocabulary.

Perception-Topk. After generating $\mathbf{o}_i$ for image I, we construct I^- by applying a random information-removing perturbation and compute $\Delta H_{i,t}$ for each token. Tokens are ranked by $\Delta H_{i,t}$, and the top$k$ most perception-dependent tokens are retained:

$$\mathcal{S}_{\text{perception}} = \{\, t \mid \mathrm{Rank}(\Delta H_{i,t}) \le k \cdot T \,\}, \tag{9}$$

where k denotes the proportion of tokens with the highest entropy increase. We construct a binary mask $M_i \in \{0,1\}^T$:

$$M_{i,t} = \begin{cases} 1, & \text{if } t \in \mathcal{S}_{\text{perception}}, \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

Table 1. CPPO vs. GRPO. All results are based on avg@8. **Bold**: the best value in each column. $\Delta^{\%}_{rel}$ shows the relative improvement over the GRPO baseline averaged over all benchmarks.

	Math Benchmarks					Visual Reasoning			
Methods	**MVista$_m$**	**DMath**	**WeMath**	**MVision$_m$**	**MVerse**	**MMMU-P$_v$**	**LogicVista**	**AVG**	$\Delta^{\%}_{rel}$
Qwen2.5-VL-3B	56.4	33.7	14.5	19.5	25.7	19.9	32.4	28.8	-
GRPO-3B	63.7	45.7	28.4	25.1	38.3	25.8	37.7	37.8	-
CPPO-3B	**66.3**	**48.9**	**30.8**	**25.3**	**39.4**	**28.5**	**40.9**	**40.0**	**6.0**
Qwen2.5-VL-7B	65.6	53.2	33.3	24.5	41.2	33.7	45.1	42.3	-
GRPO-7B	71.2	55.6	42.4	27.6	45.0	37.9	47.4	46.7	-
CPPO-7B	**72.2**	**56.9**	**44.8**	**29.9**	**46.5**	**39.0**	**48.2**	**48.2**	**3.7**

CPL is then selectively applied as:

$$\mathcal{L}_{\text{CPL},i,t} = \begin{cases} \mathcal{L}^{\text{InfoNCE}}_{o_{i,t}} & \text{if } M_{i,t} = 1, \\ 0, & \text{if } M_{i,t} = 0. \end{cases} \tag{11}$$

The overall CPL for trajectory $\mathbf{o}_i$ is:

$$\mathcal{L}_{\text{CPL}}(\mathbf{o}_i; I, I^+, I^-) = \frac{1}{|\mathbf{o}_i|} \sum_{t=1}^{|\mathbf{o}_i|} \mathcal{L}_{\text{CPL},i,t}. \tag{12}$$

Integration with RL Objective. Finally, we integrate CPL with the GRPO objective. For each sampled trajectory $\mathbf{o}_i$, we compute the standard GRPO update (Equation (2)) together with the CPL term.

Advantage Gating Mechansim. To prevent low-quality trajectories from introducing noisy gradients, we use an advantage gating mechanism: CPL is applied only when the trajectory's advantage A_i is positive. Thus, the combined objective is:

$$\mathcal{J}(\theta) = \mathbb{E}_{\mathbf{o}_i \sim \pi_{\theta_{old}}} \Big[\mathcal{J}_{\text{GRPO}}(\theta) - \lambda \tfrac{1}{G} \sum_{i=1}^{G} \mathbf{1}\{A_i > 0\} \, \mathcal{L}_{\text{CPL}}(\mathbf{o}_i; I, I^+, I^-) \Big], \tag{13}$$

where $\mathbf{1}\{\cdot\}$ denotes the indicator function and λ controls the strength of the contrastive objective. By incorporating advantage gating, CPL is imposed only on trajectories that outperform the group baseline, ensuring that CPL regularization reinforces successful perceptions while avoiding incorrect trajectories.

4. Experiments

4.1. Experimental Setup

Training Dataset. We train on ViRL39K (Wang et al., 2025a), a dataset consisting of 38.8K multimodal question–answer pairs. The dataset spans a broad range of domains, including grade school problems to broader STEM and social topics; reasoning with charts, diagrams, tables, documents, and spatial relationships.

Evaluation. Following prior works, we use the following benchmarks for evaluation: LogicVista (Xiao et al., 2024), MathVista (Lu et al., 2024), DynaMath (Zou et al., 2025), WeMath (Qiao et al., 2024), MathVision (Wang et al., 2024), MathVerse (Zhang et al., 2024), and MMMU-Pro-Vision (Yue et al., 2025). These benchmarks encompass math, general multimodal reasoning, and logical reasoning tasks. All evaluations are performed using VLMEvalKit (Duan et al., 2024). We report average accuracy@8 with an inference temperature of 1.0 to provide a more consistent and reliable measure of model performance across all the experiments in the paper.

Baselines. We use Qwen2.5-VL-3B and 7B (Bai et al., 2025) as the backbone models in all our experiments. We compare our CPPO with recent RL methods proposed for VLMs: OpenVLThinker-3B/7B, Visionary-R1-3B, PAPO-3B/7B, VL-ReThinker-7B, Vision-Matters-7B, NoisyRollout-7B, Perception-R1-7B, Vision-SR1-7B, and Look-Back-7B (semantic checkpoint). All of these prior works use Qwen2.5-VL-3B/7B as the policy model.

Perturbation Types. Information-removing perturbations include random 80% patch-wise masking and random 30% cropping (retaining only 30% of the image) to obscure the majority of the visual content. For information-preserving perturbations, we apply lightweight color perturbations such as color jitter and Gaussian noise as well as mild geometry perturbations such as random perspective and random rotation. These perturbations modify the image appearance without eliminating critical information. At each training step, one augmentation is randomly sampled from each augmentation set. Samples of perturbations along with detailed parameter settings for all perturbations are provided in Appendix E.

