

Faithful GRPO: Improving Visual Spatial Reasoning in Multimodal Language Models via Constrained Policy Optimization

Sai Srinivas Kancheti^{1*,†}, Aditya Sanjiv Kanade^{2,*}, Rohit Sinha¹,
Vineeth N. Balasubramanian^{1,2} & Tanuja Ganu²

¹Indian Institute of Technology Hyderabad, India ²Microsoft Research India
cs21resch01004@iith.ac.in, kanade850@gmail.com, rohit.sinha@prjt.cse.iith.ac.in
{vineeth.nb, tanuja.ganu}@microsoft.com

Abstract

Multimodal reasoning models (MRMs) trained with reinforcement learning with verifiable rewards (RLVR) show improved accuracy on visual reasoning benchmarks. However, we observe that accuracy gains often come at the cost of reasoning quality: generated Chain-of-Thought (CoT) traces are frequently inconsistent with the final answer and poorly grounded in the visual evidence. We systematically study this phenomenon across seven challenging real-world spatial reasoning benchmarks and find that it affects contemporary MRMs such as ViGoRL-Spatial, TreeVGR as well as our own models trained with standard Group Relative Policy Optimization (GRPO). We characterize CoT reasoning quality along two complementary axes: *logical consistency* (does the CoT entail the final answer?) and *visual grounding* (does each reasoning step accurately describe objects, attributes, and spatial relationships in the image?). To address these failure modes, we propose Faithful GRPO (FGRPO), a variant of GRPO that treats consistency and grounding as constraints enforced via Lagrangian dual ascent. FGRPO incorporates batch-level consistency and grounding constraints into the advantage computation within a group, adaptively adjusting the relative importance of constraints during optimization. We evaluate FGRPO on Qwen2.5-VL-7B and 3B backbones across seven spatial reasoning datasets. FGRPO substantially improves reasoning quality, reducing the inconsistency rate from 26.1% to 1.7% and improving visual grounding scores by +13%. It also improves final answer accuracy over standard GRPO, demonstrating that faithful reasoning enables better answers.

1 Introduction

Multimodal Large Language Models (Bai et al., 2025; Zhu et al., 2025; Li et al., 2024) (MLMs) have achieved strong performance on visual understanding tasks (Yue et al., 2024; Liu et al., 2023; Lu et al., 2023), substantially enhanced by Chain-of-Thought (CoT) reasoning (Wei et al., 2022; Wang et al., 2022). Reinforcement Learning with Verifiable Rewards (RLVR) (Lambert et al., 2024; Trung et al., 2024), as demonstrated by DeepSeek-R1 (Guo et al., 2025), has emerged as the dominant paradigm for training reasoning capabilities via a two-stage recipe of supervised finetuning (SFT) followed by Group Relative Policy Optimization (GRPO) (Shao et al., 2024b). Following this paradigm, we train a task-reward model (GRPO-T) on diverse spatial reasoning data (§3.1), which shows accuracy gains over the Qwen2.5-VL backbone (65.17% vs Qwen2.5-VL-7B backbone 64.17% from Table 1).

However, accuracy alone is an incomplete measure of reasoning ability. As shown in Figure 1, we observe pervasive reasoning failures in the traces generated by GRPO-T. A model that generates the correct answer despite flawed, contradictory, or visually ungrounded reasoning may have learned to exploit shortcuts (Yuan et al., 2024) and biases to answer

*Equal contribution.

†Work done while at Microsoft Research India.

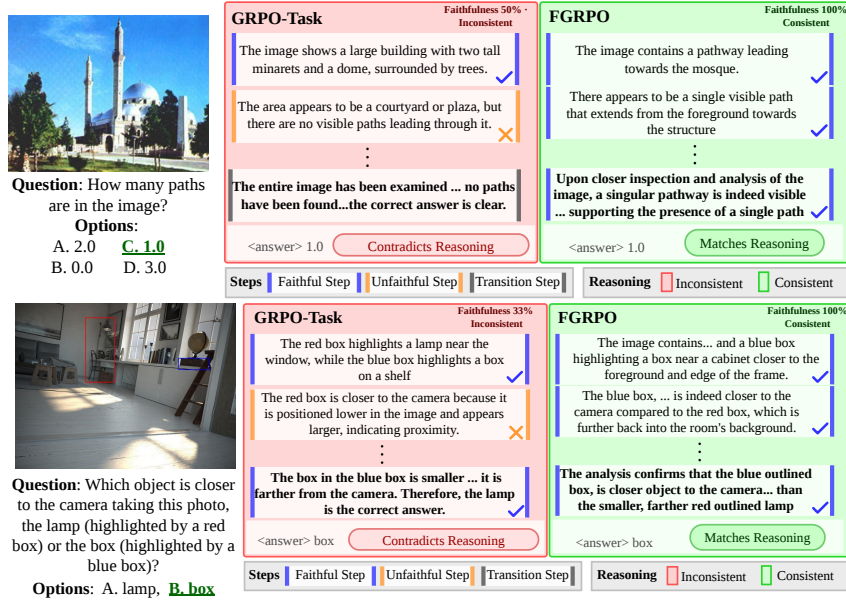


Figure 1: **Unfaithful reasoning masked by correct answers.** Both models answer correctly, but only FGRPO reasons faithfully. **Left:** The GRPO-Task model incorrectly claims there are no visible paths, contradicting its own answer of “1.0” (50% faithfulness, inconsistent). **Right:** The GRPO-Task model reasons toward “lamp” but answers “box” (33% faithfulness, inconsistent). FGRPO produces visually grounded reasoning in both cases (100% faithfulness, consistent). Sentences are color-coded: blue are grounded and orange are ungrounded.

the question, making it untrustworthy for real-world use. We identify two distinct failure modes in CoT traces generated by trained MRMs: (i) **Logical Inconsistency**, where the reasoning trace argues toward one conclusion, but the model then abruptly flips its prediction to a different final answer (Figure 1, right: the GRPO-T model reasons toward ‘lamp’ but answers ‘box’); and (ii) **Visual Ungroundedness**, where individual reasoning steps describe objects, attributes or spatial relationships that are inaccurate with respect to the visual content (Figure 1, left: the GRPO-T model claims there are no visible paths). A trace can be logically consistent yet visually ungrounded, or well-grounded yet inconsistent with its own answer. We provide additional contrastive examples spanning diverse spatial tasks in Appendix F.

We address these failure modes by defining verifiable reward signals for both consistency and grounding (§3.2). Consistency is treated as a binary reward via an LLM judge. For visual grounding, we combine a per-sentence semantic grounding reward scored by a VLM judge with a spatial grounding reward based on IoU matching of predicted bounding boxes to ground-truth regions. Each reward directly targets a specific failure mode: the consistency reward addresses logical inconsistency, while the grounding rewards address visual ungroundedness.

We then propose **Faithful GRPO (FGRPO)**, a constrained variant of GRPO that formulates consistency and visual grounding as constraints in a CMDP and enforces them approximately through adaptive dual ascent (Altman, 1999; Tessler et al., 2018). In contrast to having fixed reward weights, the Lagrange multipliers in FGRPO increase for violated constraints and decrease for satisfied constraints. Thus, the method automatically learns a curriculum on reward weights, giving a larger weightage to whichever aspect of reasoning quality is currently weak, removing the need for manual reward weight tuning. FGRPO treats consistency and grounding as *prerequisites* for trustworthy visual reasoning, making them natural candidates for constraints that need to be satisfied.

We evaluate FGRPO on Qwen2.5-VL-7B and 3B backbones across seven spatial reasoning datasets. On the 7B backbone, FGRPO improves average accuracy from 65.17% to 67.16%

over standard GRPO, while reducing the inconsistency rate from 26.1% to 1.7% and improving semantic grounding from 72.7% to 86.0%. FGRPO thus substantially improves reasoning faithfulness while also improving final-answer accuracy over standard GRPO. The same trend holds at the 3B scale, showing that the improvements are not specific to a single model size.

We summarize our contributions below:

- We characterize reasoning quality degradation in RLVR-trained MRMs along two axes: logical consistency and visual grounding. We observe this degradation in contemporary MRMs and in our own models trained with standard GRPO.
- We define verifiable reward signals for trustworthy visual reasoning, and propose Faithful GRPO (FGRPO), which treats consistency and grounding rewards as constraints to be satisfied for visual reasoning.
- We evaluate FGRPO on two backbones across seven spatial datasets and show that it improves both accuracy and reasoning quality, demonstrating that faithful reasoning and accurate answers are complementary objectives.

2 Related Work

RLVR for Multimodal Reasoning. Reinforcement Learning with Verifiable Rewards (RLVR) (Lambert et al., 2024; Trung et al., 2024) optimizes models on tasks with automatically checkable outcomes, sidestepping the need for learned reward models. DeepSeek-R1 (Guo et al., 2025) demonstrates that a two-stage pipeline – SFT on CoT traces followed by GRPO (Shao et al., 2024b) with verifiable rewards yields strong reasoning performance across domains (Wen et al., 2025). A growing body of work extends this paradigm to multimodal reasoning. Vision-R1 (Huang et al., 2025) constructs large-scale CoT SFT data and introduces progressive thinking suppression during RL. VL-Rethinker (Wang et al., 2025b) encourages self-reflection via forced rethinking triggers and selective sample replay. R1-OneVision (Yang et al., 2025) converts visual inputs into structured textual representations before applying language-only reasoning. TreeVGR (Wang et al., 2025a) and ViGoRL (Sarch et al., 2025) focus on visually grounded reasoning: TreeVGR supervises both localization and reasoning with dual IoU-based rewards, while ViGoRL grounds every reasoning step with image coordinates via MCTS-generated point-grounded CoT traces. Our work builds on the same two-stage paradigm but departs from prior MRMs in two ways: (i) we define verifiable reward signals for consistency and visual grounding that have not been used as trainable rewards in prior work, and (ii) we formulate these signals as constraints in a CMDP (Altman, 1999) and enforce them approximately through adaptive dual ascent rather than as fixed reward terms, addressing the accuracy–faithfulness tradeoff that affects existing MRMs.

Constrained Policy Optimization. Constrained Markov Decision Processes (CMDPs) (Altman, 1999) provide a principled framework for optimizing a primary objective subject to auxiliary constraints. Lagrangian relaxation is a standard tool for solving CMDPs, employed in methods such as CPO (Achiam et al., 2017), RCPO (Tessler et al., 2018), PID Lagrangian methods (Stooke et al., 2020), and first-order constrained optimization in policy space (Zhang et al., 2020). These approaches have been applied to safe exploration (Achiam & Amodei, 2019), LLM alignment (Dai et al., 2023), and recently to safety constraints for vision-language-action models (Zhang et al., 2025). Concurrent to our work, MO-GRPO (Ichihara et al., 2025) identifies that standard GRPO suffers from reward hacking under multiple objectives because within-group normalization allows high-variance rewards to dominate, and proposes scalarized per-objective advantages to equalize their influence. GDPO (Liu et al., 2026) similarly advocates decoupled normalization and utilizes conditional objectives that activate reward terms only once prerequisite scores exceed a minimum threshold. FGRPO shares the decoupled normalization strategy with these methods but differs in its use of Lagrangian dual ascent to enforce specified constraint thresholds rather than fixed or conditional weights, and in its application to enforcing reasoning quality (consistency and visual grounding) in multimodal RL.

3 Methodology

We begin by describing the standard GRPO training setup for multimodal reasoning in § 3.1. We then define two complementary axes along which reasoning quality can be measured (§3.2) and describe our reward formulation in detail. In § 3.3, we present Faithful GRPO, which treats consistency and visual grounding as constraints enforced via Lagrangian dual ascent. An overview of the FGRPO training pipeline is shown in Figure 2.

3.1 Preliminaries

Backbone and Two-Stage Training. We build on the Qwen2.5-VL-7B-Instruct backbone (Bai et al., 2025) and follow the two-stage training paradigm of DeepSeek-R1 (Guo et al., 2025). In stage one, we perform SFT on curated CoT data to imbue the backbone with baseline spatial reasoning capability. We create CoT data from a strong visual teacher (Qwen2.5-VL-72B-Instruct) using Monte Carlo Tree Search (MCTS) (Browne et al., 2012; DeLorenzo et al., 2024), which generates diverse, high-quality trajectories including synthetic backtracking for self-correction (Appendix A.4). We curate approximately 45K CoT traces from three seed datasets: SAT (Ray et al., 2025), VGR (Wang et al., 2025c), and VisCoT (Shao et al., 2024a) that collectively span diverse real-world spatial questions and image sources including COCO, GQA, OpenImages, and Flickr30k. In stage two, the SFT checkpoint is finetuned with reinforcement learning using GRPO (Shao et al., 2024b) on a curated RL dataset of 49K samples. We employ difficulty-based filtering (Xia et al., 2024; Yu et al., 2025) to select samples of intermediate difficulty and include 13K samples from TreeVGR-RL-37K (Wang et al., 2025a) for bounding-box supervision diversity. We present the full training hyperparameters, data curation details, and the MCTS procedure in Appendix A.

