SVQA-R1:REINFORCING SPATIAL REASONING IN MLLMS VIA VIEW-CONSISTENT REWARD OPTIMIZATION

Peiyao Wang **Haibin Ling***
Stony Brook University
{peiyaowang, hling}@cs.stonybrook.edu

ABSTRACT

Spatial reasoning remains a core limitation of current vision-language models (VLMs), which often struggle to understand object relations such as direction, distance, and spatial configuration. In this work, we introduce SVQA-R1, a reinforcement learning framework that improves spatial reasoning by enforcing view consistency during training. At its core is Spatial-GRPO, a group-based reward optimization method that encourages the model to generate consistent answers and reasoning across perturbed views of the same scene. We adopt two complementary perturbation strategies: (1) horizontal flipping, which supervises directional concepts like "left" and "right"; and (2) 2D viewpoint transformations—such as in-plane rotation and perspective warping—which reinforce reasoning about distance and relative positioning. This unified approach enables the model to acquire geometry-aware spatial understanding without relying on supervised fine-tuning. Experiments on multiple Spatial VQA benchmarks demonstrate that SVQA-R1 significantly outperforms strong baselines and produces interpretable, consistent reasoning across diverse viewpoints.

1 INTRODUCTION

Although recent open-source vision-language models (VLMs) such as LLaVA Liu et al. (2023) and Qwen2.5-VL Bai et al. (2025) have achieved impressive results on general visual question answering (VQA) benchmarks, they continue to underperform on tasks requiring spatial reasoning Chen et al. (2024a). This limitation stems from several core issues. First, VLMs often exhibit a strong local visual bias, attending to salient individual objects while failing to model their spatial relationships. For instance, when asked "Is the cup to the left of the laptop?", a model might correctly detect both objects yet fail to reason about their relative positions. This is largely due to training data that prioritize object classification over spatial understanding, resulting in models that rely heavily on object categories and resort to spatial guessing. Second, spatial VQA often demands fine-grained localization, which is especially challenging for VLMs in cluttered scenes or when reasoning about small or occluded objects. Finally, most existing models lack explicit reasoning steps, directly mapping visual input to answers without intermediate logical processing. This absence of structured reasoning hampers their performance on multi-step spatial tasks involving layered relationships, such as "above the fridge and on the table." Together, these factors make general-purpose VLMs ill-equipped for complex spatial reasoning.

Recent work has begun addressing these challenges by incorporating spatially targeted supervision. For example, SpatialVLM Chen et al. (2024a) proposes a synthetic data generation pipeline to create large-scale spatial VQA samples, enabling supervised training that improves performance on spatial tasks. Building on this, SpatialPIN Ma et al. (2024) integrates 3D priors to enhance object relation modeling without altering the underlying VLM architecture. While effective, both approaches rely heavily on supervised fine-tuning (SFT), which can lead to rigid output patterns and limit the model's reasoning flexibility. To overcome these limitations, we turn to reinforcement learning (RL) as a more adaptive alternative, drawing inspiration from the R1 paradigm to encourage dynamic, self-correcting spatial reasoning guided by rule-based feedback.

*Haibin Ling was involved in this work while affiliated with Stony Brook University.

The R1 paradigm, introduced by DeepSeek-R1 Guo et al. (2025), demonstrates that simple rule-based reinforcement learning can significantly enhance the reasoning abilities of large language models (LLMs) without requiring annotated thought processes. Inspired by this, we extend the R1 framework to the spatial reasoning domain and propose SVQA-R1, a novel reinforcement learning approach for spatial VQA. At its core is the Spatial-GRPO mechanism—a group-based reward optimization method tailored to vision-language models. Our framework incorporates rule-based rewards to ensure format correctness, semantic-aware rewards to handle diverse answer types (e.g., bounding boxes, captions, distances, multiple-choice), and most importantly, a view-consistent reward that aligns the model's outputs across spatially transformed views. This view-consistency constraint ensures the model does not rely on superficial cues or spurious correlations, as it must produce consistent answers across different perspectives of the same scene.

Our experiments demonstrate that SVQA-R1 significantly improves performance on spatial reasoning benchmarks. On the Q-Spatial++ dataset Liao et al. (2024), our method improves accuracy by over 30% compared to strong SFT-based baselines and outperforms several state-of-the-art open-source and closed-source models. Notably, SVQA-R1 surpasses Gemini-1.5-Flash gem (2025), and performs competitively with GPT-4o Islam & Moushi (2024), while achieving substantial gains over open-source models such as InternVL-2.5 Chen et al. (2024b) and Qwen2.5-VL Bai et al. (2025). Furthermore, our method yields interpretable, step-by-step reasoning traces through the use of structured prompting, demonstrating that reinforcement learning can improve both performance and interpretability in spatial VQA.

To summarize, our main contributions are:

(1) We introduce a rule-based and semantic-aware reinforcement learning framework for spatial reasoning in VLMs, enhanced with interpretable reasoning via prompting;

(2) We design a novel Spatial-GRPO mechanism that promotes view-consistent action space learning through image-level transformations;

(3) Our SVQA-R1 framework achieves strong results, including a **58%** accuracy on Q-Spatial++, surpassing existing open-source models by a large margin.

2 Related Work

Spatial Reasoning with VLMs Spatial reasoning is pivotal for Spatial VQA, yet early VLMs like CLIP Radford et al. (2021) often struggle with 3D relationships and compositional logic Wang et al. (2024); Patel et al. (2024); Doveh et al. (2023). While MDETR Kamath et al. (2021) improves alignment via spatial attention, precise metric estimation remains a bottleneck Liao et al. (2024). Recent efforts leverage large-scale 3D data: SpatialVLM Chen et al. (2024a) and MM-Spatial Daxberger et al. (2025) utilize massive 3D datasets (e.g., CA-VQA) to enhance quantitative reasoning and CoT capabilities Wei et al. (2022), while SpatialPIN Ma et al. (2024) introduces zero-shot 3D priors for robotics. Despite benchmarks like GQA Hudson & Manning (2019) and VSR Liu et al. (2023) exposing weaknesses in multi-hop queries, and neuro-symbolic Yi et al. (2018) or RL-guided Salter et al. (2021) methods attempting task decomposition, explicit CoT reasoning and adaptive learning for complex Spatial VQA remain underexplored. Our work bridges this gap to enhance VLM generalization and compositional reasoning.

Chain-of-Thought and R1-style Reinforcement Learning In Vision-Language Models Recent research enhances vision-language model (VLM) reasoning via structured prompting and reinforcement learning (RL). Beyond extending Chain-of-Thought (CoT) prompting Wei et al. (2022) to multi-step visual and spatial tasks Zhao et al. (2023); Wang et al. (2022), R1-style RL Guo et al. (2025) has emerged to promote verifiable, rule-guided learning. Leveraging Group Relative Policy Optimization (GRPO) Shao et al. (2024), models like Visual-RFT Wu et al. (2024b), VLM-R1 Zhou et al. (2024), and Reason-RFT Wu et al. (2024a) utilize task-specific rewards or CoT supervision to outperform supervised fine-tuning. This framework further extends to temporal reasoning via Video-R1 Xu et al. (2024), with lightweight variants like TinyLLaVA-Video-R1 Li et al. (2024) and VideoChat-R1 Zhang et al. (2024) demonstrating that RL-driven training significantly bolsters spatial and temporal understanding across scales.