Implementation Details. We use *verl* (Sheng et al., 2024) as our RL training framework. The policy models are initialized with Qwen2.5-VL-3B/7B. We train the policy model with GRPO and CPPO for 2 epochs on the ViRL39K dataset with a group size of 5 and a global batch size of 512. Both the vision encoder and LLM of the baselines were updated during training. For other RL-related hyperparameters, we

Table 2. CPPO vs. prior works for 3B and 7B models. All results are reported as avg@8. **Bold** indicates the best value in each column; underlined indicates the second best.

	Math Benchmarks					Visual Reasoning		
Methods	**MVista$_m$**	**DMath**	**WeMath**	**MVision$_m$**	**MVerse**	**MMMU-P$_v$**	**LogicVista**	**AVG**
GPT4-o	60.0	34.5	47.4	30.6	41.2	51.9	52.8	45.4
Gemini-2.0-Flash	73.4	42.1	45.8	41.3	54.6	51.7	52.3	51.6
OpenVLThinker-3B	60.0	35.6	26.3	22.3	36.9	25.0	37.4	34.7
Visionary-R1-3B	61.4	41.2	27.1	19.7	34.5	27.9	37.1	35.5
PAPO-3B	64.8	45.4	28.1	24.3	38.3	26.8	39.4	38.1
CPPO-3B	**66.3**	**48.9**	**30.8**	**25.3**	**39.4**	**28.5**	**40.9**	**40.0**
OpenVLThinker-7B	70.7	43.9	38.4	27.5	40.7	35.5	45.8	43.9
Vision-SR1-7B	67.0	52.6	33.6	28.0	40.7	38.9	43.2	43.9
Look-Back-7B	69.1	52.5	39.8	25.8	41.9	34.5	46.3	44.8
Vision-Matters-7B	68.6	54.5	40.1	25.2	45.3	35.5	45.1	45.3
PAPO-7B	71.6	54.7	39.5	26.5	44.5	38.7	45.8	46.8
PerceptionR1-7B	70.0	55.8	**45.4**	27.6	46.0	38.1	45.5	47.3
NoisyRollout-7B	71.1	55.9	44.4	29.4	46.4	38.5	47.9	47.7
CPPO-7B	**72.2**	**56.9**	44.8	**29.9**	**46.5**	**39.0**	**48.2**	**48.2**

use the default settings of *verl*. Our training settings match those of PAPO for a controlled comparison. More details are in Appendix C.

4.2. Main Results

Comparison to Baseline GRPO. Applying CPPO to the Qwen2.5-VL-3B and -7B baselines yields consistent and substantial improvements on the test benchmarks, with average absolute performance gains of 11.2% and 5.9%, respectively. As reported in Table 1, CPPO achieves a higher accuracy than GRPO across all benchmarks—average relative gains of 6.0% for the 3B model and 3.7% for the 7B model. Overall, these results confirm that CPPO is a more effective optimization strategy than GRPO, especially for mid-sized models, and establishes CPPO as a strong and scalable alternative for finetuning large VLMs. Qualitative results are given in Appendix F.

Comparison to Other Methods. As shown in Table 2, CPPO consistently surpasses prior methods across all benchmarks for the 3B model. For the 7B model, CPPO also outperforms existing approaches on all benchmarks (except WeMath), demonstrating stronger generalization. In particular, when compared to PAPO—the most relevant perception-aware RL baseline—CPPO achieves notable gains. On the 3B model, CPPO improves average performance to 40.0%, compared to PAPO's 38.1%. On the larger 7B model, CPPO reaches 48.2% versus PAPO's 46.8%. Importantly, both CPPO and PAPO are trained under identical conditions—using the same dataset (ViRL39K) and the same number of training steps—ensuring that the improvements are not due to differences in data or compute. Thus, the consistent advantage of CPPO over PAPO can be attributed directly to the introduction of contrastive loss on perception tokens, which enhances the model's ability to capture and leverage visual information more effectively. The benefit of stronger visual grounding is what we expect to translate into improved reliability for downstream agentic tasks.

Out-of-Domain Performance. Our main training is conducted on the ViRL39K (Wang et al., 2025a) dataset, a comprehensive dataset covering a wide range of topics. To further evaluate the out-of-distribution performance of CPPO compared to GRPO, we additionally train Qwen2.5-VL-3B on the Geometry3K dataset (Lu et al., 2021), a small dataset with 2.1K samples focused on geometry problems. We train the model using both GRPO and CPPO and evaluate the resulting models on out-of-distribution benchmarks. Figure 3 shows the training dynamics (Training Reward), the reward on the in-domain validation set (Geometry3K Validation Reward), and the accuracy on out-of-distribution benchmarks as training progresses. The training reward indicates that CPPO leads to faster learning and stronger generalization from the early stages of training.

Statistical Significance. Figure 3 shows one-standard-deviation bands for all benchmarks, demonstrating that the gains of CPPO over GRPO are consistent across different evaluations. To further validate this observation, we conduct paired t-tests for each benchmark. All tests yield *p-values* ≤ 0.02, confirming that the improvements of CPPO over GRPO are statistically significant.

Performance of Perception Token Detection. Figure 4 shows two samples, the policy model's outputs generated by CPPO, and the top 40% of perception tokens identified using our entropy-based method. In the 1st example, the question asks for the value of angle $\angle CAD$ in a geometry problem. The key visual clues needed to solve this question are: (1) $\angle CDA = \angle 1 = 40°$, (2) CAD forms a triangle, and (3)