GRPO. Let π_θ denote the policy, which autoregressively generates a response $o = (s_1, s_2, \dots, s_T, a)$ conditioned on a multimodal prompt x , where s_t denotes a reasoning step and a the final answer. Given a batch of N prompts, GRPO (Shao et al., 2024b) generates G rollouts $\{o_i^j\}_{i=1}^G$ for each prompt x_j and computes per-rollout scalar rewards $r_i^j = R(o_i^j)$. The advantage is obtained by normalizing within each group: $\hat{A}_i^j = (r_i^j - \mu_j) / (\sigma_j + \epsilon)$, where μ_j and σ_j are the mean and standard deviation of rewards within group j . The policy is updated via the clipped surrogate objective with a KL penalty (Shao et al., 2024b).

Task Reward. We use the standard task reward that combines format adherence and answer accuracy: $R_{\text{task}}(o) = 0.5 \cdot R_{\text{acc}}(o) + 0.5 \cdot R_{\text{fmt}}(o)$, where $R_{\text{fmt}}(o) = 1$ if o follows the `<think>...</think><answer>...</answer>` format and 0 otherwise, and $R_{\text{acc}}(o) = 1$ if the answer matches the ground truth and 0 otherwise. Since our RL dataset is formulated as MCQs, we perform exact matching after stripping punctuation.

3.2 Consistency and Grounding Rewards

Training with R_{task} improves answer accuracy, but as discussed in §1, this improvement frequently comes at the expense of reasoning quality. We now define verifiable reward signals that capture the two failure modes identified above – logical inconsistency and visual ungroundedness, enabling them to be used as training objectives.

Consistency Reward. We define the *consistency reward* $R_C(o) \in \{0, 1\}$ as a binary signal indicating whether the CoT reasoning trace logically entails the final answer. Given a response o with reasoning trace $\mathcal{T}$ (the text within `<think>` tags) and final answer $\mathcal{A}$ (the text within `<answer>` tags), we prompt a text-only LLM judge to determine whether $\mathcal{A}$ follows logically from $\mathcal{T}$: $R_C(o) = \text{LLM-Judge}(\mathcal{T}, \mathcal{A}) \in \{0, 1\}$, where the judge outputs 1 (consistent) if the reasoning’s conclusion matches the final answer, and 0 (inconsistent) otherwise. The judge evaluates only textual logical coherence, ignoring visual correctness. A trace that reasons incorrectly about the image but answers in accordance with its own reasoning is scored as consistent. We provide the consistency reward only to reasoning traces that lead to a correct answer, i.e. we mask R_C when $R_{\text{acc}}(o) = 0$. This prevents

the model from learning degenerate policies that produce reasoning traces that receive a high reward from the judge but are still incorrect. The full judge prompt is provided in Appendix D.

Visual Grounding Rewards. We define two complementary reward signals that capture whether the reasoning trace is anchored in the visual evidence.

Semantic grounding reward $R_S(o) \in [0, 1]$ measures whether individual reasoning steps accurately describe the objects, attributes, and spatial relationships visible in the image. We decompose the reasoning trace into sentences $\{s_1, \dots, s_K\}$, filter out trivial non-visual sentences (e.g., meta-reasoning, planning, hedging, arithmetic) using regex-based matching and remove near-duplicate visual claims via Jaccard overlap and token-set-ratio filtering (Bachmann, 2025). We score each remaining sentence via a VLM judge that receives the image, the question, and the reasoning context:

$$R_S(o) = \frac{1}{K'} \sum_{k=1}^{K'} \hat{s}_k, \quad \hat{s}_k = \text{VLM-Judge}(I, Q, s_{\leq k}) \in \{0, 1\}, \quad (1)$$

where K' is the number of non-trivial scored sentences and $\hat{s}_k = 1$ if the sentence is classified as CORRECT (the visual claim is accurate) and 0 if INCORRECT. Sentences classified as SKIP (no visual claim) are excluded. The VLM judge checks entity grounding, attribute verification, spatial relationship accuracy, and bounding-box content validity. As with consistency, we provide the semantic-grounding reward only to reasoning traces that lead to a correct answer, i.e. we mask R_S when $R_{\text{acc}}(o) = 0$. Details of the sentence decomposition, trivial sentence filtering, and the VLM judge prompt are provided in Appendix A.1.

Spatial grounding reward $R_G(o) \in [0, 1]$ measures whether bounding-box coordinates generated in the reasoning trace correspond to the correct image regions. For models that produce bounding boxes $\hat{B}(o) = \{\hat{b}_i\}_{i=1}^N$ within `<bbox>` tags, we compute $R_G(o)$ via Hungarian matching (Kuhn, 1955) against ground-truth boxes $B = \{b_j\}_{j=1}^M$ using Complete IoU (CIoU) (Zheng et al., 2019):

$$R_G(o) = \frac{1}{\max(N, M)} \sum_{i=1}^N \text{CIoU}(\hat{b}_i, b_{\sigma(i)}), \quad (2)$$

where σ denotes the optimal assignment. CIoU incorporates intersection-over-union, center distance, and aspect ratio penalties. This reward is available only for training samples with bounding-box annotations (VGR and TreeVGR datasets).

Empirical Motivation. We evaluate our task-reward trained model GRPO-T on seven spatial benchmarks using these reward signals as diagnostic metrics. Despite achieving 65.17% average accuracy, GRPO-T exhibits an inconsistency rate of 26.1%—over one in four samples has reasoning that contradicts the final prediction. Its mean semantic grounding score is 72.7%, indicating that over a quarter of visual claims in the reasoning traces are inaccurate with respect to the image. We observe similar degradation across five contemporary MRM baselines (Table 1, Figure 3). Naively adding R_C and R_G as reward terms does not resolve this: additive shaping reduces inconsistency but sacrifices accuracy, while multiplicative gating degrades both (Table 2). These findings motivate formulating consistency and grounding as constraints whose relative optimization pressure can adapt according to their degree of satisfaction, rather than incorporating them as fixed reward terms.

3.3 Faithful GRPO (FGRPO)

Rather than incorporating reasoning quality signals as reward terms where they compete with and are traded off against task accuracy, we propose treating them as *constraints* that must be satisfied during policy optimization. This formulation ensures that the model cannot sacrifice reasoning quality for accuracy gains.

Constrained Optimization Formulation. Let π_θ denote the policy and R_{task} denote the task reward. We formulate FGRPO as a constrained optimization problem over the two

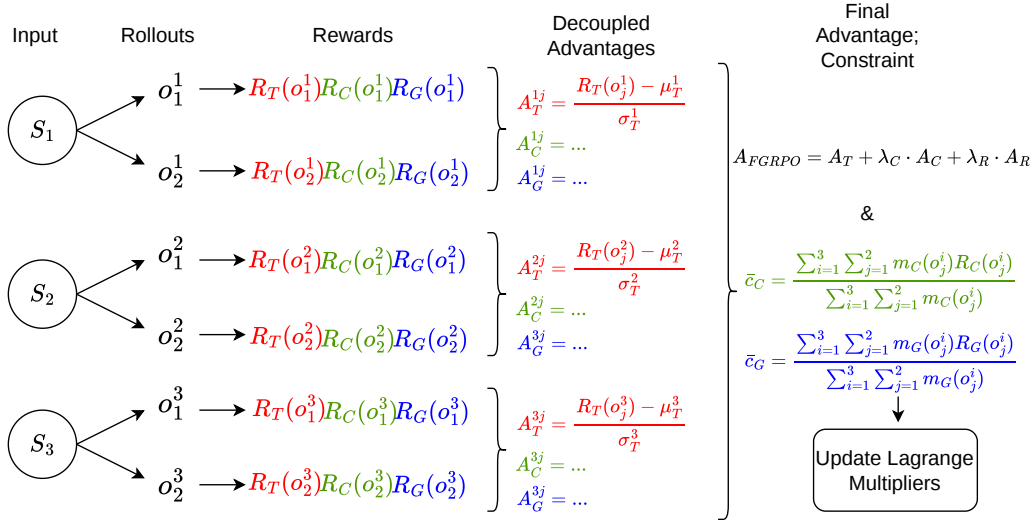


Figure 2: **Overview of the FGRPO training pipeline.** We show advantage computation for a training batch with 3 samples, each with 2 rollouts. For each prompt-image pair s_i , the policy samples $G = 2$ rollouts and we compute the task reward R_T , the consistency reward R_C , and grounding rewards R_S and R_G (only R_G is shown for clarity). The consistency and semantic-grounding rewards are provided by an online VLM judge. Each signal is independently normalized within the rollout group to obtain decoupled advantages, which are then combined into the final FGRPO advantage using Lagrange multipliers. Batch-level constraint statistics are used to update the multipliers via dual ascent, increasing the weight of violated constraints and decreasing the weight of satisfied ones.

reasoning quality axes: consistency and visual grounding.

$$\max_{\theta} \mathbb{E}_{x,o \sim \pi_{\theta}} [R_{\text{task}}(o)] \quad \text{s.t.} \quad \mathbb{E}[R_C(o)] \geq \tau_C, \quad \mathbb{E}[R_S(o)] \geq \tau_S, \quad \mathbb{E}[R_G(o)] \geq \tau_G, \quad (3)$$

where τ_C , τ_S , and τ_G are threshold hyperparameters. The consistency constraint R_C and semantic-grounding constraint R_S operates on all correct-answer samples (Appendix A.1). The spatial grounding constraint $R_G(o)$ is computed only for samples with bounding-box annotations.

Lagrangian Relaxation. Following the constrained MDP framework (Altman, 1999; Tessler et al., 2018), we convert the constrained problem into an unconstrained Lagrangian:

$$\mathcal{L}(\theta, \lambda) = \mathbb{E}[R_{\text{task}}(o)] + \lambda_C (\mathbb{E}[R_C(o)] - \tau_C) + \lambda_S (\mathbb{E}[R_S(o)] - \tau_S) + \lambda_G (\mathbb{E}[R_G(o)] - \tau_G), \quad (4)$$

where $\lambda = (\lambda_C, \lambda_S, \lambda_G) \geq 0$ are Lagrange multipliers. The policy parameters θ are updated to maximize $\mathcal{L}$, while the multipliers are updated via dual ascent:

$$\lambda_k \leftarrow \text{clip}\left(0, \lambda_{\max}, \lambda_k + \eta_{\lambda} \cdot (\tau_k - \bar{c}_k)\right), \quad k \in \{C, S, G\}, \quad (5)$$

where η_{λ} is the dual learning rate and $\lambda_{\max}$ is an upper bound for stability. The batch-average constraint score $\bar{c}_k$ is computed over all $N \times G$ rollouts using per-rollout masks $m_k(o_i^j) \in \{0, 1\}$:

$$\bar{c}_k = \frac{\sum_{j=1}^N \sum_{i=1}^G m_k(o_i^j) \cdot R_k(o_i^j)}{\sum_{j=1}^N \sum_{i=1}^G m_k(o_i^j)}, \quad (6)$$

where $m_C(o_i^j) = m_S(o_i^j) = \mathbb{1}[R_{\text{acc}}(o_i^j) > 0]$ and $m_G(o_i^j) = \mathbb{1}[\text{source}(x_j) \in \{\text{VGR}, \text{TreeVGR}\}]$. Masking R_C and R_S to correct predictions prevents reward hacking: without masking, the model could trivially satisfy consistency and grounding constraints by producing incorrect

but internally coherent and well-grounded answers, sacrificing accuracy to inflate constraint scores. Spatial grounding is masked by data availability since ground-truth bounding boxes are required. When $\bar{c}_k \geq \tau_k$, λ_k decreases; when $\bar{c}_k < \tau_k$, λ_k increases.