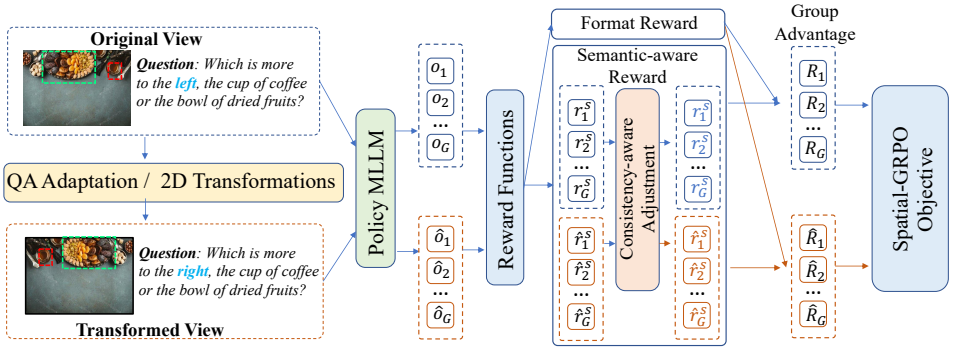


Figure 1: Overview of SVQA-R1. We generate semantically consistent question–answer pairs under QA adaptation and 2D view transformations and evaluate model outputs using format and semantic-aware rewards with consistency-aware adjustment. The resulting group advantages are aggregated to optimize the policy via the Spatial-GRPO objective.

Together, these CoT-based prompting and R1-style RL methods reflect a shift towards more explicit and interpretable reasoning in multimodal systems, enabling models to generalize better across complex spatial, temporal, and causal reasoning tasks.

3 Method

We begin by presenting our approach to generating consistent QA pairs in Section 3.1, and then introduce the mixed reward strategy in Section 3.2 and the overall Spatial-GRPO training procedure in Section 3.3.

3.1 View-consistent Reasoning

Spatial distances remain invariant under changes in image perspective, while directional relationships (e.g., “left” and “right”) are typically reversed under horizontal flipping. To encourage the policy model to learn a view-consistent action space, we apply horizontal flipping to supervise directional reasoning and use 2D viewpoint transformations, i.e., in-plane rotation and perspective warping, to supervise consistency on distance-related questions. These transformations simulate minor viewpoint changes while preserving the underlying spatial structure, enabling the model to develop a more robust and geometry-aware understanding of space.

Mirror-Consistent via QA Adaptation To promote mirror-consistent spatial reasoning, we construct horizontally flipped image samples paired with semantically aligned question-answer (QA) pairs. A key challenge arises from the fact that horizontal flipping alters the spatial configuration of the scene—terms like “left” and “right” may reverse—necessitating corresponding adjustments to the associated QA content to maintain logical consistency.

To address this, we leverage GPT-4o Islam & Moushi (2024) to automatically generate revised QA pairs for each flipped image. Given the original image’s question and answer, we prompt the model to produce a logically valid version that reflects the spatial changes in the mirrored view. However, we observe that naïve prompting often results in superficial edits. Specifically, GPT-4o tends to mechanically swap “left” with “right” without fully reasoning about the relational semantics of the scene. This frequently produces logically inconsistent or contradictory QA pairs.

To improve reliability, we enhance the prompting strategy by explicitly instructing GPT-4o to reason about the flipped image and verify the spatial correctness of the revised QA pair. The refined prompt includes specific guidance to assess object identities and their relative positions before producing the final output. This enhancement significantly improves the semantic accuracy of the generated QA pairs, particularly in complex, multi-object scenes. Additional prompt details, as well as examples of incorrect and corrected QA pairs for flipped images, are provided in the supplementary material.

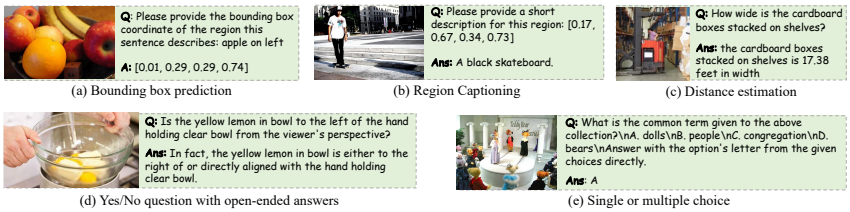


Figure 2: A diverse set of open-ended spatial question-answer types.

Novel-View Consistency via 2D Transformations Beyond horizontal flipping, we further encourage spatial consistency by synthesizing novel-view images through simple 2D viewpoint transformations, under the assumption that relative object distances remain invariant across views. This strategy allows the model to develop robustness to changes in image prescription while preserving the underlying spatial semantics. Specifically, we apply in-plane rotations and perspective warping to approximate minor camera shifts that alter the visual layout without disrupting spatial relationships. These transformed views are paired with the original QA annotations when the question semantics remain unchanged (e.g., for distance-based queries), enabling the model to learn consistent reasoning across diverse viewpoints. While mirror-based supervision emphasizes directional understanding (e.g., "left" vs. "right"), this complementary approach strengthens metric spatial reasoning, ultimately enhancing the spatial grounding capabilities of the model.

3.2 Mixed Reward Design

Unlike structured tasks such as classification or numerical prediction, spatial reasoning in vision-language models (VLMs) often involves open-ended question-answering with diverse and linguistically rich output. As illustrated in Figure 2, model responses may vary in form, ranging from short directional terms to complete relational descriptions. This inherent variability presents challenges for rule-based reward functions, which rely on exact string matching or rigid templates. Applying such methods directly risks penalizing semantically correct answers that deviate from expected wording, thereby limiting the expressiveness and performance potential of VLMs.

To accommodate this flexibility, we adopt a semantic-based reward that evaluates the alignment between the model's output and the ground truth based on meaning rather than surface form. This approach allows the model to generate natural and fluent responses while still being guided by accurate spatial understanding. In addition, motivated by findings from recent work Chen et al. (2024a) that highlight the benefits of Chain-of-Thought (CoT) reasoning in improving multistep spatial inference, we incorporate a format reward to further encourage structured and interpretable outputs. Together, these components support a more flexible and faithful reward mechanism tailored to the open-ended nature of spatial reasoning tasks.

Format Reward r^f. We include a binary signal that assigns 1 if the output follows the required structural format (e.g., `<think>...</think> <answer>...</answer>`), and 0 otherwise.

Semantic-aware Reward r^s. To capture semantic similarity beyond surface-level token overlap, we use a Sentence-BERT-based reward. Specifically, we compute sentence embeddings using a pre-trained Sentence-BERT Reimers & Gurevych (2019) model (e.g., all-MiniLM-L6-v2) and measure the cosine similarity between the predicted and reference answers. This embedding-based reward enables the model to recognize semantically equivalent expressions that lexical metrics such as BLEU fail to identify, for example, assigning a high reward to predictions like "sofa" when the ground truth is "couch". In addition, Sentence-BERT-based reward can measure the numerical difference when handling those answer types, such as distances and bounding boxes. Such a reward unification can stabilize the training, as we observe.

Final Reward. The total reward is defined as a weighted sum of all components:

$$r = \lambda_1 \cdot r^f + \lambda_2 \cdot r^s \tag{1}$$

where λ_1, λ_2 are hyperparameters. This composite reward encourages the model to generate answers that are structurally valid, lexically and semantically correct, and numerically plausible.

3.3 Spatial Group Relative Policy Optimization

While GRPO has demonstrated promising results in general reasoning tasks, it lacks explicit reward signals that are specifically designed to guide spatial reasoning. This makes it inadequate for training Vision-Language Models (VLMs) to understand and reason about spatial relationships such as "left of", "above", or "closer to". To address this limitation, we introduce Spatial Group Relative Policy Optimization (Spatial-GRPO), a joint reinforcement learning framework that explicitly targets spatial understanding.

Our approach is grounded in two key observations about human spatial cognition: (1) quantitative measurements such as distance between objects remain invariant under horizontal mirroring, and (2) qualitative relationships like "in front of" or "next to" are preserved, while directional expressions such as "left" and "right" require symmetric adjustment. We aim to replicate this perceptual stability, encouraging the model to maintain semantically accurate spatial reasoning for both views.