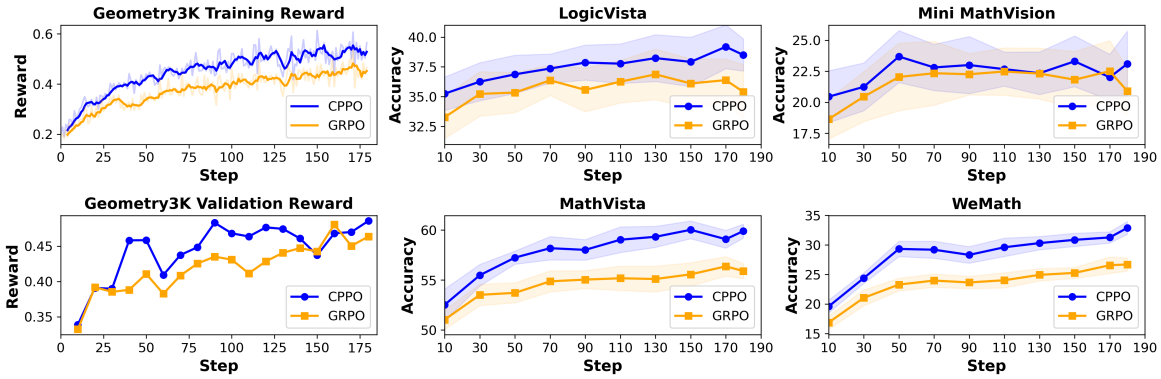


Figure 3. CPPO vs. GRPO (avg@8) on Qwen2.5-VL-3B across in-domain and out-of-domain scenarios. The X-axis represents RL training steps. The shaded area corresponds to one standard deviation over 8 responses. **1st column:** Reward comparison on the in-domain dataset during training. **2nd and 3rd columns:** Comparison on four out-of-domain visual reasoning benchmarks.

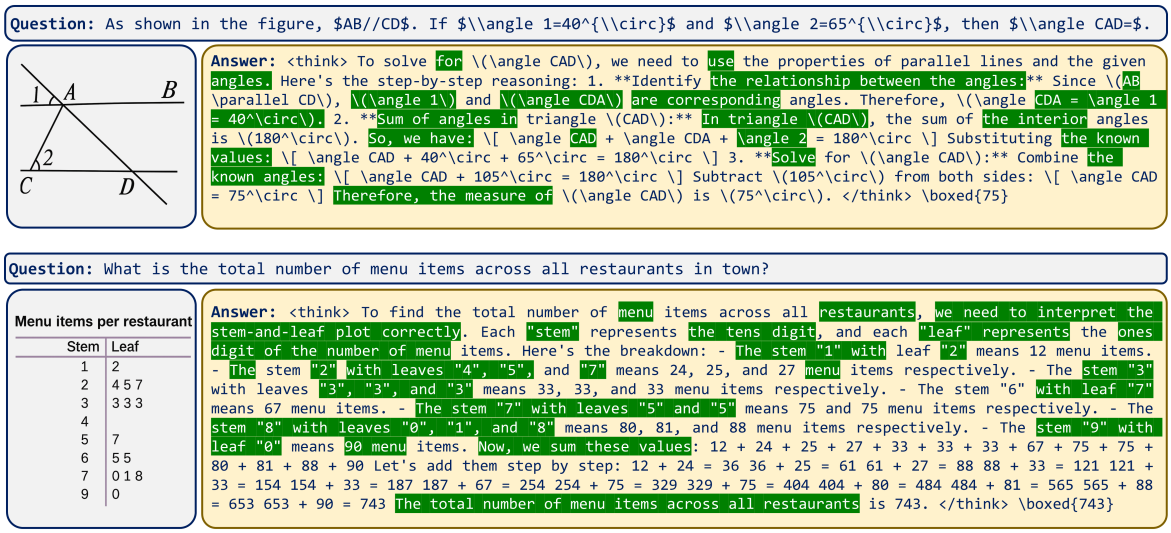


Figure 4. Sample outputs generated with CPPO with top 40% detected perception tokens.

$\angle 2 = \angle ACD$. With these three pieces of information alone, one could solve the problem without referring back to the original figure. We observe that all these critical elements are successfully highlighted within the top 40% of selected perception tokens. The 2nd example shows a stem and leaf plot summarizing the number of menu items per restaurant in a town, which is used to answer a question. Here, we find that most of the relevant numerical values are also captured within the top 40% of detected perception tokens, illustrating that the method effectively identifies the essential visual information for the question. More analysis is given in Appendix B.

4.3. Analysis of Selected Perturbations

Validating the Selected Perturbations. To assess whether the selected perturbations effectively preserve or remove information in images, we applied each perturbation to the images in four benchmarks. The Qwen2.5-VL-7B base model was then tested on every modified benchmark to measure changes in task performance attributable to these perturbations. As shown in Table 3, information-preserving augmentations reduced average model performance by less than 1.5%, indicating that, for the majority of images, the critical information remains intact. In contrast, information-removing perturbations resulted in a substantial drop in average accuracy, exceeding 14% across both removal strategies, indicating the elimination of salient information needed to answer the questions.

Impact of Selected Perturbations. We analyzed the effect of selected perturbations in the CPPO pipeline by training Qwen2.5-VL-3B on Geometry3K dataset (Lu et al., 2021) with different selected perturbations. As shown in the Table 4, a more diverse perturbation set provides a richer learning signal, hence leading to better performance.

Table 3. Validating selected info preserving/removing perturbation.

Image Perturbation Type	LogicVista	$MVista_m$	$MVision_m$	WeMath	AVG
Original Images	45.1	65.6	24.5	33.3	42.1
Information Preserving					
Color Jitter	44.5	64.3	24.6	31.6	41.3
Gaussian Blur	44.2	64.4	25.3	31.6	41.4
Random Perspective	44.4	64.0	25.3	31.1	41.2
Random Rotation	43.9	63.4	24.0	31.4	40.7
Information Removing					
Random Occlusion	31.3	40.9	19.3	13.8	26.3
Random Zoom Crop	31.2	42.9	20.1	15.1	27.3

Table 4. Impact of the perturbations used in CPPO. Co: Color, Ge: Geometry, Cr: Crop, Ma: Mask.

Perturb(+)	Perturb(-)	LogicVista	$MVista_m$	$MVision_m$	WeMath	AVG
Co	(Cr, Ma)	37.7	56.9	22.4	31.1	37.0
Ge	(Cr, Ma)	37.8	58.7	22.0	32.5	37.8
(Ge, Co)	Ma	38.7	58.8	22.6	32.3	38.1
(Ge, Co)	Cr	38.3	59.8	23.1	32.8	38.5
(Ge, Co)	(Cr, Ma)	**38.5**	**59.9**	**23.1**	**32.9**	**38.6**

Table 5. Increase in training time and performance when training with CPPO for 2 epochs vs. GRPO for 4 epochs, relative to training with GRPO for 2 epochs. Even a 100% increase in GRPO time does not match the performance achieved by CPPO.