Practical Realization within GRPO. Training proceeds via alternating optimization. In the *primal step*, the policy π_θ is updated to maximize the Lagrangian (Eq. 4) using the clipped surrogate objective from GRPO. In the *dual step*, the multipliers are updated via Eq. 5 using the constraint scores from the current batch. Since GRPO operates on rollout-level rewards that are group-normalized into advantages, we must translate the Lagrangian into this advantage-based framework. The challenge is that the constraint signals are heterogeneous: R_C is binary, R_G is continuous and only defined for a subset of samples, and R_S is a mean over per-sentence scores, so naively summing them into a single reward before normalization would allow one signal’s scale to dominate. We address this via decoupled normalization.

Advantage Computation. A straightforward approach would be to add constraint rewards directly to R_{task} as weighted terms before group normalization. However, GRPO’s within-group normalization can nullify such signals entirely: if a constraint score is constant across all rollouts for a prompt (e.g., all rollouts for x_j receive the same binary consistency score), it cancels in the mean subtraction and contributes zero gradient, regardless of its weight. To avoid this, following GDPO (Liu et al., 2026), we apply group-relative normalization *independently* to each signal before combining them. The final advantage for rollout o_i^j is:

$$\hat{A}_{\text{FGRPO}}(o_i^j) = \hat{A}_{\text{task}}(o_i^j) + \sum_{k \in \{C, S, G\}} \lambda_k \cdot \hat{A}_k(o_i^j), \quad (7)$$

where $\hat{A}_{\text{task}}(o_i^j)$ and $\hat{A}_k(o_i^j)$ are independently group-normalized advantages for the task reward and constraint k within group j , as shown in Figure 2. This decoupled normalization ensures each signal operates on a comparable scale and contributes meaningful gradient even when its within-group variance is low. The combined advantage is whitened and used in the standard clipped surrogate loss. Note that decoupled normalization with *fixed* multipliers already resolves signal cancellation, but converges to a single point on the Pareto frontier dictated by the weight vector, with no mechanism to enforce the thresholds τ_k . Adaptive dual ascent (Eq. 5) addresses this by growing λ_k when a constraint is violated and shrinking it when satisfied, producing an emergent curriculum that automatically prioritizes whichever constraint is most violated without manual weight tuning (Table 3).

4 Experiments

We evaluate FGRPO on two model scales across seven spatial reasoning benchmarks, comparing against unconstrained GRPO baselines and reward design alternatives. We then ablate the contribution of individual constraints, their composition, and the role of adaptive Lagrange multipliers.

4.1 Experimental Setup

Models. We train two backbone sizes: Qwen2.5-VL-3B-Instruct and Qwen2.5-VL-7B-Instruct (Bai et al., 2025), both following the two-stage pipeline. We curate bounding-box grounded CoT data, where each reasoning step references regions of interest via `<bbox>[x1,y1,x2,y2]</bbox>` tags before reasoning over salient objects. The SFT checkpoint trained on this bbox CoT data serves as the common initialization for all RL models.

Training Data. The RL training set comprises approximately 36K samples drawn from SAT (Ray et al., 2025), VGR (Wang et al., 2025c), and VisCoT (Shao et al., 2024a), supplemented with 13K samples from TreeVGR-RL-37K (Wang et al., 2025a) for bounding-box supervision diversity. Samples are filtered by intermediate difficulty following prior work (Xia et al., 2024; Yu et al., 2025). Ground-truth bounding-box annotations are available only for

VGR and TreeVGR-RL-37K; the spatial grounding constraint is masked accordingly (§3.3). During evaluation there is no bounding-box supervision of any kind.

Evaluation. We evaluate on seven spatial reasoning benchmarks: CVBench-2D (Tong et al., 2024a), CVBench-3D (Tong et al., 2024a), MindCube (Yin et al., 2025), MMVP (Tong et al., 2024b), OmniSpatial (Jia et al., 2025), RealWorldQA (xAI, 2024), and SAT-Real (Ray et al., 2025), spanning both in-distribution and out-of-distribution settings. We report three metrics: (i) **Accuracy** (pass@1 with greedy decoding), (ii) **Inconsistency Rate** ($IR = N_{\text{inconsistent}} / N_{\text{total}}$), the fraction of all samples whose reasoning trace is inconsistent with the final answer (lower is better), and (iii) **Semantic Grounding** $R_S(o)$, the mean per-sentence visual grounding score using a VLM judge (§3.2).

For answer extraction during accuracy computation, we use a lightweight Qwen3-4B-Instruct-2507 model. We validate this accuracy-scoring judge against GPT-5, obtaining a Cohen’s κ (Cohen, 1960) of 0.997, indicating almost perfect agreement (Appendix B.3). During training, consistency and semantic-grounding rewards are computed using the open-weight Qwen3-VL-30B-A3B-Instruct judge, while evaluation is performed with GPT-5.4 (OpenAI, 2025). GPT-5.4 is not used to provide training rewards. Thus, the reported evaluation metrics are not computed by the same model that supplies the training reward. We further test robustness to the choice of training judge using Qwen3.6-35B-A3B and Qwen3-VL-32B-Instruct, and observe that performance remains on-par across sufficiently capable judges. We also evaluate semantic grounding using GPT-5.5: GPT-5.4/GPT-5.5 scores are 72.7/71.91% for GRPO-T and 86.0/85.33% for FGRPO, with the two judges returning the same CORRECT/INCORRECT verdict on 88.1% of judged sentences and a mean absolute error of 0.1. Full judge-validation results are provided in Appendix B.3.

Implementation Details. All RL runs use AdamW with learning rate 1×10^{-6} , bf16 precision, $8 \times H100$ GPUs, $G = 5$ rollouts per prompt, and a KL coefficient of 0.001. For FGRPO, the Lagrange dual learning rate is $\eta_\lambda = 0.05$, the multiplier upper bound is $\lambda_{\max} = 5.0$, and constraint updates require a minimum of 8 applicable samples per batch. Unless otherwise noted, constraint thresholds are $\tau_C = 0.95$, $\tau_G = 0.7$, $\tau_S = 0.95$, and initial multipliers are $\lambda_C^{(0)} = \lambda_G^{(0)} = \lambda_S^{(0)} = 1.0$. We study sensitivity to the constraint thresholds in Appendix C.1 where we find that infeasibly high thresholds lead to quick saturation of the Lagrange multipliers and reduce the effectiveness of the task reward, while easily achievable lower thresholds show unstable training dynamics and fail to converge. During training we do not use proprietary API calls. The Qwen3-VL-30B-A3B-Instruct judge is open-weight and served locally using vLLM (Kwon et al., 2023). Training takes approximately 18 hours for GRPO-T and 24 hours for FGRPO, with the majority of the overhead occurring during rollout generation. Inference time is the same for FGRPO and GRPO-T. We provide further implementation details in Appendix A.

4.2 Main Results

Table 1 compares FGRPO against the Qwen2.5-VL backbone, five contemporary MRMs, and GRPO-T (task reward only). We make three observations: First, the two-stage pipeline with diverse data curation yields a strong baseline: GRPO-T alone outperforms most existing MRM baselines, including TreeVGR and ViGoRL-Spatial, which are explicitly trained for visual spatial reasoning. Second, FGRPO further improves upon this strong baseline, lifting average accuracy from 65.17 to 67.16 with consistent gains across the majority of benchmarks. Third, the same pattern holds at the 3B scale (Table 1, bottom), confirming that FGRPO’s gains are not specific to the 7B backbone.

4.3 Reasoning Quality

Inconsistency & Semantic Grounding. Figure 3 breaks down reasoning quality by dataset. FGRPO achieves higher semantic grounding than GRPO-T across all seven benchmarks (86.0% vs. 72.7% overall), an improvement of +13.3%, with the largest gains on MindCube (+22.8%) and OmniSpatial (+21.1%). These datasets require multi-step spatial reasoning where unconstrained models produce the most ungrounded sentences. TreeVGR also

Table 1: **Main results.** Pass@1 accuracy on seven spatial reasoning benchmarks. All 7B models use Qwen2.5-VL-7B-Instruct as backbone; 3B models use Qwen2.5-VL-3B-Instruct. **Bold** = best, underline = second best among open-weight 7B models. FGRPO achieves the highest average accuracy at both scales, outperforming all MRM baselines and the unconstrained GRPO-T baseline while simultaneously improving reasoning quality.

Method	CVB 2D	CVB 3D	Mind Cube	MMVP	Omni Spatial	Real WorldQA	SAT Real	Avg.
<i>Base model (7B): Qwen2.5-VL-7B-Instruct</i>								
Non-reasoning	77.17	83.78	35.11	75.78	45.23	69.02	63.11	64.17
CoT prompting	75.92	76.09	30.83	72.44	40.40	63.05	59.22	59.71
<i>Base model (3B): Qwen2.5-VL-3B-Instruct</i>								
Non-reasoning	70.58	74.00	43.71	64.67	45.92	65.88	59.00	60.54
CoT prompting	71.21	68.00	40.38	63.67	40.77	62.35	55.00	57.34
<i>MRM baselines</i>								
R1-OneVision (Yang et al., 2025)	53.31	58.00	27.09	56.16	31.54	49.87	51.50	46.78
Vision-R1 (Huang et al., 2025)	71.58	75.83	36.95	72.22	39.75	67.41	58.45	60.31
ViGoRL-Spatial (Sarch et al., 2025)	76.59	86.14	39.36	73.22	36.97	65.67	58.44	62.34
TreeVGR (Wang et al., 2025a)	74.69	73.92	<u>43.14</u>	71.00	45.99	66.80	61.00	62.36
VL-Rethinker (Wang et al., 2025b)	76.06	80.75	37.81	75.89	39.84	68.50	<u>65.00</u>	<u>63.41</u>
<i>Proprietary models</i>								
GPT-5-nano (CoT)	76.29	86.75	27.71	75.67	41.03	71.90	64.00	63.34
GPT-4o (CoT)	78.23	86.42	43.52	84.33	45.73	73.59	68.67	68.64
<i>Ours (7B)</i>								
GRPO-T	<u>79.97</u>	85.92	41.71	<u>74.00</u>	40.90	66.67	67.00	65.17
FGRPO	82.38	87.04	49.28	73.33	<u>44.78</u>	<u>67.64</u>	65.66	67.16
<i>Ours (3B)</i>								
GRPO-T	77.24	80.44	46.06	65.22	39.88	61.83	58.67	61.33
FGRPO	77.95	82.5	49.1	64.3	43.6	60.7	58.6	62.39

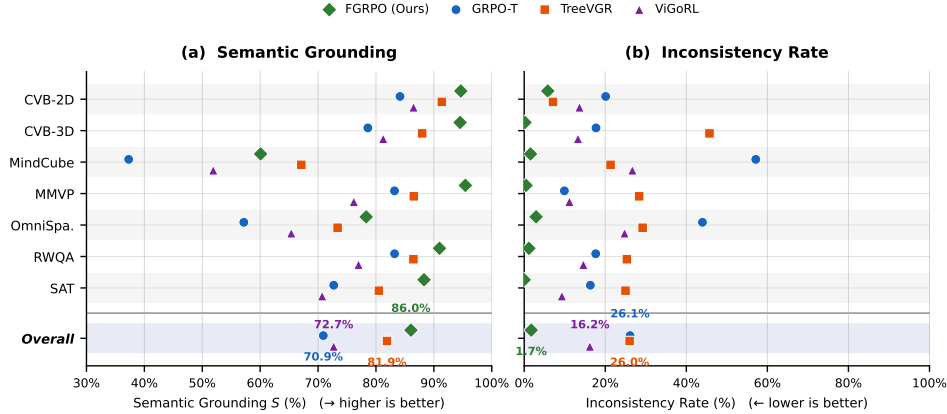


Figure 3: **Per-dataset reasoning quality breakdown.** (a) Semantic grounding (S): FGRPO achieves uniformly higher semantic grounding than GRPO-T across all seven benchmarks (86.0% vs. 72.7%), with the largest gains on MindCube (+22.8%) and OmniSpatial (+21.1%). TreeVGR also outperforms GRPO-T (81.9%). (b) Inconsistency rate: FGRPO reduces inconsistency to 1.7% on average, compared to 26.1% (GRPO-T), 26.0% (TreeVGR), and 16.2% (ViGoRL). MindCube and OmniSpatial are the most challenging for all baselines; FGRPO virtually eliminates inconsistency on every benchmark.

outperforms GRPO-T on semantic grounding (81.9%), but still trails FGRPO by 4.1%. FGRPO reduces inconsistency to 1.7% on average, compared to 26.1% (GRPO-T), 26.0% (TreeVGR), and 16.2% (ViGoRL). Inconsistency is most acute on MindCube (57.1% for GRPO-T) and OmniSpatial (44.0%), but FGRPO virtually eliminates it, achieving near-zero inconsistency on six of seven benchmarks.