Specifically, given an original image and question pair (I, Q), and the horizental fliped imge and update queation is tuple $(\hat{I}, \hat{Q})$. They will obatian grouped sample outputs $\{o_i\}_{i=1}^G$ and $\{\hat{o}_i\}_{i=1}^G$ from the old policy model π_{old}. Then we compute the rewards $r_i, \hat{r}_i$ for each sample in the original and flipped groups independently, based on their respective question-answer alignments. To promote consistent and accurate spatial reasoning across both the original and mirrored views, we aim to minimize the discrepancy between the two reward distributions. Specifically, we compare the aggregated rewards $\texttt{Avg}(\{r_i\}_{i=1}^G)$ and $\texttt{Avg}(\{\hat{r}_i\}_{i=1}^G)$ from the original and flipped groups and penalize the group that achieves a substantially higher score. This design reflects our assumption that a truly spatially grounded model should perform similarly across both views, and that large discrepancies indicate a lack of spatial consistency.

To operationalize this idea, we define a consistency-aware joint reward that encourages both high performance and inter-view agreement. Specifically, we use the semantic reward difference between the original and flipped views to quantify the discrepancy in model responses. This choice reflects our focus on semantic alignment, as it is independent of surface-level formatting variations in the generated outputs. We define the semantic difference as:

$$\Delta = \texttt{Avg}(\{r_i^s\}_{i=1}^G) - \texttt{Avg}(\{\hat{r}_i^s\}_{i=1}^G) \tag{2}$$

The modified reward for the semantic rewards is as follows:

$$r_i^{s'} = r_i^s + \eta|\Delta|, \quad \text{if} \quad r_i^s > \delta \quad \text{and} \quad \Delta >= 0 \tag{3}$$

$$\hat{r}_i^s = \hat{r}_i^s - \eta|\Delta|, \quad \text{if} \quad \hat{r}_i^s > \delta \quad \text{and} \quad \Delta < 0 \tag{4}$$

The final Spatial-GRPO maximizes the following objective:

$$\begin{aligned} \mathcal{J}_{\text{Spatial-GRPO}}(\theta) =& \mathbb{E}_{q,\hat{q},\{o_i,\hat{o}_i\}\sim\pi_{\text{old}}} \left[\frac{1}{2G} \sum_{i=1}^{G} (R_i + \hat{R}_i) \right] - \beta D_{\text{KL}}(\pi_\theta \parallel \pi_{\text{ref}}), \\ R_i =& \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \min(\alpha_t(\theta) A_i,\, \text{clip}(\alpha_t(\theta), 1-\epsilon, 1+\epsilon) A_i), \end{aligned} \tag{5}$$

where $\alpha_t(\theta) = \frac{\pi_\theta(o_{i,t}|q, o_{i,<t})}{\pi_{\text{old}}(o_{i,t}|q, o_{i,<t})}$, $A_i = \frac{r_i - \text{mean}(\{r_i\}_{i=1}^G)}{\text{std}(\{r_i\}_{i=1}^G)}$ is the estimated advantage, η is a scaling coefficient, β is the KL penalty coefficient, and π_θ, π_{old}, π_{ref} denote the current, old, and reference policy models, respectively. ϵ is the threshold for clipping. q is the input pair of the image and the question. The definition of $\hat{R}_i$ is similar to R_i.

4 Experiment

4.1 Experimental Settings

We adopt Qwen2.5-VL-3B and Qwen2.5-VL-7B Bai et al. (2025) as our base model and use its pre-trained weights. The maximum generation length is set to 2048 tokens. For the reward computation,

		Metrics			In Range		
Source	Model	Success Rate(%)↑	Samples Completed(%)↑	sMAPE(%)↓	50-100(%)↑	100-150(%)↑	150-200(%)↑
Closed	GPT-4o$^+$ Islam & Moushi (2024)	61.06	-	-	-	-	-
	Gemini-1.5-Flash$^+$ gem (2025)	26.73	-	-	-	-	-
Open	UCSC-VLAA Chen et al. (2025)	25.74	99.01	119.33	10.00	7.99	9.01
	Qwen2.5VL-3B Bai et al. (2025)*	48.51	100.00	80.93	**20.79**	13.86	13.86
	Qwen2.5VL-3B Bai et al. (2025)	21.78	92.08	117.24	9.68	5.38	8.60
	SpaceThinker-Qwen2.5VL-3B Spa (2025)	27.72	98.02	125.56	16.16	7.07	5.05
	InternVL-2.5 Chen et al. (2024b)	20.97	91.02	120.14	8.17	4.69	8.11
	SVQA-R1	**58.42**	**100.00**	**68.36**	18.81	**18.81**	**21.78**

Table 1: The test results on Q-Spatial++ Liao et al. (2024). $*$ indicates models not using the thinking prompt. $+$ means that the results are copied from Q-Spatial++ Liao et al. (2024).

	Metrics			In Range		
Model	Success Rate(%)↑	Samples Completed(%)↑	sMAP (%)↓	50-100(%)↑	100-150(%)↑	150-200(%)↑
Base	21.78	92.08	117.24	9.68	5.38	8.60
SFT	37.62	100.00	109.13	16.83	6.93	13.86
SFT (CoT)	27.72	97.03	124.85	11.22	13.27	4.08
SFT (CoT)*	27.72	98.02	125.56	16.16	7.07	5.05
SVQA-R1	**58.42**	**100.00**	**68.36**	**18.81**	**18.81**	**21.78**

Table 2: Compare the SFT and GRPO on Q-Spatial++ Liao et al. (2024). $*$ indicates the initialization is from UCSC-VLAA Chen et al. (2025), while others are from the Base.

we set the weighting coefficients λ_1 and λ_2 to 0.5, balancing between the semantic and format rewards. When updating the semantic reward, we apply a threshold $\delta = 0.5$, and set the final reward scaling factor $\eta = 1$. During GRPO training, we generate 8 candidate answers per input as a group. We set the gradient accumulation steps to 2 and the per-device batch size to 8. Training is conducted on 8 NVIDIA A6000 GPUs.

4.2 DATASETS

Training Dataset. For training, we adopt the SpaceLLaVA VQA (2024) dataset, which contains more than 28,000 multi-turn dialogues designed for visual question answering (VQA). This dataset includes a rich mixture of spatial VQA and general VQA samples. All training samples are automatically generated using the VQASynth VQA (2024) framework, guided by the spatial prompting techniques introduced in SpatialVLM Chen et al. (2024a). We split the multi-turn dialogues into multiple single-turn data.

Test Dataset. For numerical task evaluation, we adopt the Q-Spatial++ Liao et al. (2024) dataset. Q-Spatial++ is a subset of the Q-Spatial-Bench dataset, designed to evaluate quantitative spatial reasoning in large vision-language models. It comprises 87 freshly captured images and 101 human expert-annotated questions, focusing exclusively on horizontal distances between objects in real-world scenes. To ensure high precision, physical measurements of object distances were taken during image capture, providing accurate ground-truth answers.

In addition, to verify the generalizability of our method, we construct two additional test sets. One is based on Vqasynth_Spacellava VQA (2024) and the other is based on OpenSpaces VQA (2024). Both are generated using the VQASynth pipeline. The OpenSpaces dataset is built by synthesizing spatial VQA samples from the first 30K rows of the localized narratives split of the Cauldron dataset using VQASynth, while Vqasynth_Spacellava samples are collected directly via multi-turn chat interactions. For each source, we collect approximately 1,000 multi-turn dialogues and convert them into single-turn QA pairs in the format of (image, question, answer) from their test set. As a result, the Vqasynth_Spacellava and OpenSpaces test sets contain 3,120 and 5,000 image-question-answer pairs, respectively.