	$MVista_m$	WeMath	$MVision_m$	LogicVista	AVG
GRPO: 2 Epochs	63.7	28.4	25.1	37.7	38.7
GRPO: 4 Epochs (100%)	65.8	28.5	**25.4**	38.2	39.5
CPPO: 2 Epochs (39%)	**66.3**	**30.8**	**25.4**	**39.4**	**40.5**

4.4. Complexity Analysis

Compared to GRPO, CPPO adds additional computation due to two extra forward passes required to compute token distributions conditioned on the positive (I^+) and negative (I^-) images. This increases the time per training step. As shown in Table 5, Training Qwen2.5-VL-3B with CPPO for two epochs takes 39% more time than GRPO under identical resources. One may argue that the same compute could instead be used to train GRPO longer. To test this, we trained Qwen2.5-VL-3B with GRPO for two additional epochs on ViRL39K (Wang et al., 2025a). This doubles the training time (100% increase) yet still underperforms CPPO trained for only two epochs (CPPO@2 epochs averages 40.5% vs. GRPO@4 epochs at 39.5%). These results indicate that CPPO's gains arise from the contrastive learning signal rather than increased compute.

4.5. Ablations

We adopt Qwen2.5-VL-3B as the baseline and conduct all ablations on the Geometry3K (Lu et al., 2021), which contains 2.1K samples. We select Geometry3K both to enable faster training and to demonstrate generalizability of CPPO.

Loss Weighting (λ). We experiment with different λ values in Equation (13). λ controls the strength of perceptual grounding. As given in Table 6, the best performance is obtained with $\lambda = 0.02$, achieving 38.6% average accuracy. In general, CPPO with different λ values outperforms GRPO with an average accuracy of 34.7%.

Table 6. Ablation on λ values.

λ	LogicVista	$MVista_m$	$MVision_m$	WeMath	AVG
0.01	37.4	59.2	21.9	31.4	37.5
0.02	38.5	59.9	23.1	32.9	**38.6**
0.03	38.6	57.8	22.9	28.8	37.0
0.04	35.6	55.9	21.7	27.6	35.2

Table 7. Ablation on top-k perception tokens.

K	LogicVista	$MVista_m$	$MVision_m$	WeMath	AVG
5%	32.5	52.2	21.4	20.1	31.6
25%	36.7	57.9	22.7	30.7	37.0
50%	38.5	59.9	23.1	32.9	**38.6**
75%	37.6	57.4	22.3	29.1	36.6
100%	36.3	56.9	22.0	29.5	36.2

Top$k$. Table 7 presents the analysis of different K values for top$k$ perception token detection. The results show that average accuracy improves as K increases from 5% to 50%, but declines when K is further expanded from 50% to 100%. We hypothesize that this trend arises because larger K values include more tokens that the policy model is already confident about (i.e., tokens with lower entropy change), which are less informative perception tokens. Incorporating these tokens can slow down the training and ultimately lead to worse performance when models are trained for the same number of epochs. More ablations on different components of CPPO are given in Appendix D.

5. Conclusion

In this work, we introduced CPPO, a perception-aware RL-based method for finetuning VLMs. CPPO leverages an entropy-based approach to disentangle *perception* tokens from *reasoning* tokens, where perception tokens capture visual information extracted from the input image. To better align training with perception quality, we proposed a Contrastive Perception Loss (CPL)—an unsupervised, model-free objective that penalizes perception errors. Extensive experiments demonstrate that CPPO outperforms recent RL methods for VLMs, achieving state-of-the-art performance across multiple math and visual reasoning benchmarks. Because perception failures are a primary driver of unsafe behavior in vision–language agents, we believe a self-supervised perception-aware RL recipe like CPPO is a promising building block for training agents that must operate reliably in open-ended environments. Limitations and additional analysis are discussed in the Appendix.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning, with particular focus on improving the perceptual reliability of vision–language model (VLM) agents deployed in open-ended environments. Better visual grounding directly reduces a known failure mode for agents that must reason and act in open-ended environments: hallucinated or careless perception of the scene. We see no specific risks of this work beyond those already associated with the broader development of capable multimodal agents.

References

Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., Zhong, H., Zhu, Y., Yang, M., Li, Z., Wan, J., Wang, P., Ding, W., Fu, Z., Xu, Y., Ye, J., Zhang, X., Xie, T., Cheng, Z., Zhang, H., Yang, Z., Xu, H., and Lin, J. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Bi, J., Liang, S., Zhou, X., Liu, P., Guo, J., Tang, Y., Song, L., Huang, C., Sun, G., He, J., et al. Why reasoning matters? a survey of advancements in multimodal reasoning (v1). *arXiv preprint arXiv:2504.03151*, 2025.

Cannons, K., Alvar, S. R., Hossain, M. A., Rezaei, A., Gholami, M., Heidarikhazaei, A., Weimin, Z., Zhang, Y., and Akbari, M. From segments to scenes: Temporal understanding in autonomous driving via vision-language model. *arXiv preprint arXiv:2512.05277*, 2025.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PmLR, 2020.

Chen, Z., Wu, J., Wang, W., Su, W., Chen, G., Xing, S., Zhong, M., Zhang, Q., Zhu, X., Lu, L., et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24185–24198, 2024.

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Deng, Y., Bansal, H., Yin, F., Peng, N., Wang, W., and Chang, K.-W. Openvlthinker: Complex vision-language reasoning via iterative sft-rl cycles, 2025. URL https://arxiv.org/abs/2503.17352.

Duan, H., Yang, J., Qiao, Y., Fang, X., Chen, L., Liu, Y., Dong, X., Zang, Y., Zhang, P., Wang, J., et al. Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 11198–11201, 2024.