4.4 Accuracy vs Consistency Tradeoff in RLVR

Figure 4 traces the accuracy–inconsistency trajectory across training iterations for GRPO-T and FGRPO, starting from an identical warmstart checkpoint. For GRPO-T, accuracy climbs from 57.7% to 65.17% over 1000 steps, but inconsistency more than triples (7.4% $\rightarrow$ 26.1%), producing increasingly unfaithful reasoning chains even as answer correctness improves. FGRPO, on the other hand, keeps inconsistency near or below the starting rate throughout training while pushing accuracy to 67.16%, converging at 1.7% inconsistency.

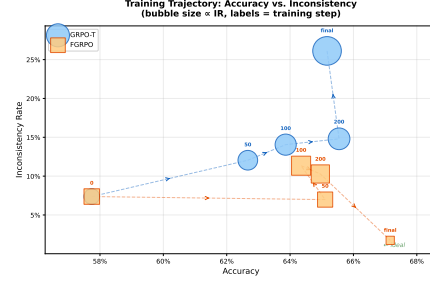


Figure 4: FGRPO (squares) vs GRPO-T (circles); larger shape size indicates higher inconsistency.

4.5 Reward Design and Multiplier Strategy

Coupled vs. decoupled advantage. The top block of Table 2 shows three GRPO variants that incorporate R_C and R_G as additional reward terms, combined into a single scalar before group normalization. Adding consistency and grounding additively ($R = \frac{1}{3}R_{acc} + \frac{1}{3}(R_{acc} \cdot R_C) + \frac{1}{3}(R_{fmt} \cdot R_G)$) reduces inconsistency to 4.2% but sacrifices accuracy. Multiplicative gating ($R = 0.5 \cdot R_{acc} \cdot R_C + 0.5 \cdot R_{fmt} \cdot R_G$) fares worse: accuracy drops by 1.70 points while inconsistency remains at 19.6%. The bottom block uses FGRPO’s decoupled advantage formulation (Eq. 7), where each signal is normalized independently before the advantages are combined. Even with only the consistency constraint, FGRPO improves accuracy while reducing inconsistency. Adding the grounding constraints (R_S and R_G) further improves accuracy to 67.16% while explicitly optimizing visual faithfulness.

Table 2: First three rows are GRPO.

Method	Acc (%) $\uparrow$	IR (%) $\downarrow$
<i>GRPO (coupled advantage)</i>		
Task only (GRPO-T)	65.17	26.12
+ Additive R_C, R_G	64.97	4.22
+ Multiplicative R_C, R_G	63.47	19.61
<i>FGRPO (decoupled advantage)</i>		
Consistency only (R_C)	66.16	0.54
Full ($R_C + R_S + R_G$)	67.16	1.73

Adaptive vs. fixed multipliers. Table 3 compares adaptive dual ascent against fixed multipliers ($\lambda_k = 1.0$, no updates). Fixed multipliers already yield strong consistency (IR 1.11%) and a +1.2 accuracy gain, confirming that the decoupled advantage formulation is the primary driver. Adaptive dual ascent (Eq. 5) further improves accuracy to 67.16 by reallocating optimization pressure as constraints are progressively satisfied.

Table 3: Adaptive λ s yield best accuracy

Method	Acc (%) $\uparrow$	IR (%) $\downarrow$
GRPO-T	65.17	26.12
FGRPO (fixed λ)	66.32	1.11
FGRPO (adaptive λ)	67.16	1.73

We provide additional analyses in Appendix C, including the role of accuracy masking, sensitivity to constraint thresholds, and robustness to the choice of training judge.

5 Conclusion

We presented Faithful GRPO (FGRPO), a constrained variant of GRPO that formulates logical consistency and visual grounding as constraints in a CMDP and enforces them through adaptive Lagrangian optimization. By defining verifiable reward signals for consistency (via an LLM judge) and grounding (via per-sentence VLM scoring and CIoU-based spatial matching), and combining decoupled advantage normalization with adaptive Lagrange multipliers, FGRPO ensures that accuracy gains do not come at the expense of reasoning quality. On seven spatial reasoning benchmarks, FGRPO reduces the inconsistency rate from 26.1% to 1.7%, improves semantic grounding from 72.7% to 86.0% (+13.3%), and simultaneously improves average answer accuracy from 65.17% to 67.16% (+2%) over standard GRPO. Our results thus show that reasoning faithfulness and accuracy can be complementary objectives. We hope that FGRPO encourages the community to move beyond accuracy-only evaluation and to treat reasoning quality as a first-class objective in multimodal RL training.

Acknowledgments

Sai Srinivas Kancheti would like to thank Microsoft Research India for the MSR India PhD Award and grant, under which this work was carried out. We thank the anonymous reviewers for their valuable feedback that improved the presentation of this paper.

References

- Josh Achiam and Dario Amodei. Benchmarking safe exploration in deep reinforcement learning. 2019. URL <https://api.semanticscholar.org/CorpusID:208283920>.
- Joshua Achiam, David Held, Aviv Tamar, and P. Abbeel. Constrained policy optimization. *ArXiv*, abs/1705.10528, 2017. URL <https://api.semanticscholar.org/CorpusID:10647707>.
- Eitan Altman. Constrained markov decision processes. 1999. URL <https://api.semanticscholar.org/CorpusID:14906227>.
- Max Bachmann. rapidfuzz/rapidfuzz: Release 3.13.0, April 2025. URL <https://doi.org/10.5281/zenodo.15133267>.
- Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *ArXiv*, abs/2502.13923, 2025. URL <https://api.semanticscholar.org/CorpusID:276449796>.
- Cameron Browne, Edward Jack Powley, Daniel Whitehouse, Simon M. M. Lucas, Peter I. Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez Liebana, Spyridon Samothrakis, and Simon Colton. A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in Games*, 4:1–43, 2012. URL <https://api.semanticscholar.org/CorpusID:9316331>.
- Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20:37 – 46, 1960. URL <https://api.semanticscholar.org/CorpusID:15926286>.
- Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. Safe rlhf: Safe reinforcement learning from human feedback. *ArXiv*, abs/2310.12773, 2023. URL <https://api.semanticscholar.org/CorpusID:264306078>.
- Matthew DeLorenzo, Animesh Basak Chowdhury, Vasudev Gohil, Shailja Thakur, Ramesh Karri, Siddharth Garg, and Jeyavijayan Rajendran. Make every move count: Llm-based high-quality rtl code generation using mcts. *ArXiv*, abs/2402.03289, 2024. URL <https://api.semanticscholar.org/CorpusID:267411899>.
- Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, 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.
- Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL <http://dx.doi.org/10.1038/s41586-025-09422-z>.
- Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaoshen Cao, Zheyu Ye, Fei Zhao, Zhe Xu, Yao Hu, and Shaohui Lin. Vision-r1: Incentivizing reasoning capability in multimodal large language models. *ArXiv*, abs/2503.06749, 2025. URL <https://api.semanticscholar.org/CorpusID:276902576>.

- Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6700–6709, 2019.
- Yuki Ichihara, Yuu Jinnai, Tetsuro Morimura, Mitsuki Sakamoto, Ryota Mitsuhashi, and Eiji Uchibe. Mo-grpo: Mitigating reward hacking of group relative policy optimization on multi-objective problems. *ArXiv*, abs/2509.22047, 2025. URL <https://api.semanticscholar.org/CorpusID:281658606>.
- Mengdi Jia, Zekun Qi, Shaochen Zhang, Wenyao Zhang, Xinqiang Yu, Jiawei He, He Wang, and Li Yi. Omnispatial: Towards comprehensive spatial reasoning benchmark for vision language models. *arXiv preprint arXiv:2506.03135*, 2025.
- Harold W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics (NRL)*, 52, 1955. URL <https://api.semanticscholar.org/CorpusID:9426884>.
- Alina Kuznetsova, Hassan Rom, Neil Gordon Alldrin, Jasper R. R. Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Mallocci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4. *International Journal of Computer Vision*, 128:1956 – 1981, 2018. URL <https://api.semanticscholar.org/CorpusID:53296866>.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.
- Nathan Lambert, Jacob Daniel Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, et al. Tulu 3: Pushing frontiers in open language model post-training. *ArXiv*, abs/2411.15124, 2024. URL <https://api.semanticscholar.org/CorpusID:274192505>.
- Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *ArXiv*, abs/2408.03326, 2024. URL <https://api.semanticscholar.org/CorpusID:271719914>.
- Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015. URL <https://arxiv.org/abs/1405.0312>.
- Shih-Yang Liu, Xin Dong, Ximing Lu, Shizhe Diao, Peter Belcák, Mingjie Liu, Min-Hung Chen, Hongxu Yin, Yu-Chiang Frank Wang, Kwang-Ting Cheng, Yejin Choi, Jan Kautz, and Pavlo Molchanov. Gdpo: Group reward-decoupled normalization policy optimization for multi-reward rl optimization. *ArXiv*, abs/2601.05242, 2026. URL <https://api.semanticscholar.org/CorpusID:284543897>.
- Yuanzhan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. Mmbench: Is your multi-modal model an all-around player? In *European Conference on Computer Vision*, 2023. URL <https://api.semanticscholar.org/CorpusID:259837088>.
- Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2017. URL <https://api.semanticscholar.org/CorpusID:53592270>.
- Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chun yue Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations*, 2023. URL <https://api.semanticscholar.org/CorpusID:264491155>.
- OpenAI. GPT-5 system card. <https://openai.com/index/gpt-5-system-card/>, August 2025.

- Arijit Ray, Jiafei Duan, Ellis L Brown II, Reuben Tan, Dina Bashkurova, Rose Hendrix, Kiana Ehsani, Aniruddha Kembhavi, Bryan A. Plummer, Ranjay Krishna, Kuo-Hao Zeng, and Kate Saenko. SAT: Dynamic spatial aptitude training for multimodal language models. In *Second Conference on Language Modeling*, 2025. URL <https://openreview.net/forum?id=DW8U8ZWa1U>.
- Gabriel Sarch, Snigdha Saha, Naitik Khandelwal, Ayush Jain, Michael J. Tarr, Aviral Kumar, and Katerina Fragkiadaki. Grounded reinforcement learning for visual reasoning. *ArXiv*, abs/2505.23678, 2025. URL <https://api.semanticscholar.org/CorpusID:278996797>.
- Hao Shao, Shengju Qian, Han Xiao, Guanglu Song, Zhuofan Zong, Letian Wang, Yu Liu, and Hongsheng Li. Visual cot: Unleashing chain-of-thought reasoning in multi-modal language models, 2024a.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Jun-Mei Song, Mingchuan Zhang, Y. K. Li, Yu Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *ArXiv*, abs/2402.03300, 2024b. URL <https://api.semanticscholar.org/CorpusID:267412607>.
- Adam Stooke, Joshua Achiam, and P. Abbeel. Responsive safety in reinforcement learning by pid lagrangian methods. In *International Conference on Machine Learning*, 2020. URL <https://api.semanticscholar.org/CorpusID:220404280>.
- Chen Tessler, Daniel Jaymin Mankowitz, and Shie Mannor. Reward constrained policy optimization. *ArXiv*, abs/1805.11074, 2018. URL <https://api.semanticscholar.org/CorpusID:44095973>.
- Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, Austin Wang, Rob Fergus, Yann LeCun, and Saining Xie. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 87310–87356. Curran Associates, Inc., 2024a. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/9ee3a664ccfeabc0da16ac6f1f1cfe59-Paper-Conference.pdf.
- Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9568–9578, June 2024b.
- Luong Trung, Xinbo Zhang, Zhanming Jie, Peng Sun, Xiaoran Jin, and Hang Li. ReFT: Reasoning with reinforced fine-tuning. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7601–7614, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.410. URL <https://aclanthology.org/2024.acl-long.410/>.
- Haochen Wang, Xiangtai Li, Zilong Huang, Anran Wang, Jiacong Wang, Tao Zhang, Jiani Zheng, Sule Bai, Zijian Kang, Jiashi Feng, Zhuochen Wang, and Zhaoxiang Zhang. Traceable evidence enhanced visual grounded reasoning: Evaluation and methodology. *arXiv preprint arXiv:2507.07999*, 2025a.
- Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhui Chen. VI-rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. *Advances in Neural Information Processing Systems* 38, pp. 34852–34878, 2025b. doi: 10.52202/085713-1037.
- Jiacong Wang, Zijian Kang, Haochen Wang, Haiyong Jiang, Jiawen Li, Bohong Wu, Ya Wang, Jiao Ran, Xiao Liang, Chao Feng, and Jun Xiao. Vgr: Visual grounded reasoning, 2025c. URL <https://arxiv.org/abs/2506.11991>.

- Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed H. Chi, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *ArXiv*, abs/2203.11171, 2022. URL <https://api.semanticscholar.org/CorpusID:247595263>.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed H. Chi, F. Xia, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *ArXiv*, abs/2201.11903, 2022. URL <https://api.semanticscholar.org/CorpusID:246411621>.
- Xumeng Wen, Zihan Liu, Shun Zheng, Zhijian Xu, Shengyu Ye, Zhirong Wu, Xiao Liang, Yang Wang, Junjie Li, Ziming Miao, Jiang Bian, and Mao Yang. Reinforcement learning with verifiable rewards implicitly incentivizes correct reasoning in base llms. *ArXiv*, abs/2506.14245, 2025. URL <https://api.semanticscholar.org/CorpusID:279410727>.
- xAI. Grok-1.5 Vision Preview. <https://x.ai/news/grok-1.5v>, April 2024.
- Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. Less: Selecting influential data for targeted instruction tuning. *ArXiv*, abs/2402.04333, 2024. URL <https://api.semanticscholar.org/CorpusID:267522839>.
- Yi Yang, Xiaoxuan He, Hongkun Pan, Xiyan Jiang, Yan Deng, Xingtao Yang, Haoyu Lu, Dacheng Yin, Fengyun Rao, Minfeng Zhu, et al. R1-onevision: Advancing generalized multimodal reasoning through cross-modal formalization. *IEEE International Conference on Computer Vision*, pp. 2376–2385, 2025. doi: 10.1109/ICCV51701.2025.00229.
- Baiqiao Yin, Qineng Wang, Pingyue Zhang, Jianshu Zhang, Kangrui Wang, Zihan Wang, Jieyu Zhang, Keshigeyan Chandrasegaran, Han Liu, Ranjay Krishna, Saining Xie, Manling Li, Jiajun Wu, and Li Fei-Fei. Spatial mental modeling from limited views, 2025. URL <https://arxiv.org/abs/2506.21458>.
- Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014.
- Qiyang Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Honglin Yu, Weinan Dai, Yuxuan Song, Xiang Wei, Haodong Zhou, Jingjing Liu, Wei Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yong-Xu Wu, and Mingxuan Wang. Dapo: An open-source llm reinforcement learning system at scale. *ArXiv*, abs/2503.14476, 2025. URL <https://api.semanticscholar.org/CorpusID:277104124>.
- Yu Yuan, Lili Zhao, Kai Zhang, Guangting Zheng, and Qi Liu. Do llms overcome shortcut learning? an evaluation of shortcut challenges in large language models. In *Conference on Empirical Methods in Natural Language Processing*, 2024. URL <https://api.semanticscholar.org/CorpusID:273404079>.
- Xiang Yue, Tianyu Zheng, Yuansheng Ni, Yubo Wang, Kai Zhang, Shengbang Tong, Yuxuan Sun, Ming Yin, Botao Yu, Ge Zhang, Huan Sun, Yu Su, Wenhua Chen, and Graham Neubig. Mmmu-pro: A more robust multi-discipline multimodal understanding benchmark. In *Annual Meeting of the Association for Computational Linguistics*, 2024. URL <https://api.semanticscholar.org/CorpusID:272397682>.
- Borong Zhang, Yuhao Zhang, Jiaming Ji, Yingshan Lei, Josef Dai, Yuanpei Chen, and Yaodong Yang. Safevla: Towards safety alignment of vision-language-action model via safe reinforcement learning. *ArXiv*, abs/2503.03480, 2025. URL <https://api.semanticscholar.org/CorpusID:276782535>.
- Yiming Zhang, Quan Ho Vuong, and Keith W. Ross. First order constrained optimization in policy space. *arXiv: Learning*, 2020. URL <https://api.semanticscholar.org/CorpusID:225090131>.

- Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyao Luo, Zhangchi Feng, and Yongqiang Ma. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand, 2024. Association for Computational Linguistics. URL <http://arxiv.org/abs/2403.13372>.
- Yaowei Zheng, Juntong Lu, Shenzhi Wang, Zhangchi Feng, Dongdong Kuang, and Yuwen Xiong. Easyr1: An efficient, scalable, multi-modality rl training framework. <https://github.com/hiyouga/EasyR1>, 2025.
- Zhaohui Zheng, Ping Wang, Wei Liu, Jinze Li, Rongguang Ye, and Dongwei Ren. Distance-iou loss: Faster and better learning for bounding box regression. *ArXiv*, abs/1911.08287, 2019. URL <https://api.semanticscholar.org/CorpusID:208158250>.
- Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Yuchen Duan, Hao Tian, Weijie Su, Jie Shao, Zhangwei Gao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *ArXiv*, abs/2504.10479, 2025. URL <https://api.semanticscholar.org/CorpusID:277780955>.

Appendix: Faithful GRPO: Improving Visual Spatial Reasoning in Multimodal Language Models via Constrained Policy Optimization

A Training and Data Curation Details

In this appendix we provide comprehensive details on the training pipeline, data curation, evaluation setup, and reward computation summarized in the main paper.

A.1 Semantic Grounding Reward: Sentence Decomposition and VLM Judge

The semantic grounding reward $R_S(o)$ (Eq. 1) requires evaluating whether each reasoning step in the CoT trace accurately reflects the image content. We decompose the reasoning trace into individual sentences and score each via a VLM judge.

Sentence Decomposition. Given a response o with reasoning trace within `<think>` tags, we decompose the trace into individual sentences $\{s_1, s_2, \dots, s_K\}$. Each sentence is classified as either *visual* (makes a specific claim about image content) or *trivial* (e.g., meta-reasoning, planning, logical conclusions, hedging, or arithmetic). We use a simple regex-based filter to remove sentences that do not perform visual reasoning. For instance, trivial sentences include planning and meta-intent (e.g., “Looking at the image, I need to check the bottle.”), logical conclusions (e.g., “Therefore, the answer must be option B.”), arithmetic, and statements without a visual keyword or bounding-box reference. We additionally use RapidFuzz (Bachmann, 2025) for efficient detection of near-duplicate visual claims. We remove sentences with high Jaccard overlap between their token sets and sentences with a high token-set-ratio, which captures near-matching formulations that token overlap alone can miss. These filters reduce the number of online VLM judge calls required during training.

VLM-as-Judge Scoring. Each remaining sentence s_k is evaluated by a VLM judge that receives the image, the question, and the reasoning context up to the current sentence. The judge classifies the sentence into one of three categories:

- **CORRECT** ($\hat{s}_k = 1$): The sentence makes a specific visual claim that is accurate—objects are present, attributes match, spatial relationships are correct, and any referenced bounding boxes contain the described content.
- **INCORRECT** ($\hat{s}_k = 0$): The sentence makes a specific visual claim that is factually inaccurate with respect to the image.
- **SKIP**: The sentence makes no specific visual claim or only repeats a previous observation without adding new visual information. These sentences are excluded from the reward computation.

The full judge prompt is provided in Appendix D. Judge reliability is analyzed in Appendix B.3.

A.2 Two-Stage Training Pipeline

We follow the two-stage training paradigm of DeepSeek-R1 (Guo et al., 2025). In stage one, we perform supervised finetuning (SFT) on curated Chain-of-Thought (CoT) data to imbue the model with baseline spatial reasoning capability. The MLM policy $\pi_\theta(o|x)$ autoregressively generates text $o = (s_1, s_2, \dots, s_T, a)$ conditioned on a multimodal prompt x , where s_t denotes a reasoning step and a denotes the final answer. For each SFT training sample $(x, o = (o_1, o_2, \dots, o_T))$, the negative log-likelihood $L_{\text{sft}} = -\mathbb{E}_{(x,o) \sim D} \sum_{t=1}^T \log \pi_\theta(o_t|x, o_{<t})$ is minimized. We use LLaMA-Factory (Zheng et al., 2024) for SFT with 3 epochs, AdamW (Loshchilov & Hutter, 2017) optimizer (learning rate $1\text{e-}6$, weight decay 0.01), batch size 32, and cosine learning rate schedule with warmup ratio 0.03. We freeze the vision encoder, keeping the LLM backbone and the visual projector trainable. The best checkpoint is selected based on validation loss. SFT is performed on $4 \times$ NVIDIA A100 80GB GPUs using DeepSpeed ZeRO Stage 3 and takes approximately 12 hours per variant.

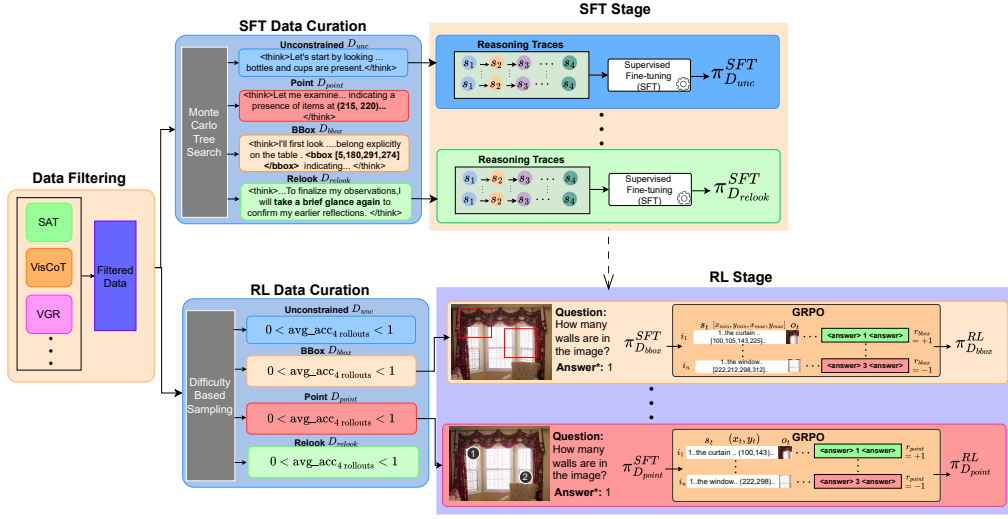


Figure 5: An overview of the two-stage training pipeline. We curate CoT training data using MCTS with a strong visual teacher, covering bounding-box grounded visual reasoning. We perform SFT on the curated CoT data using data from SAT (Ray et al., 2025), VGR (Wang et al., 2025c) and VisCoT (Shao et al., 2024a), followed by RL training. We add a subset of TreeVGR-RL-37K (Wang et al., 2025a) to construct diverse RL data with bounding-box supervision.

In stage two, the SFT checkpoint is finetuned using RLVR with Group Relative Policy Optimization (GRPO) (Shao et al., 2024b) to learn generalizable reasoning behaviors beyond the training distribution. We use a modified version of Easy-R1 (Zheng et al., 2025) for RL training. RL training uses GRPO with a group size of $G = 5$ rollouts per prompt, learning rate $1e-6$, and rollout batch size of 128. vLLM (Kwon et al., 2023) is used to generate rollouts efficiently. RL training is performed on $8 \times$ NVIDIA H100 GPUs. The full hyperparameter configurations for both stages are provided in Tables 4 and 5.

Table 4: SFT Hyperparameters

Hyperparameter	Value
Epochs	3
Learning rate	$1e-6$
Weight decay	0.01
Warmup ratio	0.03
Batch size (per device)	4
Gradient accumulation	2
Scheduler	Cosine
Precision	bf16
Flash attention	fa2
Freeze vision tower	True
Max sequence length	8192
Deepspeed config	ZeRO Stage 3

Table 5: GRPO Training Hyperparameters

Hyperparameter	Value
Total Epochs	3
Learning rate	$1e-6$
Weight decay	0.01
Warmup ratio	0.0
Optimizer	AdamW (bf16)
Group size (G)	5
KL coefficient	0.001
Clip ratio	0.28
Gradient clipping	1.0
Rollout batch size	128
Global batch size	64
Rollout engine	vLLM
Max prompt length	8192
Max response length	1024
Freeze vision tower	True
Precision	bf16

For FGRPO training, we additionally specify constraint-specific hyperparameters in Table 6.