Notably, since our model is only trained on the Vqasynth_Spacellava training set, the OpenSpaces test set can serve as an out-of-distribution benchmark to assess the robustness of spatial reasoning. Furthermore, OpenSpaces is enriched with more diverse and fine-grained spatial VQA questions, whereas Vqasynth_Spacellava includes a broader mixture of general VQA and spatial reasoning

Source	Method	Metrics				Spatial-VQA Type			
		LLM (%)↑	BLEU-1 (%)↑	BLEU-2 (%)↑	sBERT↑	Bbox		Yes/No	Distance
						mIoU(%)↑	Acc.@0.75(%)↑	Acc.(%)↑	Acc.(%)↑
		Vqasynth_Spacellava							
Closed	GPT-4o OpenAI (2024)	65.92	65.02	58.50	89.23	45.02	26.50	60.05	22.51
	Gemini-1.5-Flash gem (2025)	31.52	34.48	27.37	63.22	34.90	20.50	49.77	14.54
Open (3B)	QWen2.5VL-3B* Bai et al. (2024)	45.54	33.33	31.31	68.04	10.00	5.80	32.73	5.91
	QWen2.5VL-3B Bai et al. (2024)	45.33	10.00	7.96	66.17	41.01	19.45	39.89	11.29
	SpaceThinker-QWen2.5VL-3B	39.39	0.00	0.00	16.14	37.00	11.60	29.37	10.75
	InternVL-2.5(4B) Chen et al. (2024b)	46.69	10.30	8.20	67.49	42.24	20.23	41.09	10.95
	UCSC-VLAA Chen et al. (2025)	46.11	0.0	0.0	20.90	42.29	21.84	44.21	13.44
	SVQA-R1	**60.77**	**63.33**	**56.43**	**87.84**	**43.49**	**24.57**	**58.31**	**23.66**
Open (7B)	Qwen2.5VL-7B* Bai et al. (2024)	49.37	37.80	35.62	72.33	14.32	9.64		9.90
	Qwen2.5VL-7B Bai et al. (2024)	49.50	14.52	11.92	70.71	45.08	23.51	43.61	15.30
	InternVL-2.5(8B) Chen et al. (2024b)	50.02	14.20	12.44	71.07	46.52	24.32	45.60	14.80
	UCSC-VLAA Chen et al. (2025)	51.32	8.20	6.52	25.06	46.58	25.90	48.45	17.05
	SVQA-R1	**66.78**	**67.45**	**59.05**	**90.85**	**47.23**	**27.43**	**62.30**	**25.80**
		OpenSpaces							
Closed	GPT-4oOpenAI (2024)	34.54	58.70	44.65	89.54	-	-	41.56	15.88
	Gemini-1.5-Flash gem (2025)	20.56	45.63	30.12	75.41	-	-	29.10	6.71
Open (3B)	QWen2.5VL-3B* Bai et al. (2024)	30.61	33.33	33.33	68.05	-	-	21.37	2.18
	QWen2.5VL-3B Bai et al. (2024)	**32.49**	49.71	32.35	**87.67**	-	-	29.92	3.83
	SpaceThinker-QWen2.5VL-3B	16.29	1.03	1.00	31.29	-	-	17.10	10.81
	InternVL-2.5(4B) Chen et al. (2024b)	28.95	50.41	34.37	81.25	-	-	21.77	4.15
	UCSC-VLAA Chen et al. (2025)	18.03	54.68	37.94	87.66	-	-	22.16	8.15
	SVQA-R1	30.97	**55.00**	**41.68**	85.50	-	-	**39.38**	**14.44**
Open (7B)	Qwen2.5VL-7B* Bai et al. (2024)	33.65	37.53	37.30	73.43	-	-	25.36	6.48
	Qwen2.5VL-7B Bai et al. (2024)	34.40	54.50	37.63	89.54	-	-	33.90	6.89
	InternVL-2.5(8B) Chen et al. (2024b)	32.95	53.44	38.40	85.22	-	-	26.73	8.45
	UCSC-VLAA Chen et al. (2025)	23.04	57.60	40.93	89.61	-	-	26.18	12.25
	SVQA-R1	**35.12**	**59.04**	**45.63**	**90.52**	-	-	**43.31**	**17.43**

Table 3: Compare the results on the open-ended spatial VQA task on Vqasynth_Spacellava VQA (2024) and OpenSpaces VQA (2024). $*$ indicates models do not use the thinking prompt.

queries. This setup allows us to probe the model's ability to generalize across both spatial and non-spatial domains.

4.3 Evaluation Metrics

Numerical Task. We adopt three metrics to evaluate the numerical prediction performance: success rate (%), samples completed (%), and Symmetric Mean Absolute Percentage Error (sMAPE %). Success rate (%) measures the proportion of predictions that fall within a predefined tolerance range of the ground truth. It reflects how often the model produces "acceptable" answers. We define a prediction as success if $\max(\text{GT}/\text{Pred}, \text{Pred}/\text{GT}) < 2$. Samples Completed (%) denotes the percentage of test cases where the model outputs a parsable and meaningful numerical value, indicating the reliability and completeness of the model's responses. sMAPE (%) evaluates the prediction accuracy in a scale-invariant manner by comparing predicted values $\hat{y}$ to the ground truth y. It is defined as:

$$\text{sMAPE} = \frac{1}{n} \sum_{i=1}^{n} \frac{|\hat{y}_i - y_i|}{(|\hat{y}_i| + |y_i|)/2 \times 100\%} \tag{6}$$

To ensure fairness, we only compute sMAPE over test cases where the model produces a valid numerical prediction. Cases without a parseable number are excluded from the average and instead reflected in the "Samples Completed (%)" metric.

Open-ended Spatial VQA Task. To evaluate the accuracy of open-ended answers in our Spatial VQA task, we adopt three complementary metrics: BLEU-1 Papineni et al. (2002), Sentence-BERT Reimers & Gurevych (2019) cosine similarity, and an LLM-based evaluation using GPT-4o OpenAI (2024). BLEU-1 computes unigram overlaps between the predicted and reference answers. While it captures basic lexical agreement, it often fails to reflect semantic correctness, especially when answers are paraphrased or differ in surface form. To address this limitation, we additionally report Sentence-BERT similarity, which computes the cosine similarity between sentence-level embeddings of the predicted and reference answers. This embedding-based metric provides a more flexible evaluation of semantic alignment and is better suited for free-form VQA responses where multiple correct phrasings may exist. Finally, we incorporate an LLM-based evaluation using GPT-4o, which is asked to assess the correctness of the prediction. This metric leverages GPT-4o's advanced reasoning to provide a human-like judgment.

	Modules				Q-Spatial++			Vqasynth_Spacellava		
Model	Base	Format Reward	Semantic Reward	spatial -GRPO	Success Rate (%) ↑	Samples Completed (%) ↑	sMAPE (%) ↓	llm (%) ↑	BLEU-1 (%) ↑	sBERT ↑
(a)	✓				21.78	92.08	117.24	45.33	10.00	66.17
(b)		✓			33.66	94.06	107.95	49.99	51.55	85.84
(c)			✓		49.50	100.00	81.61	60.34	0.00	16.14
(d)		✓	✓		52.48	100.00	74.49	60.35	54.29	83.51
(e)		✓	✓	✓	**58.42**	**100.00**	**68.36**	**60.77**	**63.33**	**87.84**

Table 4: Ablation on reward function and modules on qunatity dataset Q-Spatial++ and dataset Vqasynth_Spacellava.

Further, we categorize questions into types (e.g., binary, descriptive, spatial) and perform type-specific evaluation. This allows us to more accurately assess the model's behavior under different reasoning demands. To evaluate bounding box predictions in our Spatial VQA task, we adopt two metrics: mean IoU (mIoU) and Accuracy@0.75. mIoU computes the average Intersection over Union between predicted and ground-truth boxes, reflecting overall localization quality. Accuracy@0.75 measures the percentage of predictions with $IoU \geq 0.75$, indicating the proportion of highly accurate localizations. For Yes/No questions, we report standard answer accuracy, measuring the proportion of predictions that exactly match the ground-truth labels. For Distance Estimation, we consider a prediction correct if it falls within the range from 50% to 200% of the ground truth, capturing relative correctness under spatial uncertainty.