Gholami, M., Rezaei, A., Weimin, Z., Mao, S., Zhou, S., Zhang, Y., and Akbari, M. Spatial reasoning with vision-language models in ego-centric multi-view scenes. *arXiv preprint arXiv:2509.06266*, 2025.

Gou, Y., Chen, K., Liu, Z., Hong, L., Jin, X., Li, Z., Kwok, J. T., and Zhang, Y. Perceptual decoupling for scalable multi-modal reasoning via reward-optimized captioning. *arXiv preprint arXiv:2506.04559*, 2025.

Guo, Z., Liu, M., Ji, Z., Bai, J., Zhang, L., and Zuo, W. Decoupled visual interpretation and linguistic reasoning for math problem solving. *arXiv preprint arXiv:2505.17609*, 2025.

Hansen, N. and Wang, X. Generalization in reinforcement learning by soft data augmentation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 13611–13617. IEEE, 2021.

Hejna, J., Rafailov, R., Sikchi, H., Finn, C., Niekum, S., Knox, W. B., and Sadigh, D. Contrastive preference learning: Learning from human feedback without reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=iX1RjVQODj.

Huang, S., Qu, X., Li, Y., Luo, Y., He, Z., Liu, D., and Cheng, Y. Spotlight on token perception for multimodal reinforcement learning. *arXiv preprint arXiv:2510.09285*, 2025a.

Huang, W., Jia, B., Zhai, Z., Cao, S., Ye, Z., Zhao, F., Xu, Z., Hu, Y., and Lin, S. Vision-r1: Incentivizing reasoning capability in multimodal large language models, 2025b. URL https://arxiv.org/abs/2503.06749.

Laskin, M., Srinivas, A., and Abbeel, P. CURL: Contrastive unsupervised representations for reinforcement learning. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5639–5650. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/laskin20a.html.

Leng, S., Zhang, H., Chen, G., Li, X., Lu, S., Miao, C., and Bing, L. Mitigating object hallucinations in large vision-language models through visual contrastive decoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13872–13882, 2024.

Li, Y., Wei, L., Zheng, K., Huang, J., Kong, L., Sun, L., and Huang, W. Vision matters: Simple visual perturbations can boost multimodal math reasoning. *arXiv preprint arXiv:2506.09736*, 2025a.

Li, Z., Yu, W., Huang, C., Liu, R., Liang, Z., Liu, F., Che, J., Yu, D., Boyd-Graber, J., Mi, H., and Yu, D. Self-rewarding vision-language model via reasoning decomposition, 2025b. URL https://arxiv.org/abs/2508.19652.

Liu, X., Ni, J., Wu, Z., Du, C., Dou, L., Wang, H., Pang, T., and Shieh, M. Q. Noisyrollout: Reinforcing visual reasoning with data augmentation, 2025. URL https://arxiv.org/abs/2504.13055.

Lu, P., Gong, R., Jiang, S., Qiu, L., Huang, S., Liang, X., and Zhu, S.-C. Inter-GPS: Interpretable geometry problem solving with formal language and symbolic reasoning. In Zong, C., Xia, F., Li, W., and Navigli, R. (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 6774–6786, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.528. URL https://aclanthology.org/2021.acl-long.528/.

Lu, P., Bansal, H., Xia, T., Liu, J., Li, C., Hajishirzi, H., Cheng, H., Chang, K.-W., Galley, M., and Gao, J. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations (ICLR)*, 2024.

Qiao, R., Tan, Q., Dong, G., Wu, M., Sun, C., Song, X., GongQue, Z., Lei, S., Wei, Z., Zhang, M., Qiao, R., Zhang, Y., Zong, X., Xu, Y., Diao, M., Bao, Z., Li, C., and Zhang, H. We-math: Does your large multimodal model achieve human-like mathematical reasoning?, 2024. URL https://arxiv.org/abs/2407.01284.

Schwarzer, M., Anand, A., Goel, R., Hjelm, R. D., Courville, A., and Bachman, P. Data-efficient reinforcement learning with self-predictive representations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=uCQfPZwRaUu.

Seo, H., Kang, D. U., Cho, H., Lee, J., and Chun, S. Y. On epistemic uncertainty of visual tokens for object hallucinations in large vision-language models. *arXiv preprint arXiv:2510.09008*, 2025.

Shen, J., Zhao, H., Gu, Y., Gao, S., Liu, K., Huang, H., Gao, J., Lin, D., Zhang, W., and Chen, K. Semi-off-policy reinforcement learning for vision-language slow-thinking reasoning, 2025. URL https://arxiv.org/abs/2507.16814.

Sheng, G., Zhang, C., Ye, Z., Wu, X., Zhang, W., Zhang, R., Peng, Y., Lin, H., and Wu, C. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv:2409.19256*, 2024.

Wang, H., Qu, C., Huang, Z., Chu, W., Lin, F., and Chen, W. Vl-rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. *arXiv preprint arXiv:2504.08837*, 2025a.

Wang, K., Pan, J., Shi, W., Lu, Z., Ren, H., Zhou, A., Zhan, M., and Li, H. Measuring multimodal mathematical reasoning with math-vision dataset. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. URL https://openreview.net/forum?id=QWTCcxMpPA.

Wang, Z., Guo, X., Stoica, S., Xu, H., Wang, H., Ha, H., Chen, X., Chen, Y., Yan, M., Huang, F., et al. Perception-aware policy optimization for multimodal reasoning. *arXiv preprint arXiv:2507.06448*, 2025b.

Xia, J., Zang, Y., Gao, P., Li, Y., and Zhou, K. Visionary-r1: Mitigating shortcuts in visual reasoning with reinforcement learning, 2025. URL https://arxiv.org/abs/2505.14677.

Xiao, T., Xu, X., Huang, Z., Gao, H., Liu, Q., Liu, Q., and Chen, E. Advancing multimodal reasoning capabilities of multimodal large language models via visual perception reward. *arXiv preprint arXiv:2506.07218*, 2025.