Table 6: FGRPO Constraint Hyperparameters

Hyperparameter	Value
Advantage estimator	Decoupled (Eq. 7)
Constraint keys	[consistency, bbox_reward, faithfulness]
<i>Consistency constraint</i>	
Threshold (τ_C)	0.95
Initial λ_C	1.0
Masking	Only correct predictions ($R_{\text{acc}} > 0$)
<i>Spatial grounding constraint</i>	
Threshold (τ_G)	0.7
Initial λ_G	1.0
Masking	Only VGR/TreeVGR samples
<i>Semantic grounding constraint</i>	
Threshold (τ_S)	0.95
Initial λ_S	1.0
Reward level	Outcome (with per-sentence rewards)
Masking	Only correct predictions ($R_{\text{acc}} > 0$)
Dual learning rate (η_λ)	0.05
Max Lagrange multiplier (λ_{max})	5.0
Min applicable samples	8

A.3 Training Data Curation

We curate both SFT and RL datasets from three seed sources that span diverse real-world spatial questions and images.

Seed Datasets. We select SAT (Ray et al., 2025) (150K samples), VGR (Wang et al., 2025c) (90K samples), and VisCoT (Shao et al., 2024a) (363K samples) as seed datasets. These datasets span common image sources such as COCO (Lin et al., 2015), GQA (Hudson & Manning, 2019), OpenImages (Kuznetsova et al., 2018), and Flickr30k (Young et al., 2014), and cover a diverse range of spatial questions. We discard irrelevant data such as charts, tables, and visual math questions.

CoT Data Curation for SFT. We curate CoT data by distilling reasoning chains from a strong visual teacher (Qwen2.5-VL-72B-Instruct). For each seed domain, we cluster question embeddings and pick representative samples from each cluster: 1.5K from SAT, 1.5K from VGR, and 3K from VisCoT, for a total of 6K seed samples. For each sample, we run Monte Carlo Tree Search (MCTS) to generate reasoning traces (see §A.4). MCTS-based trajectories are not restricted to the base policy of the teacher, allowing us to generate approximately 45K high-quality diverse rollouts from the relatively small pool of 6K samples. We generate CoT data using the bounding-box grounded reasoning format, which produces reasoning chains with explicit <bbox> coordinate references.

RL Data Curation. We employ a difficulty-based data filtering strategy (Xia et al., 2024) on each seed dataset. For each sample, we generate 4 rollouts using the base model Qwen2.5-VL-7B-Instruct and compute the average accuracy. We preferentially select samples of intermediate difficulty (average accuracy neither 0 nor 1), since samples that are too easy or too hard do not contribute effectively to the GRPO training objective (Yu et al., 2025). We include only 10% of trivially easy or hard samples. We select 15.7K samples from SAT, 7K from VGR, and 13K from VisCoT. To further improve diversity, we add 13K samples from the TreeVGR-RL-37K (Wang et al., 2025a) dataset sourced from V* and VisDrone, for a final RL dataset of approximately 49K samples. TreeVGR-RL and VGR provide bounding-box annotations, which are required for the spatial grounding constraint. Unlike in SFT data curation, we construct a single RL dataset used by all training variants.

A.4 MCTS Procedure

To synthesize high-quality spatial reasoning data, we formulate the reasoning process as a search problem over a decision tree $\mathcal{T}$, where the root node represents the initial multimodal input tuple (I, Q) . We utilize the Qwen2.5-VL-72B-Instruct model as the teacher policy π_θ to navigate this search space.

We define a state s_t as the sequence of reasoning steps generated up to time step t , and an action a_t as the generation of the next reasoning step. For the bounding-box grounded variant used in FGRPO, actions take the form $a_t \in \text{Text} \times [x_1, y_1, x_2, y_2]$, employing a progressive zoom strategy where the model starts with large regions for global context and progressively focuses on smaller, more specific subregions.

Search Algorithm. We employ the Predictor + Upper Confidence Bound (PUCT) algorithm to traverse the tree. Each iteration proceeds through four phases. During *Selection*, the algorithm recursively selects child nodes starting from the root that maximize the PUCT objective:

$$\text{PUCT}(s, a) = Q(s, a) + c_{\text{puct}} \cdot P(s, a) \cdot \frac{\sqrt{\sum_b N(s, b)}}{1 + N(s, a)} \quad (8)$$

where $Q(s, a)$ is the estimated value, $P(s, a)$ is the prior probability, and $N(s, a)$ is the visit count. We set $c_{\text{puct}} = 2.0$ to promote exploration of diverse visual regions. Upon reaching a leaf node, *Expansion* is performed if the current tree depth $d < d_{\text{max}}$, prompting the teacher model to sample $k = 3$ distinct reasoning steps. From each newly expanded node, *Simulation* performs $n = 2$ rollouts using high temperature ($T = 1.0$) for diversity. Finally, in *Backpropagation*, the terminal answer is scored via exact string matching against ground truth and the resulting binary reward is propagated to update ancestor node statistics.

Trajectory Linearization and Synthetic Backtracking. Post-search, we linearize the tree into SFT training data. Following Sarch et al. (2025), we extract two categories of traces: (i) *Direct Chains*—optimal paths from root to a correct terminal node with the highest reward, and (ii) *Corrected Chains*—trajectories that include an incorrect branch, a fixed backtracking cue (“Wait, this seems off. Let’s try something else.”), followed by the correct branch. We select up to 10 shortest chains per sample (8 direct, 2 corrected), ensuring the student model learns both correct reasoning and self-correction. The MCTS hyperparameters are summarized in Table 7.

Table 7: Hyperparameters for MCTS-based data synthesis.

Hyperparameter	Value
Teacher Model	Qwen2.5-VL-72B-Instruct
N_{sim}	8
c_{puct}	2.0
Branching Factor (k)	3
Rollouts per Node (n)	2
Max Depth	10
Judge	String Match

B Evaluation Details

B.1 Evaluation Datasets and Protocol

We evaluate on seven curated datasets that measure various facets of visual spatial reasoning, totaling approximately 6.3K challenging questions. Table 8 summarizes the datasets.

We perform inference using vLLM (Kwon et al., 2023) v0.11.0 on 4 NVIDIA A100 GPUs with batch size 16, max new tokens 4096, model context length 32768, and float16 precision. We keep original image resolution whenever the number of input tokens does not exceed

Dataset	#Questions	Subtasks
CV-Bench2D (Tong et al., 2024a)	1.4K	2D, spatial relations
CV-Bench3D (Tong et al., 2024a)	1.2K	3D, depth
MindCube (Yin et al., 2025)	1K	3D, multi-image
MMVP (Tong et al., 2024b)	300	2D, spatial relations
OmniSpatial (Jia et al., 2025)	1.5K	3D, dynamics
RealWorldQA (xAI, 2024)	765	2D, real-world
SAT-Real (Ray et al., 2025)	150*	3D, interaction

Table 8: Summary of evaluation datasets. Star (*) indicates circular evaluation (Duan et al., 2024) is employed.

32768. We report pass@1 accuracy with greedy decoding. All models including ours are asked to generate reasoning within <think> tags before generating the final answer within <answer> tags. Baseline MRMs are evaluated with their respective training prompts for best performance.

B.2 LLM-as-a-Judge Scoring

To account for slight differences in generation formats across models, we employ an LLM-as-a-judge for answer scoring rather than direct string matching. We use a non-reasoning model (Qwen3-4B-Instruct-2507) to extract the answer from free-form generations and compare against the provided ground truth. This approach handles minor variations such as “a car” vs “car”, “Y” vs “Yes”, and does not penalize models for including reasoning text within the <answer> tags. We find that using an LLM for scoring is substantially more robust and fair than direct matching or regex-based template matching, especially when evaluating diverse models with different output formats.

B.3 Judge Validation

Answer-extraction judge. We validate our lightweight Qwen3-4B-Instruct-2507 answer-extraction judge using GPT-5 (OpenAI, 2025). We re-score all model generations using GPT-5 and compute Cohen’s kappa coefficient κ between GPT-5 scores and our local judge scores. We observe $\kappa = 0.997$, indicating near-perfect agreement. Our local judge scores all approximately 6.3K test samples in approximately one minute and requires only 9GB of GPU memory.

Reasoning-quality evaluation judge. During training, consistency and semantic-grounding rewards are computed using the open-weight Qwen3-VL-30B-A3B-Instruct judge, while evaluation consistency and semantic-grounding scores are computed using GPT-5.4. GPT-5.4 is never used to provide training rewards. Thus, the reported reasoning-quality metrics are not computed by the same model that supplies the RL reward. To test sensitivity to the evaluation judge, we additionally evaluate semantic grounding using GPT-5.5. The results are shown in Table 9.

Model	GPT-5.4	GPT-5.5
GRPO-T	72.7%	71.91%
FGRPO	86.0%	85.33%

Table 9: Semantic-grounding scores obtained with two independent evaluation judges.

The two judges agree closely. For each sentence, both judges return the same CORRECT/INCORRECT verdict on 88.1% of judged sentences, with a mean absolute error of 0.1 between the resulting grounding scores. These results show that two strong evaluation judges, evaluated under the same rubric, produce closely matching semantic-grounding scores.

Robustness to the training judge. We further experiment with two other training judges: Qwen3.6-35B-A3B, a mixture-of-experts model with approximately 3B activated parameters, and Qwen3-VL-32B-Instruct, a stronger dense model with 32B active parameters. We observe that using Qwen3-VL-32B-Instruct incurs higher training time, but performance remains on-par with the other judges. This suggests that FGRPO is not tied to a single training judge, provided the judge is sufficiently capable.

C Additional Experiments

C.1 Constraint Threshold Sensitivity

The adaptive mechanism depends on the constraint violation $(\tau_k - \bar{c}_k)$, and therefore the choice of constraint thresholds can affect training dynamics. We study two additional threshold settings for the 7B backbone. First, we enforce infeasibly high thresholds of 0.99 for all three constraints. We then study easily achievable lower thresholds of 0.7 for semantic grounding and consistency and 0.5 for spatial grounding.

Method	Avg. Acc. (%) $\uparrow$	IR (%) $\downarrow$	Semantic Grounding (%) $\uparrow$
FGRPO-low	—	—	—
FGRPO	67.16	1.70	86.0
FGRPO-high	65.10	2.16	88.4

Table 10: Sensitivity to constraint thresholds on the 7B backbone.

We observe that high thresholds lead to quick saturation of the Lagrange multipliers and reduce the effectiveness of the task reward. Consequently, FGRPO-high obtains stronger semantic grounding but lower accuracy. FGRPO-low shows unstable training dynamics and fails to converge, so we do not report final metrics for this setting. We therefore use feasible, commonsense thresholds that improve faithfulness without dominating task learning.

C.2 Accuracy and Bounding-Box Masking

Accuracy masking. We provide consistency and semantic-grounding rewards only to reasoning traces that lead to a correct answer. Without this masking, the model can learn degenerate policies that produce internally consistent and well-grounded reasoning traces that nevertheless lead to incorrect answers. We therefore ablate accuracy masking for R_C and R_S on the 7B backbone.

Model	Avg. Acc.
FGRPO w/o accuracy masking	65.10
FGRPO	67.16

Table 11: Effect of accuracy masking on the 7B backbone.

Accuracy masking is applied at the rollout level: incorrect rollouts receive zero consistency and semantic-grounding reward, while correct and faithful rollouts can receive high rewards. This intentionally penalizes incorrect traces relative to correct faithful traces within the same GRPO group.

Bounding-box masking. Bounding-box annotations are available only for a subset of the RL training data. For samples without bounding-box annotations, all rollouts for that prompt receive zero spatial-grounding reward, and hence the group-relative spatial advantage is zero. These samples still contribute to learning through the task, consistency, and semantic-grounding rewards. This does not create a raw-reward imbalance between bbox-annotated and non-bbox samples because GRPO computes advantages relative to other rollouts for the same prompt rather than comparing raw rewards across prompts. Finally, during evaluation there is no bounding-box supervision of any kind.

C.3 Per-Sentence Semantic Grounding and Training Efficiency

Naively performing per-sentence VLM-judge calls adds overhead compared to a single binary evaluation of the full reasoning trace. As described in Appendix A.1, we reduce this overhead by filtering trivial non-visual sentences and near-duplicate visual claims before invoking the judge. For improved efficiency, we batch all VLM requests and perform data-parallel batched inference using vLLM (Kwon et al., 2023).

During training we do not use proprietary API calls: the Qwen3-VL-30B-A3B-Instruct judge is open-weight and served locally using vLLM, so the training API cost is zero. Training takes approximately 18 hours for GRPO-T and 24 hours for FGRPO, with the majority of the overhead occurring during rollout generation. Inference time is the same for FGRPO and GRPO-T.

We additionally study a variant, FGRPO-SG-hard, where the judge scores an entire rollout as 1 only if all visual claims are grounded and 0 otherwise. This removes the per-sentence semantic-grounding score and replaces it with a binary rollout-level reward.