4.4 Evaluation on Numerical Tasks

Our method, SVQA-R1, outperforms all open-source models across nearly all evaluation metrics and shows a competitive result compared to GPT-4o. In particular, it achieves a 58.42% success rate, which represents a 31% absolute improvement over SpaceThinker-Qwen2.5VL-3B, a model fine-tuned with supervised Spatial-CoT data. This demonstrates the effectiveness of our reinforcement learning strategy in enhancing spatial quantitative reasoning. Moreover, SVQA-R1 surpasses Qwen2.5VL-3B* and Qwen2.5VL-3B by 47% and 10% in success rate, respectively. It also achieves the lowest sMAPE (68.36), indicating superior precision in numerical estimation. These results suggest that reinforcement learning not only improves the model's ability to align with spatial reasoning tasks but also makes the thinking prompt more adaptable to numerical estimation. Interestingly, we observe that adding the thinking prompt without reinforcement learning (Qwen2.5VL-3B) slightly degrades performance compared to the vanilla Qwen2.5VL-3B*, implying that prompt tuning alone may not benefit spatial reasoning unless coupled with proper optimization signals.

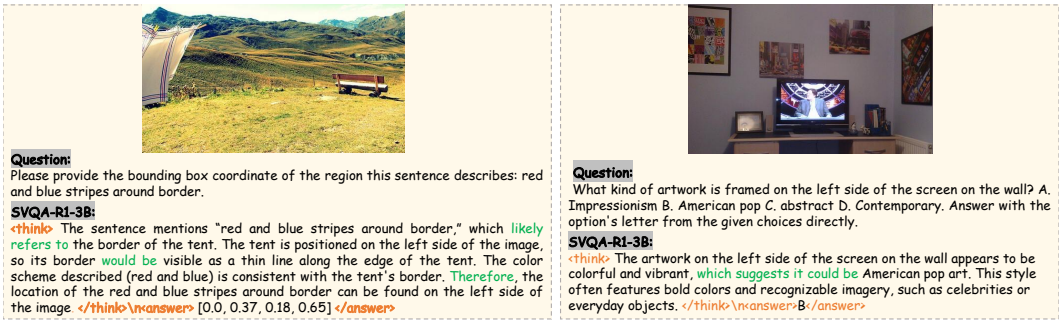
4.5 Evaluation on Open-ended Spatial VQA

In Figure 3, we present a comprehensive comparison of various models on the open-ended spatial VQA benchmarks, Vqasynth_Spacellava and OpenSpaces. Among all open-source models, our proposed SVQA-R1 consistently achieves the best overall performance across nearly all evaluation metrics, including both language-based (e.g., BLEU, sBERT) and spatial-reasoning-based (e.g., bounding box accuracy, yes/no, and distance) evaluations.

On Vqasynth_Spacellava, SVQA-R1 shows strong gains on the distance reasoning task, achieving the highest accuracy among all models. We attribute this improvement to our mirror-consistent training strategy, which encourages the model to produce stable spatial interpretations under horizontal flipping. Since spatial distances remain invariant under such transformations, this training constraint may help the model develop more robust representations of spatial magnitudes. Additionally, SVQA-R1 also shows notable improvements in bounding box prediction (Acc.@0.75 and mIoU), indicating enhanced understanding of object-centric spatial relations through our view-consistency framework.

On OpenSpaces, where bounding box annotations are unavailable, SVQA-R1 continues to outperform baselines, particularly in language-grounded metrics such as BLEU-1, BLEU-2, and sBERT. It also achieves the highest performance on both yes/no and distance reasoning, confirming that the benefits of our view-consistency strategies generalize across datasets.

We also observe that SpaceThinker-Qwen2.5-VL yields relatively low BLEU scores despite generating semantically plausible answers. Upon closer inspection, we find that this degradation is largely due to answer length/style mismatches: the model often produces minimalistic responses like "yes" or



(a) (d)

Figure 3: Visualization of model output.

"true," whereas the ground truth contains longer, more descriptive phrases. This discrepancy penalizes BLEU-1, measuring unigram overlap, even in cases where the model's output is correct in meaning.

4.6 Ablation Study

From Table 4, we observe that even using only the format reward in GRPO (comparison between (a) and (b)) leads to a significant improvement in performance, especially in Success Rate on Q-Spatial++, which increases by over 12%. We hypothesize that format guidance helps the model arrive at final answers more reliably by reinforcing output structure during the reasoning process. On Vqasynth_Spacellava, while the improvement in LLM accuracy is moderate, BLEU-1 improves dramatically (from 10.00% to 51.55%), suggesting that format reward strongly enhances syntactic alignment with the ground truth. In contrast, using only the semantic reward (comparison between (a) and (c)) improves LLM accuracy (from 45.33% to 60.34%) and also yields a large gain in Success Rate on Q-Spatial++. However, BLEU-1 drops to 0.00% and sBERT to 16.14%, indicating that the model outputs are semantically close but fail to match token-wise with reference answers. This is expected, as semantic reward emphasizes meaning-level similarity, while BLEU and sBERT are sensitive to surface form and token alignment. After incorporating spatial-GRPO (e), all metrics improve consistently. The sMAPE is significantly reduced, reflecting better numeric precision in quantity reasoning. Meanwhile, BLEU-1 and sBERT also achieve the highest scores, confirming that SVQA-R1 enhances semantic alignment but also supports more accurate numerical understanding.

4.7 Qualitative Analysis

Figures 8(a) to (d) are sampled from the Vqasynth_Spacellava dataset and evaluated using our **SVQA-R1** model with 3B parameters. Since Vqasynth_Spacellava is an open-ended spatial VQA dataset, the questions are diverse yet inherently spatial. For example, in Figure 8(b), the model is required to estimate the distance between people distributed across the scene, while in Figure 8(d), it needs to recognize which person is on the "left", both requiring spatial relationship reasoning.

From the results, we observe that the model tends to reason step-by-step, exhibiting logical progression rather than generating a flat caption. For instance, it often begins with a clarifying intent such as *"I'll need to ..."*, then introduces uncertainty-aware expressions like *"likely," "would be,"* or *"suggest it could be"* to hypothesize plausible answers. After outlining its reasoning steps, the model often concludes with a summary like *"Therefore, ..."*, indicating a final decision based on prior analysis. Please refer to the supplementary material for more visualization.

5 Conclusion and Limitation

In this work, we present a novel SVQA-R1 to enhance spatial reasoning capacity in MLLMs. Inspired by the success of the recent R1-style training recipe introduced by Deepseek-R1, we invent a novel view-consistent reward function to encourage the policy model to learn a view-consistent action space. We evaluate our approach on multiple benchmarks, and it exhibits promising results, compared to both closed-source and open-source models. After the self-exploration via Spatial-GRPO in MLLMs, we observe interpretable and reasonable reasoning paths.closed-source and open-source models. Though the mirror flipping and 2D transformation efficiently augments the existing single-view

images, more advanced techniques such as Nerf and Gaussian Splatting for novel view synthesis are not investigated yet, where we leave them for future work.

REFERENCES

Spacethinker-qwen2.5vl-3b. https://github.com/remyxai/VQASynth, 2024.

Spacethinker-qwen2.5vl-3b. https://huggingface.co/remyxai/SpaceThinker-Qwen2.5VL-3B, 2025.

Gemini-1.5-flash. https://ai.google.dev/gemini-api/docs/models, 2025.

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

Yuxiao Bai, Yintong Zhang, Yuxuan Wu, et al. Qwen2.5 technical report. https://github.com/QwenLM/Qwen2, 2024. Accessed: 2025-08-01.

Boyuan Chen, Zhuo Xu, Sean Kirmani, Brain Ichter, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14455–14465, 2024a.

Hardy Chen, Haoqin Tu, Fali Wang, Hui Liu, Xianfeng Tang, Xinya Du, Yuyin Zhou, and Cihang Xie. Sft or rl? an early investigation into training r1-like reasoning large vision-language models. *arXiv preprint arXiv:2504.11468*, 2025.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024b.