Xiao, Y., Sun, E., Liu, T., and Wang, W. Logicvista: Multimodal llm logical reasoning benchmark in visual contexts, 2024. URL https://arxiv.org/abs/2407.04973.

Xu, H., Sharaf, A., Chen, Y., Tan, W., Shen, L., Van Durme, B., Murray, K., and Kim, Y. J. Contrastive preference optimization: Pushing the boundaries of LLM performance in machine translation. In Salakhutdinov, R., Kolter, Z., Heller, K., Weller, A., Oliver, N., Scarlett, J., and Berkenkamp, F. (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 55204–55224. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/xu24t.html.

Yang, S., Li, J., Lai, X., Yu, B., Zhao, H., and Jia, J. Visionthink: Smart and efficient vision language model via reinforcement learning. *arXiv preprint arXiv:2507.13348*, 2025a.

Yang, S., Niu, Y., Liu, Y., Ye, Y., Lin, B., and Yuan, L. Look-back: Implicit visual re-focusing in mllm reasoning, 2025b. URL https://arxiv.org/abs/2507.03019.

Yang, Y., He, X., Pan, H., Jiang, X., Deng, Y., Yang, X., Lu, H., Yin, D., Rao, F., Zhu, M., et al. R1-onevision: Advancing generalized multimodal reasoning through cross-modal formalization. *arXiv preprint arXiv:2503.10615*, 2025c.

Yue, X., Zheng, T., Ni, Y., Wang, Y., Zhang, K., Tong, S., Sun, Y., Yu, B., Zhang, G., Sun, H., Su, Y., Chen, W., and Neubig, G. MMMU-pro: A more robust multi-discipline multimodal understanding benchmark. In Che, W., Nabende, J., Shutova, E., and Pilehvar, M. T. (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15134–15186, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.736. URL https://aclanthology.org/2025.acl-long.736/.

Zhang, R., Jiang, D., Zhang, Y., Lin, H., Guo, Z., Qiu, P., Zhou, A., Lu, P., Chang, K.-W., Gao, P., and Li, H. Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? In *arXiv*, 2024.

Zhao, L., Deng, Y., Zhang, W., and Gu, Q. Mitigating object hallucination in large vision-language models via image-grounded guidance. *arXiv preprint arXiv:2402.08680*, 2024.

Zheng, R., Wang, X., Sun, Y., Ma, S., Zhao, J., Xu, H., Daumé III, H., and Huang, F. Taco:temporal latent action-driven contrastive loss for visual reinforcement learning. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 48203–48225. Curran Associates, Inc., 2023.

Zhu, K., Zhao, L., Ge, Z., and Zhang, X. Self-supervised visual preference alignment. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 291–300, 2024.

Zhu, L., Guan, Y., Liang, D., Ju, J., Luo, Z., Qin, B., Luan, J., Liu, Y., and Bai, X. Shuffle-r1: Efficient rl framework for multimodal large language models via data-centric dynamic shuffle, 2025. URL https://arxiv.org/abs/2508.05612.

Zou, C., Guo, X., Yang, R., Zhang, J., Hu, B., and Zhang, H. Dynamath: A dynamic visual benchmark for evaluating mathematical reasoning robustness of vision language models, 2025. URL https://arxiv.org/abs/2411.00836.

A. Proof for Proposition 1

Proposition 1 (Entropy increase as a proxy for vision dependence). *Let I denote the original image, I^- a perturbed variant that removes query-relevant perceptual information, and $\mathbf{o}_i$ the sequence of tokens generated by the policy when conditioned on I. The increase in entropy of a token $o_{i,t} \in \mathbf{o}_i$, when the policy is conditioned on I^- rather than I, serves as a proxy for the degree to which the policy model associates $o_{i,t}$ with the query-relevant visual content of I. This increase is calculated as follows:*

$$\Delta H_{i,t} = H(o_{i,t}|q, I^-, \mathbf{o}_{i,<t}) - H(o_{i,t}|q, I, \mathbf{o}_{i,<t}).$$

Proof. Recall the identity relating conditional mutual information (denoted by MI) and conditional entropy:

$$\begin{aligned} &H(o_{i,t} \mid X, q, \mathbf{o}_{i,<t}) \;= \\ &H(o_{i,t} \mid q, \mathbf{o}_{i,<t}) \;-\; MI(o_{i,t}; X \mid q, \mathbf{o}_{i,<t}))\,. \end{aligned} \tag{14}$$

Applying this with both $X = I$ and $X = I^-$ and subtracting, we obtain

$$\begin{aligned} &H_{i,t}(I^-) - H_{i,t}(I) \;= \\ &H\big(o_{i,t} \mid I^-, q, \mathbf{o}_{i,<t}\big) - H(o_{i,t} \mid I, q, \mathbf{o}_{i,<t}) \;= \\ &MI(o_{i,t}; I \mid q, \mathbf{o}_{i,<t}) \;-\; MI\big(o_{i,t}; I^- \,\big|\, q, \mathbf{o}_{i,<t}\big)\,. \end{aligned} \tag{15}$$

I^- is obtained from I by an information-removing augmentation that obscures query-relevant visual information. Our main assumption is that the conditional mutual information between perception tokens in $\mathbf{o}_i$ and I should be greater than their conditional mutual information with the perturbed image I^-. Formally, if $o_{i,t}$ is a perception token, we assume the following inequality holds for its conditional mutual information:

$$MI(o_{i,t}; I \mid q, \mathbf{o}_{i,<t}) - MI\big(o_{i,t}; I^- \,\big|\, q, \mathbf{o}_{i,<t}\big) \;\geq\; 0. \tag{16}$$

Substituting this inequality into (15) yields

$$H_{i,t}(I^-) - H_{i,t}(I) \;\geq\; 0. \tag{17}$$

Thus, an increase in predictive entropy, $\Delta H_{i,t}$, serves as a principled proxy for identifying vision-dependent tokens in the output sequence. □