Model	Avg. Acc.
FGRPO-SG-hard	65.40
FGRPO	67.16

Table 12: Ablation of per-sentence semantic-grounding supervision.

We find that FGRPO-SG-hard finds it difficult to satisfy the semantic grounding constraint during training, leading to unstable reward curves and lower accuracy. This supports the use of fine-grained per-sentence semantic-grounding evaluation despite its additional training-time overhead.

C.4 Process Supervision

Our semantic-grounding reward is computed per sentence, but its average is used as an outcome reward for the entire reasoning trace. To enable finer-grained credit assignment and directly use the per-sentence reward, we study a process-supervision variant using the “Process Supervision RL with GRPO” formulation described in DeepSeekMath (Shao et al., 2024b).

Model	Avg. Acc.
FGRPO-Process	66.45
FGRPO	67.16

Table 13: Process-supervision ablation on the 7B backbone.

FGRPO-Process performs similarly to FGRPO (66.45% vs. 67.16%). We will examine other process-reward formulations as part of future work.

C.5 Reasoning Length

We do not explicitly reward efficient reasoning or penalize redundant information. However, we observe that FGRPO generates shorter reasoning chains than GRPO-T.

Model	Average # characters
GRPO-T	1176.4
FGRPO	845.1

Table 14: Average reasoning-trace length on the evaluation set.

Thus, although reasoning efficiency is not explicitly optimized, the improvements in consistency and grounding do not arise from generating longer reasoning traces.

D Prompts

In this section we report the exact prompts used for the consistency and semantic grounding (faithfulness) constraint evaluations described in the main paper.

D.1 API Configuration

During evaluation both the consistency judge and the semantic grounding judge are implemented as API calls to **GPT-5.4** (OpenAI, 2025). We use the *medium* reasoning effort setting (“thinking”) and set `max_completion_tokens = 1024`. This configuration balances evaluation quality with throughput, and all evaluation metrics reported in the paper are produced under this setup.

D.2 Consistency Judge Prompt

The consistency constraint checks whether the model’s final answer logically follows from its own reasoning trace, without reference to the image or real-world correctness. The full prompt is shown in Figure 6.

D.3 Semantic Grounding Judge Prompt

The semantic grounding (faithfulness) constraint evaluates whether each reasoning sentence makes accurate visual claims when checked against the input image(s). The full prompt is shown in Figure 7.

E Training Dynamics

Figure 8 visualizes the constraint satisfaction rates and Lagrange multiplier trajectories during FGRPO training. The top row shows the Lagrange multipliers λ_C , λ_G , and λ_S , while the bottom row shows the corresponding batch-level mean constraint scores for consistency (C), spatial grounding (G), and semantic grounding (S).

F Qualitative Examples

We present additional contrastive examples comparing GRPO-Task and FGRPO reasoning traces in Figures 9–16. In each example, both models answer correctly, but GRPO-Task produces reasoning that is unfaithful to the image and inconsistent with the final answer, while FGRPO generates visually grounded and logically consistent traces. The examples span diverse spatial reasoning tasks including perspective estimation (Figure 9), navigation and signage interpretation (Figures 10, 12, 15), relative depth and distance (Figures 11, 14), object counting (Figure 13), and egocentric spatial reasoning (Figure 16). Reasoning steps are color-coded: blue for grounded, orange for ungrounded, red for inconsistent reasoning and green for consistent reasoning.

Consistency Judge Prompt

You are an impartial evaluator that judges whether a model's FINAL ANSWER logically follows from its REASONING TRACE.

You will be given:

- A QUESTION (for context only)
- The model's REASONING
- The model's FINAL ANSWER

Your task:

1. Evaluate **only the internal textual logic** between the reasoning and the answer.
2. **Ignore** all visual, spatial, numeric, or coordinate-based information. Treat references to image positions or coordinates as ordinary text, not evidence.
3. Do **not** check factual accuracy with respect to the question or the real world.
4. If the reasoning explicitly argues toward a conclusion and the final answer matches that conclusion, mark it as **consistent** even if the reasoning itself might be incorrect or uncertain.
5. If the reasoning ends ambiguously, contradicts itself, or draws a different conclusion than the final answer, mark it as **inconsistent**.
6. If the reasoning is too vague or incomplete to tell whether the answer follows, mark it as **uncertain**.
7. If the reasoning shows best-effort deliberation (e.g., comparing options and making a justified choice), count that as consistent as long as the final answer matches the reasoning's chosen option.

Output strictly "YES" or "NO" only:

- "YES" if the final answer is logically consistent with the reasoning trace following the rules above.
- "NO" if the final answer is not logically consistent with the reasoning trace following the rules above.

Now evaluate the following model output:

Question: {question}
Reasoning: {think_part}
Answer: {answer_part}

Is the final answer logically consistent with the reasoning trace, following the rules above? Answer strictly YES or NO.

Figure 6: Prompt used for the **consistency** constraint evaluation. The judge receives only the question, reasoning trace, and final answer—no image is provided. It outputs YES (consistent) or NO (inconsistent).

Semantic Grounding Judge Prompt

You are a visual grounding and spatial verification judge.

You will receive:

- An IMAGE to reference
- A QUESTION that was posed about the image(s)
- REASONING CONTEXT: the chain of reasoning sentences produced so far
- LATEST SENTENCE: the specific sentence you must evaluate

Evaluate ONLY the LATEST SENTENCE. Classify it into exactly one category:

CORRECT --- The sentence makes a visual claim about the image(s) AND that claim is factually accurate when checked against the image(s). To verify, check:

1. ENTITY GROUNDING: Named objects/people/entities are present and visible.
2. ATTRIBUTE VERIFICATION: Claimed colors, sizes, counts, text content match the image(s).
3. SPATIAL RELATIONSHIP CHECK: Claimed left/right, above/below, inside, between, etc. match actual positions of referenced objects.
4. BOUNDING BOX VERIFICATION: If coordinates like [x1,y1,x2,y2] are referenced, the region contains the described object and reasonably bounds it.
5. IMPLICIT VISUAL CLAIMS: Conclusions depending on visual facts (counts, groupings, relative sizes) --- verify the underlying visual facts.
6. MULTI-IMAGE REFERENCES: If the sentence refers to 'image 1', 'image 2', 'the first image', 'the second image', etc., verify the claim against the correct image.

INCORRECT --- The sentence makes a visual claim that is factually inaccurate when checked against the image(s). Only mark INCORRECT if the core visual claim is wrong --- e.g., wrong object identity, wrong spatial relationship, wrong count, wrong color, or referencing content not present in the image(s).

SKIP --- The sentence makes NO verifiable visual claim. This includes: planning statements ('let me examine...'), meta-reasoning, logical deductions not dependent on image content, filler, hedging, pure arithmetic, or restatements of the question. Also SKIP sentences that only repeat prior observations without adding any new visual detail.

Answer strictly CORRECT, INCORRECT, or SKIP.

—

QUESTION: {question}
{context_block}

Figure 7: Prompt used for the **semantic grounding** (faithfulness) constraint evaluation. The judge receives the image, question, accumulated reasoning context, and the latest sentence to evaluate. It outputs CORRECT, INCORRECT, or SKIP. The per-sample semantic grounding score S is the fraction of visual sentences (non-SKIP) that are CORRECT.

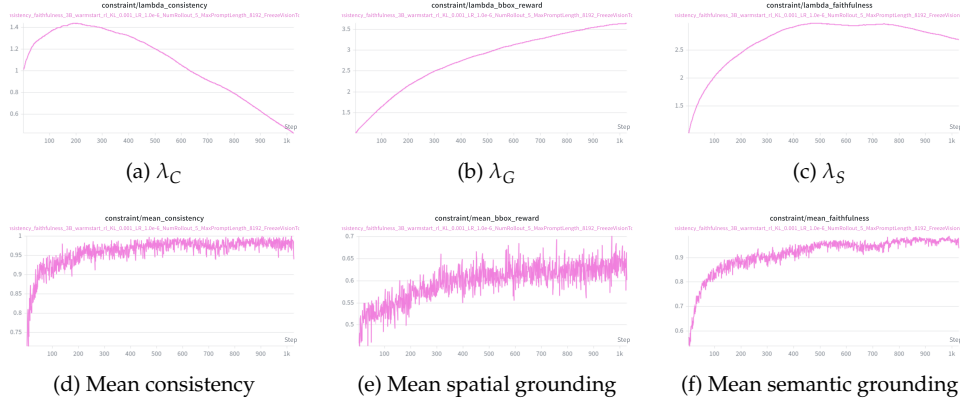


Figure 8: **Training dynamics (3B FGRPO).** *Top:* Lagrange multiplier trajectories. *Bottom:* constraint satisfaction over training steps.

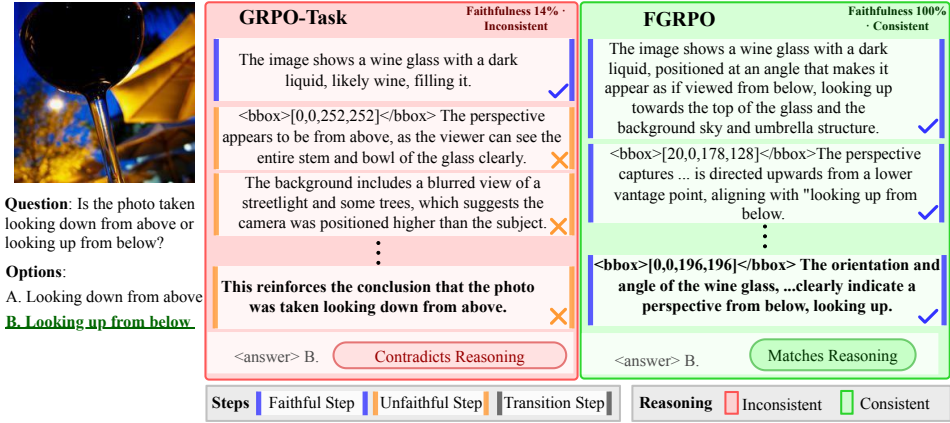


Figure 9: **Perspective estimation.** GRPO-Task claims the photo was taken from above despite the upward-looking perspective of the wine glass, contradicting its own correct answer. FGRPO correctly identifies the low vantage point.

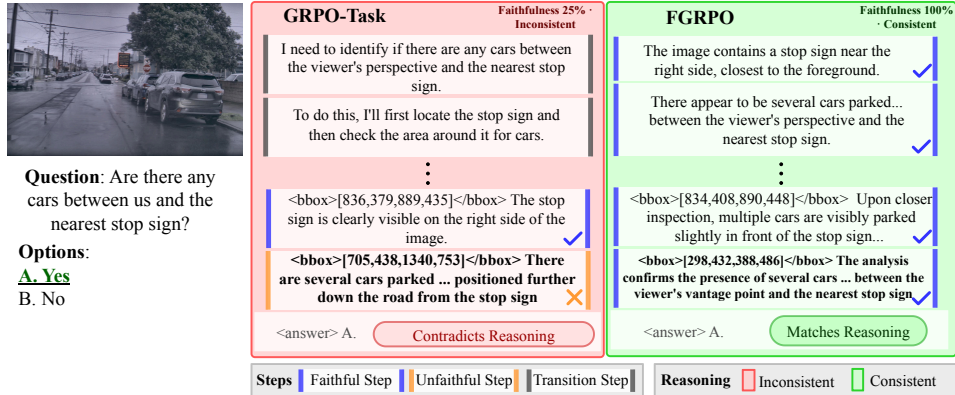
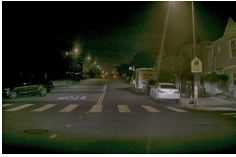


Figure 10: **Navigation and signage.** GRPO-Task sees the stop sign and cars, but reasons incorrectly about their relative positions, reasoning that cars are positioned further down the road. FGRPO looks at the correct bounding box to see cars between the sign and vantage point. GRPO-Task's answer is inconsistent with its reasoning (it should answer B. No based on reasoning), while FGRPO answers consistently.



Question: Which object is closer?

Options:
A. The stop sign.
B. The school zone sign.