Erik Daxberger, Nina Wenzel, David Griffiths, Haiming Gang, Justin Lazarow, Gefen Kohavi, Kai Kang, Marcin Eichner, Yinfei Yang, Afshin Dehghan, et al. Mm-spatial: Exploring 3d spatial understanding in multimodal llms. *arXiv preprint arXiv:2503.13111*, 2025.

Sivan Doveh, Assaf Arbelle, Sivan Harary, Roei Herzig, Donghyun Kim, Paola Cascante-Bonilla, Amit Alfassy, Rameswar Panda, Raja Giryes, Rogerio Feris, et al. Dense and aligned captions (dac) promote compositional reasoning in vl models. *Advances in Neural Information Processing Systems*, 36:76137–76150, 2023.

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

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.

Raisa Islam and Owana Marzia Moushi. Gpt-4o: The cutting-edge advancement in multimodal llm. *Authorea Preprints*, 2024.

Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. Mdetr-modulated detection for end-to-end multi-modal understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1780–1790, 2021.

Yifei Li, Yiran Chen, Yuxuan Liu, et al. Tinyllava-video-r1: Enhancing video reasoning in small vision-language models via reinforcement learning. *arXiv preprint arXiv:2504.09641*, 2024.

Yuan-Hong Liao, Rafid Mahmood, Sanja Fidler, and David Acuna. Reasoning paths with reference objects elicit quantitative spatial reasoning in large vision-language models. *arXiv preprint arXiv:2409.09788*, 2024.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

Chenyang Ma, Kai Lu, Ta-Ying Cheng, Niki Trigoni, and Andrew Markham. Spatialpin: Enhancing spatial reasoning capabilities of vision-language models through prompting and interacting 3d priors. *Advances in neural information processing systems*, 2024.

OpenAI. Gpt-4o technical report. `https://openai.com/index/gpt-4o`, 2024. Accessed: 2025-05-14.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311–318, 2002.

Maitreya Patel, Naga Sai Abhiram Kusumba, Sheng Cheng, Changhoon Kim, Tejas Gokhale, Chitta Baral, et al. Tripletclip: Improving compositional reasoning of clip via synthetic vision-language negatives. *Advances in neural information processing systems*, 37:32731–32760, 2024.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, pp. 3982–3992, 2019.

Sasha Salter, Dushyant Rao, Markus Wulfmeier, Raia Hadsell, and Ingmar Posner. Attention-privileged reinforcement learning. In *Conference on Robot Learning*, pp. 394–408. PMLR, 2021.

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

Jiaqi Wang, Haoxuan Zhang, Linxi Xie, et al. Raven: Reasoning with visual commonsense for planning in human-robot interaction. *arXiv preprint arXiv:2206.07281*, 2022.

Jiayu Wang, Yifei Ming, Zhenmei Shi, Vibhav Vineet, Xin Wang, Sharon Li, and Neel Joshi. Is a picture worth a thousand words? delving into spatial reasoning for vision language models. *Advances in Neural Information Processing Systems*, 37:75392–75421, 2024.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Haoran Wu, Bohan Zhou, Yijie Wang, et al. Reason-rft: Reasoning-aware reinforcement fine-tuning for vision-language models. *arXiv preprint arXiv:2503.20752*, 2024a.

Haoran Wu, Bohan Zhou, Yijie Wang, et al. Visual-rft: Visual reinforcement fine-tuning for large vision-language models. *arXiv preprint arXiv:2503.01785*, 2024b.

Qinghao Xu, Yijie Wåang, Bohan Zhou, et al. Video-r1: Reinforcing video reasoning in multimodal large language models. *arXiv preprint arXiv:2503.21776*, 2024.

Kexin Yi, Jiajun Wu, Chuang Gan, Antonio Torralba, Pushmeet Kohli, and Josh Tenenbaum. Neural-symbolic vqa: Disentangling reasoning from vision and language understanding. *Advances in neural information processing systems*, 31, 2018.

Jihao Zhang, Yichong Zhou, Yifan Li, et al. Videochat-r1: Reinforcing spatio-temporal perception in video multimodal large language models. *arXiv preprint arXiv:2504.06958*, 2024.

Wayne Zhao, Shuai Zheng, Han Zhang, et al. Cot-vlm: Chain-of-thought prompting for visual language models. *arXiv preprint arXiv:2309.04761*, 2023.

Bohan Zhou, Yijie Wang, Hongxin Liu, et al. Vlm-r1: Towards a stable and generalizable r1-style large vision-language model. *arXiv preprint arXiv:2504.07615*, 2024.

A EXAMPLES OF HORIZONTAL IMAGE FLIPPING

The prompt in Table 5 is used to perform horizontal image flipping. The red-highlighted text indicates the modified sections designed to validate the logical consistency of the rewritten questions and answers. Figure 4 presents an example VQA pair before and after applying the verification prompt. As shown, the verification prompt corrects the logic of the question and answer, resulting in a more accurate training dataset.

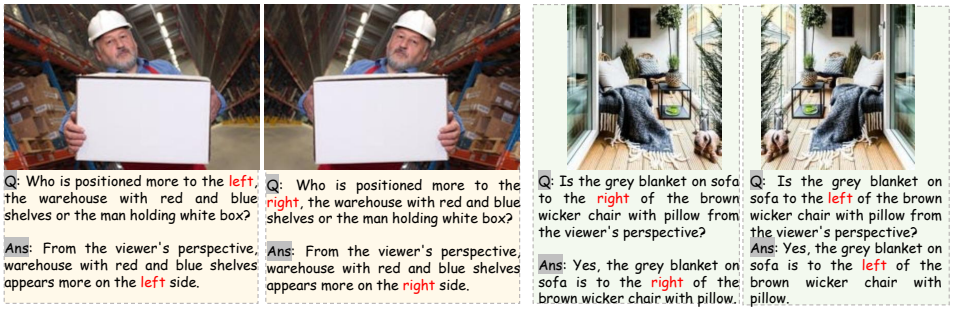
Table 5: Prompt used for horizontal image flipping.

You are a spatial reasoning assistant. You are given a question and its corresponding answer based on an image. Now assume that the image has been horizontally flipped. Your task is to rewrite the question and answer so that they remain logically correct for the flipped image. Write them **as if the flipped image was the original**, and **do not mention the flip** in your output. Also, verify the correctness of the left/right spatial relationship in the original answer. If the rewritten answer is inconsistent with the horizontal flip (i.e., the object that was on the left is still on the left), you must fix it. If you find an error, correct the object-direction mapping accordingly. Original question: {question}. Original answer: {answer}. Return your output as a **valid JSON object**, and nothing else.

Table 6: Instruction prompt for the flipping data. The placeholders {question} and {answer} will be replaced with specific content. Text in red highlights the verification and correction instructions.

(a) Question and Answer for Original Image (b) Question and Answer for Flipped Image

Figure 4: (a) The original image, question, and answer of a sample. (b) The flipped image, question, and the answers before and after verification enhancement.

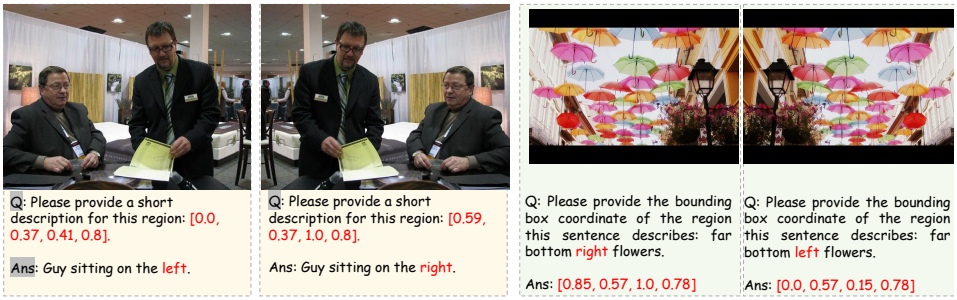


(a) Original VS. Flipped image and QA (b) Original VS. Flipped image and QA

Figure 5: Visualization of original and flipped image and QA for left-right spatial reasoning.