B. Analysis on Performance of Perception Token Detection

To quantitatively evaluate our perception detection method, we used the inference outputs of Qwen2.5-VL-3B and -7B on four test sets: MathVista-MINI, LogicVista, MathVision-MINI, and WeMath. We then passed these outputs to GPT5-mini, which was used to separate the perception-related information from the rest of the model's response. This extracted perception information serves as our ground truth. We measure the accuracy of our detection method by calculating the ROUGE-1 F1 score between the detected perception tokens and the GPT5-mini outputs. It is important to note that GPT5-mini's separation is not flawless; thus, this evaluation should be viewed as a proof-of-concept rather than a definitive benchmark. Figure 5 shows that the ROUGE-1 F1 score improves as we increase the number of top$k$ perception tokens, up to the point where 100% of perception tokens are included. Here, 100% refers to selecting all tokens with positive ΔH in Proposition 1, rather than all output tokens. At each top$k$ percentage, we also select the same number of tokens randomly to serve as a baseline. Figure 5 shows that there is significant gap between our entropy-based method and random selection.

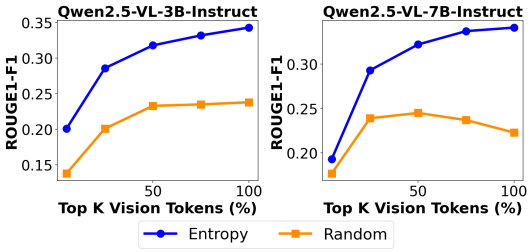


Figure 5. Quantitative evaluation of perception token detection.

C. Extra Training Details

Table 8 shows the summary of hyper-parameters used in training of 3B and 7B models.

Table 8. Summary of training hyperparameter configurations.

Parameter	Configuration
Main Results	
Model Base	Qwen2.5-VL-Instruct
Global Batch Size	512
Rollout Temperature	1.0
Learning Rate	$1e^{-6}$
Rollout Number	5
Training Epochs	2
Optimizer	AdamW
Policy Loss Aggregation	`token-mean`
β	0.01
τ	0.1
k	50%
λ	0.02
Ablations Specific	
Dataset	Geometry3K
Training Epochs	12
Global Batch Size	128

D. Ablation on Main Components of CPPO

Table 9 reports the ablation study on the key components of CPPO. Starting from GRPO, applying CPL to all tokens raises the average accuracy from 34.7% to 35.0%. Restricting CPL to only the top 50% of perception tokens yields a larger gain, increasing accuracy to 36.6%. Finally, introducing advantage gating—where the contrastive loss is applied only to rollouts with positive advantage—further improves performance to 38.6%. These results highlight that each component makes a meaningful contribution, and together they account for the overall effectiveness of CPPO.

Table 9. Ablation on the components of CPPO.

Methods	LogicVista	MVista_{m}	$\text{MVision}_{\text{m}}$	WeMath	AVG
Qwen2.5-VL-3B	32.4	56.4	19.5	14.5	30.7
GRPO	35.4	55.9	20.9	26.7	34.7
+ CPL on All Tokens	35.6	56.0	20.8	27.2	35.0
+ CPL on Top$k$ Perc. Tokens	36.4	56.6	22.5	30.9	36.6
+ Advantage Gating	38.5	59.9	23.1	32.9	**38.6**

E. Image Perturbation Details

The information-removing perturbations, such as random occlusion and random zoom crop, eliminate key visual details necessary for understanding the image. In contrast, the information-preserving perturbations—including color jitter, random perspective, random rotation, and Gaussian blur—modify the image without discarding critical information. Table 10 shows the Torchvision parameters selected for each perturbation. Samples of information-removing and information-preserving perturbations used in CPPO pipeline are also shown in Figure 6.

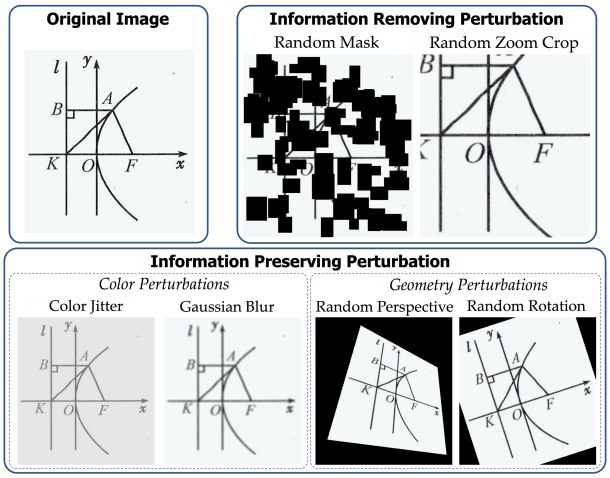


Figure 6. Samples of information-removing and information-preserving perturbations used in CPPO pipeline.

Table 10. Selected image perturbation parameters.

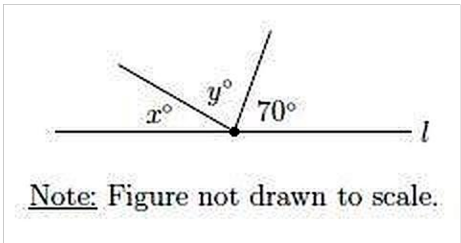
Perturbation	Parameters
Color Jitter	Brightness: $(0.2, 1.3)$ Contrast: $(0.2, 1.8)$ Saturation: $(0.2, 1.8)$
Random Perspective	Distortion Scale: 0.2
Random Rotation	Degrees: 10
Gaussian Blur	Kernel Size: 3
Random Occlusion	80% Patch-wise Masking
Random Zoom Crop	Retain 30% of Image

F. Qualitative Results

Figures 7–9 show three qualitative examples. We observe that CPPO has corrected the perception mistakes of models trained with GRPO. For example, in Figure 7, the model trained with GRPO states that "the angle x is given as 70 degrees" that is a wrong perception information extracted from the image. However, the model trained with CPPO corrected this statement by "The two line segments form angles that add up to 180 degrees". Note that when perception tokens are wrong, even with correct reasoning trajectory, the final answer is wrong.