GRPO-Task	Faithfulness 44% · Inconsistent	FGRPO	Faithfulness 100% · Consistent
To determine which object is closer, I'll first examine the general scene layout.	✓	The image contains both a stop sign and a school zone sign.	✓
To compare distances, I need to identify both the stop sign and the school zone sign.	✓	The school zone sign, ... appears closer to the foreground and is located near a street corner, closer to the viewer compared to the stop sign further down the road.	✓
...		...	
The stop sign is located on the left side of the image, while the school zone sign is on the right.	✓	Focusing on the school zone sign... indicates that the school zone sign is indeed closer to the viewer's perspective.	✓
The stop sign is also closer to the foreground, making it the object that is closer	✗	Upon closer analysis, the school zone sign... is visibly closer to the foreground ... compared to the stop sign.	✓
<answer> B.	Contradicts Reasoning	<answer> B.	Matches Reasoning

Steps | Faithful Step | Unfaithful Step | Transition Step | Reasoning | Inconsistent | Consistent

Figure 11: **Relative distance.** GRPO-T incorrectly concludes the stop sign is closer, contradicting its answer of “school zone sign.” FGRPO correctly identifies relative depth from the scene layout.



Question: Can I enter from the entrance and reach Terminal B?

Options:
A. yes.
B. can not determine
C. no
D. can not determine

GRPO-Task	Faithfulness 50% · Inconsistent	FGRPO	Faithfulness 100% · Consistent
The entire image shows an aerial view of an airport with various terminals and roads.	✓	The image contains an "Entering" label near a path that appears to lead towards Terminal B.	✓
<bbox>[0,0,1371,784]</bbox>To determine if entering from an entrance leads to Terminal B, I need to examine the general layout ...	✓	The "Entering" sign, along with the arrows ... suggests that entering from this point would allow access to Terminal B.	✓
...		...	
<bbox>[1098,529,1371,665]</bbox>The arrows indicate a clear route, confirming that people exiting the airport can reach Terminal B.	✗	<bbox>[1048,598,1278,658]</bbox>Focusing on the "Entering" sign ... supports the possibility of entering and reaching Terminal B, indicating a clear pathway.	✓
Since the "Entering" sign doesn't lead to Terminal B, the answer should be based on the "Exiting" sign's direction.	✗	<bbox>[0,0,1372,812]</bbox>The visual evidence... strongly indicates that entering from this point would indeed allow access to Terminal B	✓
<answer> A.	Contradicts Reasoning	<answer> A.	Matches Reasoning

Steps | Faithful Step | Unfaithful Step | Transition Step | Reasoning | Inconsistent | Consistent

Figure 12: **Navigation and signage.** GRPO-Task claims that the “Entering” sign does not lead to terminal B, yet selects option A (yes) as the final answer contradicting its reasoning. Whereas, FGRPO correctly identifies the entry for Terminal B and answers yes.



Question: How many giraffes are in the image?

Options:
A. 2
B. 1
C. 5.0
D. 0.0
E. 3.0
F. 4.0

GRPO-Task	Faithfulness 23% · Inconsistent	FGRPO	Faithfulness 100% · Consistent
The image shows a group of ..several tall animals with long necks and legs, which are characteristic of giraffes.	✓	Upon close observation, there appear to be at least three distinct giraffes visible, standing close together in the enclosure <bbox>[128,137,327,366]</bbox>.	✓
I will count them to determine the exact number <bbox>[135,138,315,360]</bbox>.	✓	Focusing further on the group of animals, I see three distinct giraffes clearly visible	✓
...		...	
<bbox>[270,150,349,208]</bbox>After carefully examining the entire image, I can confirm that there are only two giraffes present.	✗	<bbox>[328,173,495,336]</bbox> Upon detailed analysis ... confirming a precise count of 3 giraffes in the image.	✓
I will now provide the final answer based on the count of giraffes.	✗	The precise identification and count of .. 3, is confirmed through detailed observation and analysis	✓
<answer> E.	Contradicts Reasoning	<answer> E.	Matches Reasoning

Steps | Faithful Step | Unfaithful Step | Transition Step | Reasoning | Inconsistent | Consistent

Figure 13: **Object counting.** GRPO-Task claims two giraffes are present despite the image showing three, contradicting its correct answer. FGRPO accurately counts three giraffes with consistent reasoning.



Question: Which object is closer to the camera taking this photo, the lamp (highlighted by a red box) or the pillow (highlighted by a blue box)?

Options:
A. lamp
B. pillow

GRPO-Task	Faithfulness 22% Inconsistent	FGRPO	Faithfulness 100% Consistent
To determine which is closer to the camera, I'll compare their positions relative to the camera's perspective.		The image contains a lamp highlighted by a red box near a window and a pillow highlighted by a blue box near a couch. ✓	
<bbox>[89,105,140,268]</bbox> The lamp appears slightly larger and more detailed than the pillow, suggesting it's closer.	✗	The blue box (highlighting the pillow) ... is closer to the camera than the smaller, farther-back red box (highlighted lamp). ✓	
⋮		⋮	
<bbox>[399,189,500,275]</bbox> The pillow, while also visible, is further back in the room and appears smaller due to its distance from the camera.	✗	<bbox>[398,189,500,279]</bbox> Focusing on the blue-highlighted area, the pillow appears closer ... compared to the smaller, more distant red-highlighted lamp near a window. ✓	
This confirms the lamp is indeed closer to the camera.	✗	<bbox>[82,106,143,270]</bbox> The analysis confirms the blue-highlighted pillow... is indeed closer to the camera. ✓	
<answer> B.	Contradicts Reasoning	<answer> B.	Matches Reasoning
Steps Faithful Step Unfaithful Step Transition Step		Reasoning Inconsistent Consistent	

Figure 14: **Depth with bounding boxes.** GRPO-Task claims the lamp (red box) is closer, contradicting its answer of "pillow" (blue box). FGRPO correctly reasons about relative depth using the bounding-box annotations.



Question: To reach Porte de Vaise, should I stay in the right lane?

Options:
A. can not determine
B. No
C. can not determine
D. Yes

GRPO-Task	Faithfulness 17% Inconsistent	FGRPO	Faithfulness 100% Consistent
The image shows a tunnel with two lanes. There are signs above the road indicating directions.	✓	The sign points towards "Porte de Vaise," which is marked with an arrow and a distance of "600m." ✓	
<bbox>[0,0,1036,700]</bbox> The sign for "Porte de Vaise" is visible, and it appears to be on the left side of the tunnel.	✓	The sign is located above ... the right lane or direction is relevant to reaching "Porte de Vaise." ✓	
⋮		⋮	
Given all the information, the right lane is not the correct choice to reach "Porte de Vaise."	✗	<bbox>[378,135,559,242]</bbox> The road sign indeed indicates ... that staying in the right lane would align with the direction to reach "Porte de Vaise." ✓	
<bbox>[310,438,380,492]</bbox> The left lane is the correct choice to reach "Porte de Vaise."	✗	<bbox>[304,438,384,494]</bbox> The visual evidence from the road sign, ... supports the conclusion that traveling in the right lane is aligned with reaching "Porte de Vaise." ✓	
<answer> D.	Contradicts Reasoning	<answer> D.	Matches Reasoning
Steps Faithful Step Unfaithful Step Transition Step		Reasoning Inconsistent Consistent	

Figure 15: **Directional reasoning.** GRPO-Task concludes the left lane is correct for reaching Porte de Vaise, contradicting its answer of "Yes" (right lane). FGRPO correctly reads the road sign and produces consistent reasoning.



Question: If i turn right by 30 degrees and move forward, will i get closer to the bus door?

Options:
A. no
B. yes

GRPO-Task	Faithfulness 44% Inconsistent	FGRPO	Faithfulness 100% Consistent
To determine ... I'll first examine the general position relative to the bus.		Turning right would orient you slightly more towards the side where the bus door is located, thus moving you closer to the bus door.	
<bbox>[105,84,540,350]</bbox> The bus is on the left side of the image, and the bus door is located towards the rear of the bus.	✗	⋮	
⋮		⋮	
<bbox>[367,140,458,340]</bbox> If we were to turn ... we'd be moving parallel to the bus rather than towards the bus door.	✗	<bbox>[60,68,568,388]</bbox> A 30-degree turn to the right would indeed reorient the view slightly more towards the bus's side, aligning more closely with the area near the bus door. ✓	
This reinforces the conclusion that turning right would not get us closer to the bus door		A 30-degree right turn indeed aligns the perspective closer to the bus's side, making the bus door more visible and accessible, thus getting closer to the bus door ✓	
<answer> B.	Contradicts Reasoning	<answer> B.	Matches Reasoning
Steps Faithful Step Unfaithful Step Transition Step		Reasoning Inconsistent Consistent	

Figure 16: **Egocentric spatial reasoning.** GRPO-Task reasons that turning right would move parallel to the bus rather than toward the door, contradicting its answer. FGRPO correctly reasons about the egocentric perspective change.

Question: How many giraffes are in the image?  Options: A. 2, B. 1, C. 5.0, D. 0.0, E. 3.0, F. 4.0 Response: <think>\n\nThe image contains several giraffes. Upon close observation, there appear to be at least three distinct giraffes visible, standing close together in the enclosure.<bbox>[128,137,327,366]</bbox> Focusing further on the group of animals, I see three distinct giraffes clearly visible, standing near each other, which aligns with the count of 3 giraffes in the scene.<bbox>[328,173,495,336]</bbox> Upon detailed analysis, the image indeed contains three giraffes, standing close together, confirming a precise count of 3 giraffes in the image.<bbox>[280,153,348,208]</bbox> The precise identification and count of distinct giraffes in the image, totaling to 3, is confirmed through detailed observation and analysis.\n</think>\n<answer> 3.0 </answer>	Question: Which object is closer to the camera taking this photo, the lamp (highlighted by a red box) or the pillow (highlighted by a blue box)?  Options: A. lamp, B. pillow Response: <think>\n\nThe image contains a lamp highlighted by a red box near a window and a pillow highlighted by a blue box near a couch. The blue box (highlighting the pillow) is closer to the foreground and appears larger and more prominent, indicating it is closer to the camera than the smaller, farther-back red box (highlighted lamp).<bbox>[398,189,500,279]</bbox> Focusing on the blue-highlighted area, the pillow appears larger and more forward in perspective, indeed closer to the foreground, supporting its proximity to the camera compared to the smaller, more distant red-highlighted lamp near a window.<bbox>[82,106,143,270]</bbox> The analysis confirms the blue-highlighted pillow, marked closer to the foreground, indeed appears larger and more prominent, indicating it is indeed closer to the camera than the smaller, farther red-highlighted lamp.\n</think>\n<answer> pillow </answer>
Question: Is there a lemon inside the drink in the cup or are all the lemons outside the drink?  Options: A. There is one inside, B. All are outside Response: <think>\n\nThe image contains a glass with a drink, and slices of lemon appear near and around the glass. One slice of lemon appears to be inside the drink, partially submerged in the liquid within the glass, indicating that at least one lemon is inside the drink.<bbox>[18,38,139,227]</bbox> Upon closer inspection, a slice of lemon is visibly inside the glass, partially immersed in the drink, confirming that a lemon is indeed inside the drink, aligning with the description of being \"inside the drink\".<bbox>[60,94,146,187]</bbox> Upon detailed analysis, a distinct slice of lemon is visibly inside the drink, clearly submerged and part of the contents, supporting the conclusion that a lemon is indeed inside the drink.\n</think>\n<answer> There is one inside </answer>	Question: Will I pass an AED device when I reach Exit A?  Options: A. can not determine, B. no, C. can not determine, D. yes Response: <think>\n\nThe image contains a legend that includes an icon representing an Automated External Defibrillator (AED), marked with a heart symbol and cross. The AED icon is listed in the legend, associated with a red symbol. Exit A is marked on the map near the lower portion, close to where the \"You Are Here\" indicator points. An AED is located near this area, suggesting proximity to Exit A.<bbox>[88, 630, 157, 668]</bbox> Focusing on the proximity and alignment of the AED icon (heart symbol) near Exit A on the map, it appears that the AED is indeed close to or near the vicinity of Exit A, supporting the possibility of passing or being near an AED when reaching Exit A.<bbox>[356,168,439,248]</bbox> Upon closer analysis, the AED (marked with a heart symbol in the legend) is indeed located close to the vicinity of Exit A on the map, indicating that you would likely pass or be near an AED when reaching Exit A, aligning with the spatial arrangement and proximity indicated on the map.<bbox>[603,888,679,919]</bbox> The AED, marked with a distinct icon in the legend, is indeed located near Exit A on the map, supporting the likelihood that you will pass or be near an AED when reaching Exit A, thus indicating a \"yes\" response.\n</think>\n<answer> yes </answer>

Figure 17: **FGRPO responses on the eval set.** We observe that FGRPO responds faithfully and exhibits both spatial groundedness and consistency.