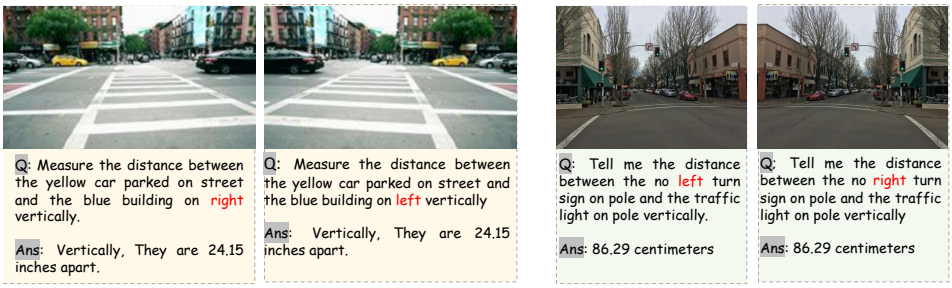
A.1 LEFT-RIGHT SPATIAL REASONING

This category highlights cases where the model's answer depends on reasoning about horizontal spatial relationships, such as determining whether an object is on the left or right. Image flipping directly alters this spatial context, and an accurate model should adapt its response accordingly. These examples in Figure 5 illustrate the model's ability to reverse its reasoning when the visual orientation changes.



(a) Original VS. Flipped image and QA (b) Original VS. Flipped image and QA

Figure 6: Visualization of original and flipped image and QA for bounding box consistency.



(a) Original VS. Flipped image and QA (b) Original VS. Flipped image and QA

Figure 7: Visualization of original and flipped image and QA for unaffacted cases.

A.2 Bounding Box Consistency

In this section, we examine the consistency of predicted bounding boxes under image flipping. A spatially grounded model should correctly adjust the bounding box coordinates to reflect the flipped image layout. We present examples in Figure 6, revealing the model's sensitivity to spatial transformations.

A.3 Cases Unaffected by Flipping

Some questions are semantically or spatially invariant to horizontal flipping. For example, questions about object attributes, counts, or global scene understanding often yield the same answer regardless of image orientation. This section showcases such cases in Figure 7, confirming that the model maintains output stability when the visual change does not affect semantic interpretation.

B Determining the Task Type

In order to evaluate the model's performance more accurately across different types of visual reasoning, we categorize each sample into one of several task types: bounding box prediction, binary (yes/no) classification, or distance estimation. Note that not all examples could be confidently assigned to a specific task type. We exclude ambiguous or unsupported cases from type-specific evaluation. This classification enables us to apply type-specific evaluation metrics and better understand the model's behavior under each reasoning requirement.

In the following subsections, we detail the criteria used to assign a task type to each instance based on its question structure, answer format, or annotated metadata.

	Q-Spatial++			Vqasynth_Spacellava		
η	Success Rate (%) ↑	Samples Completed (%) ↑	sMAPE (%) ↓	llm (%) ↑	BLEU-1 (%) ↑	sBERT ↑
0	52.32	100.00	73.24	59.82	53.94	81.74
1	58.42	100.00	68.36	60.77	63.33	87.84
2	51.49	100.00	77.66	61.47	63.34	87.86
10	47.52	100.00	80.16	57.17	62.34	85.48

Table 7: Performance under different weights applied to the reward difference between original and flipped images.

	Q-Spatial++			Vqasynth_Spacellava		
Model	Success Rate (%) ↑	Samples Completed (%) ↑	sMAPE (%) ↓	llm (%) ↑	BLEU-1 (%) ↑	sBERT ↑
Original	58.42	100.00	68.36	60.77	63.33	87.84
VLAA	53.49	100.00	76.10	61.35	56.83	83.41

Table 8: Performance using different base model initializations.

B.1 Bounding Box

For bounding box-related questions, we identify them by checking whether the question text contains indicative spatial phrases. Specifically, we use the following keyword list:

- **BBOX_KEYWORDS:** `bounding box, box coordinates, coordinates, bbox, where is, x coordinate, y coordinate, draw a box, top left, bottom right, region of.`

Using this rule-based classification, we identify **293 samples** (accounting for **9.4%** of the dataset) as bounding box prediction tasks.

B.2 Yes/No

We define alias sets to identify binary (Yes/No) questions and answers, as shown below:

- **YES_ALIASES:** `yes, it is, appears to be, looks like, seems like, definitely, likely, indeed.`
- **NO_ALIASES:** `no, not, doesn't, isn't, unlikely, I don't think, probably not.`

Based on these expressions, we classify **1,742 samples** (accounting for **34.84%** of the dataset) as Yes/No questions. Since the model outputs for binary questions are often expressed in natural language rather than strictly as "yes" or "no," we use GPT-4o to assess whether the predicted answer is semantically consistent with the ground truth.

The evaluation is performed via a simple prompt-based classification. For each prediction, we provide GPT-4o with the following instruction, and the `{pred}` and `{gt}` will be replaced by the specific content:

```
You are an evaluator. Given a Yes/No question, a
ground-truth answer, and a predicted answer, determine
whether the predicted answer means the same as the
ground-truth answer.
Predicted answer: {pred}, ground-truth answer: {gt}.
Output only 0 or 1, where 0 indicates incorrect and 1
indicates correct.
```

B.3 Distance Prediction

To identify whether a question requires distance estimation, we rely on a set of indicative phrases commonly associated with spatial measurement. The following keywords are used for matching:

	Metrics			In Range		
Model	Success Rate (%) ↑	Samples Completed (%) ↑	sMAPE (%) ↓	50-100 (%) ↑	100-150 (%) ↑	150-200 (%) ↑
(a)	58.42	100.00	68.36	18.81	18.81	21.78
(b)	20.98	100.00	184.64	5.62	7.96	10.85

Table 9: Performance using different prompts for quantity dataset Q-Spatial++.

Prompt Type	Description
Original	{Question} First output the thinking process in <think> </think> tags and then output the final answer in <answer> </answer> tags.
Spatial Reasoning Prompt	Question: { Question } Use the following 4 steps sequentially to answer the question: Step 1 **Analyze the question** Step 2 **Identify up to 10 reference scales in the image, ranging from large to small sizes, and list them in the specified format** - A reference scale must be typical in size. - A reference scale can be the dimensions of an object or an object part. - A reference scale must NOT be floor tiles or floor planks. - Formulate the reference scales using the format: """The [choose from front-to-back, side-to-side, left-to-right, diameter, height (top to bottom edge), or mounting height (bottom edge to floor)] of [object or object part] is approximately [dimension estimate].""" Step 3 **Propose a robust step-by-step plan to answer the question by using the reference scales in Step 2** - A robust step-by-step plan performs the estimation in a coarse-to-fine manner. - First, use a reliable and large-sized reference scale as the primary reference for estimation. - Then, gradually use a reliable and smaller-sized reference scale for adjustment. - Repeat until the estimation is precise enough. - When performing visual comparison, be aware of perspective distortion. - Do NOT rely on pixel measurements from the images. Step 4 **Focus on the image and follow the plan in Step 3 to answer the question**

Table 10: Example of different prompts.

- **DISTANCE_KEYWORDS:** `how far, distance between, distance from, which is closer, which is farther, closer, further, nearer, farthest, measure the distance, what is the distance, spacing between, gap between.`

If any of these phrases appear in the question text (matched in a case-insensitive manner), we classify the corresponding sample as a distance prediction task. This classification enables the use of appropriate numerical evaluation metrics, such as absolute error or symmetric Mean Absolute Percentage Error (sMAPE). Based on this criterion, we identify **186 samples** (accounting for **6%** of the dataset) as distance-related questions.