G. Limitations

This work has several limitations that should be addressed in future research. First, due to our computational constraints, we did experiments up to 3B and 7B models. Exploring larger VLMs, such as 72B models, is an important direction for future work. Second, our evaluation was limited to Qwen2.5-VL baselines; extending the analysis to other baselines, such as InternVL (Chen et al., 2024), would provide a more comprehensive comparison. While we demonstrated the effectiveness of CPPO using 40K training samples, future studies should investigate large-scale training with substantially larger datasets. Finally, our experiments target standard multimodal reasoning benchmarks; we did not yet evaluate CPPO-trained policies inside actual agent harnesses (web/computer-use environments, robotic-style perception loops). We hypothesize that the improvements in visual grounding will translate into reduced hallucinated actions in such settings, but leave that empirical evaluation to future work.



Hint: Please answer the question requiring an integer answer and provide the final value, e.g., 1, 2, 3, at the end.
Question: In the figure above, two line segments meet at a point on line l. If the value of y is equal to the square of the value of x, what is the value of y?

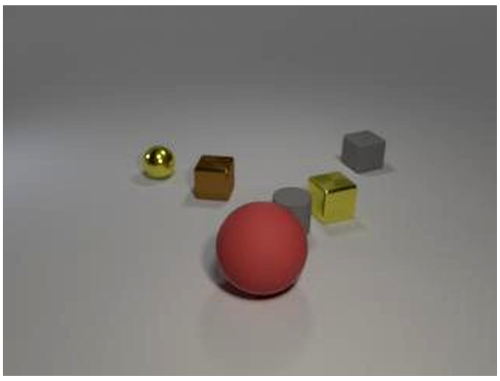
CPPO

<think>
The two line segments form angles that add up to 180 degrees because they are supplementary angles. This means that the sum of their measures is 180 degrees. Therefore, we can write the following equation:
\[x^\circ + y^\circ + 70^\circ = 180^\circ \]
Since \(y = x^2 \), we can substitute \(y \) in the equation:
\[x^\circ + x^2^\circ + 70^\circ = 180^\circ \]
Simplifying, we get:
\[x^2^\circ + x^\circ = 110^\circ \]
This is a quadratic equation that we can solve for \(x \):
\[x^2^\circ + x^\circ - 110^\circ = 0 \]
This equation can be solved using the quadratic formula, but for simplicity, we can guess and check values of \(x \) that are integers, because if \(y = x^2 \), and \(x \) and \(y \) are angles, they must be less than 180 degrees. We can test some values:
If \(x = 10 \):
\[10^2^\circ + 10^\circ = 100^\circ + 10^\circ = 110^\circ \]
This satisfies the equation. So, \(x = 10 \) and \(y = x^2 = 10^2 = 100 \).
Therefore, the value of \(y \) is 100.
</think>
\boxed{100}

GRPO

<think>
We are given that the angle y is equal to the square of the angle x, and the angle x is given as 70 degrees. Therefore, we can set up the following equation:
y = x^2
Substituting 70 for x, we get:
y = 70^2
y = 4900
Therefore, the value of y is 4900.
</think>
\boxed{4900}

Figure 7. Sample generated responses by CPPO and GRPO. GRPO exhibits a perception error that is corrected in the CPPO response.



Hint: Please answer the question requiring an integer answer and provide the final value, e.g., 1, 2, 3, at the end.
Question: Subtract all brown cubes. Subtract all gray cylinders. How many objects are left?

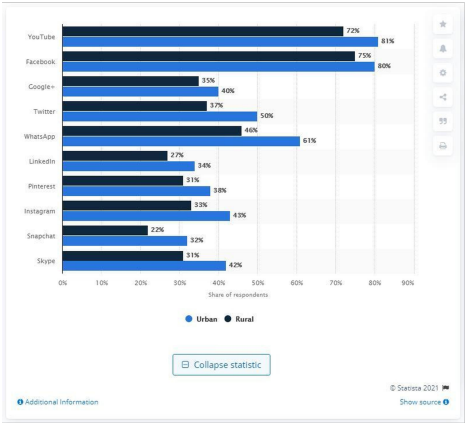
CPPO

<think>
I have six objects: a yellow sphere, a brown cube, a red sphere, a gray cylinder, a yellow cube, and a gray cube. After removing the brown cube and gray cylinder, I am left with four objects.
</think>
\boxed{4}

GRPO

<think>
To answer the question, I'll follow these steps:
1. Identify the initial number of objects: 5 (yellow spherical object, brown cube, gray cube, gray cylinder, yellow cube)
2. Remove the brown cube: This leaves us with 4 objects
3. Remove the gray cylinder: This leaves us with 3 objects
Therefore, after removing both the brown cube and the gray cylinder, there are 3 objects left.
Final answer: 3
</think>
\boxed{3}

Figure 8. Sample generated responses by CPPO and GRPO. GRPO exhibits a perception error that is corrected in the CPPO response.



Hint: Please answer the question requiring an integer answer and provide the final value, e.g., 1, 2, 3, at the end.
Question: What is the difference between highest and lowest value of dark blue bar?

CPPO

<think>
To find the difference between the highest and lowest values of the dark blue bar, I need to identify these values from the chart.
From the chart:
- The highest dark blue bar represents Facebook, with a value of 75%.
- The lowest dark blue bar represents Snapchat, with a value of 22%.
The difference is calculated as:
75% - 22% = 53%
Therefore, the difference between the highest and lowest value of the dark blue bar is 53%.
</think>
\boxed{53}

GRPO

<think>
To determine the difference between the highest and lowest value of the dark blue bars:
1. Identify the highest value: YouTube has the highest dark blue value, which is 72.
2. Identify the lowest value: Snapchat has the lowest dark blue value, which is 22.
3. Calculate the difference: 72 - 22 = 50.
Therefore, the difference between the highest and lowest value of the dark blue bars is 50.
</think>
\boxed{50}

Figure 9. Sample generated responses by CPPO and GRPO. GRPO exhibits a perception error that is corrected in the CPPO response.