C Different initialization of the MLLMs

In Table 8, we investigate the impact of different base model initializations on downstream performance. By initializing the model from various pretrained checkpoints, we aim to assess how the starting point influences training dynamics and final accuracy. It turns out that the officially original checkpoint performs better on both datasets.

D Differnt inference prompt

We further experiment with different prompting strategies to encourage step-by-step reasoning in the model's responses. Specifically, we compare the original prompt (a) with variants such as spatial

	Q-Spatial++			Vqasynth_Spacellava		
Step	Success Rate (%) ↑	Samples Completed (%) ↑	sMAPE (%) ↓	llm (%) ↑	BLEU-1 (%) ↑	sBERT ↑
500	43.56	100.00	84.35	60.06	52.95	84.74
1000	50.50	100.00	79.32	54.25	55.36	86.51
1500	44.55	100.00	79.31	52.13	54.29	86.23
2000	42.57	100.00	78.57	52.37	53.38	85.64

Table 11: Performance in different training steps.

reasoning prompts (b) and others tailored to the task context, as summarized in Table 10, and the results are in Table 9.

We find that although the new prompt encourages the model to produce more reasoning steps, it results in worse performance on the distance estimation task. We hypothesize that this may be due to the fact that the model was originally fine-tuned using the original prompt format, which may not align well with the new prompt structure. This mismatch could lead to degraded performance when adapting to the modified prompting style.

E RESCALING TRAINING

We further analyze intermediate checkpoints during training to observe the progression of model performance and reasoning quality. Interestingly, we find that performance initially improves with more training steps but begins to decline after a certain point. This suggests that more training does not always lead to better performance, and there may exist an optimal checkpoint that balances learning and overfitting.

As shown in Table 11, we compare selected checkpoints and observe that the model at step 1000 achieves the best performance among them. We believe that evaluating more fine-grained checkpoints could potentially reveal an even better-performing model.

F INFERRED WORDS IN CHAIN-OF-THOUGHT REASONING

To better understand the nature and quality of the model's reasoning, we conduct a lexical analysis of the generated Chain-of-Thought (CoT) responses. In particular, we examine whether the model produces reasoning-oriented expressions, such as causal connectors, speculative language, and sequential markers, that are commonly associated with multi-step inference.

We define a set of indicative CoT-related keywords spanning various reasoning functions (e.g., uncertainty, causality, step-by-step logic). By counting their occurrences in the generated responses, we aim to quantify the model's usage of explicit reasoning cues and gain insight into the linguistic characteristics of its CoT outputs.

To quantify the presence of reasoning patterns in model outputs, we define a set of indicative keywords commonly associated with speculative language, causal reasoning, and step-by-step inference:

- **COT_KEYWORDS:** `likely, probably, possibly, maybe, might be, could be, seems, appears to, I think, I guess, I'm not sure, because, since, therefore, thus, so, hence, as a result, that means, which implies, accordingly, first, next, then, finally, in the first step, in the second step, after that, subsequently, clearly, obviously, evidently, definitely, in fact, it is important to note, if, suppose, assuming that, in case, let's say, consider that.`

Based on this keyword list, we observe that the average number of reasoning-related words per response is **0.8**, indicating that most responses contain at least one token suggestive of inferred or stepwise reasoning.

G THE SCALING COEFFICIENT η FOR THE REWARD DIFFERENCE

In Table 7, we also explore the effect of applying different scaling coefficients to the reward difference during reward correction. By adjusting the scaling factor of the reward difference, we aim to control the impact of relative feedback and examine its influence on model training and convergence.

We observe that assigning a larger weight to the reward difference improves the performance of the distance estimation task, as reflected by a higher sMAPE score. This indicates that the model produces more accurate numerical predictions. However, we also notice a decrease in the success rate. We hypothesize that while emphasizing the reward gap helps the model focus more on the exact value, it may simultaneously reduce its confidence or stability, leading to uncertainty in determining whether the output is sufficiently close to the ground truth.

In the case of the VQASynth_Spacellava dataset, we find that increasing the reward weight (comparing weight 1 vs. weight 2) results in slightly better BLEU and Sentence-BERT scores, but a lower LLM-based evaluation score. We conjecture that as the weight increases, the model becomes more sensitive to numerical precision or structured form, but tends to overlook semantic fidelity, which affects alignment with LLM-based judgments.

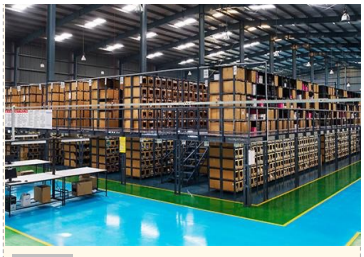
H MORE VISUALIZATION EXAMPLES

We show more visualizations in Fig. 8 to qualitatively analyze the behavior of our model on diverse spatial reasoning scenarios. These examples cover a wide range of spatial relations, including left/right comparison, relative size judgment, object localization, vertical ordering, and approximate distance estimation. As illustrated, the model is able to generate spatially grounded reasoning chains and produce consistent answers across different spatial query types. Notably, even for challenging cases that require relational comparison or coarse metric estimation, the model demonstrates stable reasoning patterns instead of relying on dataset bias or superficial cues, highlighting the effectiveness of our spatially guided training strategy.

Question:
Which kind of device is to the left of the screen?

SVQA-R1-3B:
<think> The problem involves identifying the type of device to the left of the screen. The screen is described as a monitor or computer screen, which is typically a type of display device.
1. The device to the left of the screen is a laptop or computer, as it has a keyboard in front of it.
2. Laptops often serve as display devices, and their monitors are often referred to as screens.
3. Therefore, the device to the left of the screen is a laptop.
</think><answer> laptop </answer>

(a)



Question:
Is the empty metal shelving unit smaller than the metal shelving with boxes?

SVQA-R1-3B :
<think> Comparing the empty metal shelving unit to the one with boxes, it can be observed that the empty shelving unit occupies a fraction of the space, while the shelving unit with boxes occupies a larger portion of the floor space. Therefore, it's evident that the empty metal shelving unit is smaller than the metal shelving with boxes. </think>\n<answer> Yes
</answer>

(b)

Question:
Which side of the image are the people on?

SVQA-R1-3B :
<think> The people are clearly visible in the image and are positioned in the middle ground. They appear to be on horseback, walking up a mountain trail. The direction of movement is towards the mountains in the background.
</think>\n<answer> They are on the left side of the image. </answer>

(c)

Question:
Between the woman wearing gold earrings and the fruit display with apples and oranges, which one is positioned higher?

SVQA-R1-3B:
<think> The woman wearing gold earrings appears on the left side of the image, while the fruit display with apples and oranges is more towards the top of the image. The fruit display is higher up than the woman. </think><answer> fruit display with apples and oranges </answer>

(d)

Question:
Tell me the distance between the grey metal support beams with yellow shelving units and the yellow metal shelving units with cardboard boxes vertically.
SVQA-R1-3B:
<think> To calculate the distance between the grey metal support beams with yellow shelving units and the yellow metal shelving units with cardboard boxes vertically, we need to consider several factors:
1. The position of the grey metal support beams along the length of the shelving units.
2. The thickness of the yellow metal shelving units.
3. The vertical distance between the grey support beams and the next shelf.
Assuming there are no gaps between the shelves or supports, and taking into account the uniform structure of the shelving units:
- The distance between the grey metal support beams would be the width of the shelving units.
- The vertical distance would be the height from the grey support beam to the next shelf, which we can estimate based on the shelving unit's height.
Given that the image is an overall view and a precise distance is not clearly visible, we need to estimate the vertical distance. Typically, shelving units such as these are 3-4 feet high.
Let's estimate a conservative distance. For simplicity, we can assume the distance is approximately double the height of the shelves between the grey support beams.
Therefore, the distance between the support beams and the shelving units would be about twice the height of a typical shelving unit.
</think><answer> The distance would be approximately twice the height of a typical shelving unit </answer>

(e)

Figure 8: More visualization examples with thinking evidence.