Towards Spatial Trace with Reasoning in Vision-Language Models for Robotics

Enshen Zhou[1*‡], Yibo Li[1*], Jingkun An[1*], Jiayuan Zhang[1*], Shanyu Rong[2,3], Mengzhen Liu[2], Yi Han[1,3], Yuheng Ji[3,5], Huajie Tan[2,3], Jiawei He[3], Pengwei Wang[3], Zhongyuan Wang[3], Cheng Chi[3†], Lu Sheng[1,4†], Shanghang Zhang[2,3†]

[1]School of Software, Beihang University, [2] State Key Laboratory of Multimedia Information Processing, School of Computer Science, Peking University, [3] Beijing Academy of Artificial Intelligence, [4] Beijing Key Laboratory of Intelligent Creative Content Generation and Immersive Experience, [5] CASIA

zhouenshen@buaa.edu.cn, leeibo@buaa.edu.cn, anjingkun02@gmail.com
chicheng@baai.ac.cn, lsheng@buaa.edu.cn, shanghang@pku.edu.cn

Abstract. Spatial tracing, as a fundamental embodied interaction ability for robots, is inherently challenging as it requires multi-step metric-grounded reasoning compounded with complex spatial referring and real-world metric measurement. However, existing methods struggle with this compositional task. To this end, we propose RoboTracer, a 3D-aware VLM that first achieves both 3D spatial referring and measuring via a universal spatial encoder and a regression-supervised decoder to enhance scale awareness during supervised fine-tuning (SFT). Moreover, RoboTracer advances multi-step metric-grounded reasoning via reinforcement fine-tuning (RFT) with metric-sensitive process rewards, supervising key intermediate perceptual cues to accurately generate spatial traces. To support SFT and RFT training, we introduce TraceSpatial, a large-scale dataset of 30M QA pairs, spanning outdoor/indoor/tabletop scenes and supporting complex reasoning processes (up to 9 steps). We further present TraceSpatial-Bench, a challenging benchmark filling the gap to evaluate spatial tracing. Experimental results show that RoboTracer surpasses baselines in spatial understanding, measuring, and referring, with an average success rate of 79.1%, and also achieves SOTA performance on TraceSpatial-Bench by a large margin, exceeding Gemini-2.5-Pro by 36% accuracy. Notably, RoboTracer can be integrated with various control policies to execute long-horizon, dynamic tasks across diverse robots (UR5, G1 humanoid) in cluttered real-world scenes. Please see the project page at https://zhoues.github.io/RoboTracer.

Keywords: Spatial tracing · Spatial Reasoning · Vision-Language Model

1 Introduction

Embodied robots usually have to execute actions based on increasingly complex, spatially constrained instructions [1, 64, 66, 89], such as "*Water flowers from left*

* Equal contribution † Corresponding author ‡ Project leader

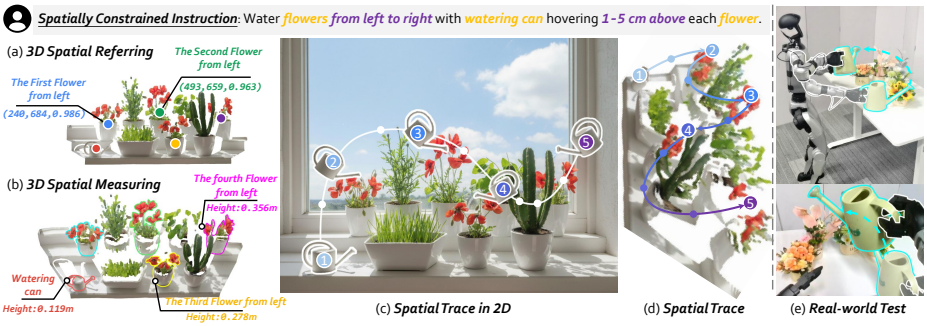

Fig. 1: Spatial tracing is pivotal for embodied robots to translate the spatially constrained instructions into 3D positional sequences (*i.e.*, spatial traces) in complex 3D scenes. This task demands **(a)** 3D spatial referring to resolve spatial relations and locate relevant objects involved in the trace, and **(b)** 3D spatial measuring to understand absolute, real-world metric quantities related to the trace. For example, **(a)** 3D positions of the watering can, and each flower pot are localized from left to right, and **(b)** their corresponding heights in meters are measured. By performing multi-step, metric-grounded reasoning over the key information above, the generated spatial trace can support not only **(c)** multi-step manipulation, but also **(d)** collision-free motion, thereby **(e)** enabling efficient control of diverse robots across tasks in cluttered scenes.

to right with watering can hovering 1–5 cm above each one" in Fig. 1, where recent data-scarce Vision-Language-Action (VLA) models fail to master. In this case, it would be beneficial to generate a 3D positional sequence, named as *spatial trace*, as an intuitive bridge to interpret the instruction following procedure in 3D space and guide the generation of actual action trajectories for robots. However, this surrogate task (*i.e.*, *spatial tracing*) is inherently challenging as it requires multi-step, metric-grounded reasoning in complex 3D scenes. To be specific, each reasoning step requires two key components: **(1)** *3D spatial referring* to resolve spatial relationships and accurately localize objects involved in the trace generation (*e.g.*, identifying flowers with their from left to right order and locating them). **(2)** *3D spatial measuring* to understand absolute, real-world metric quantities related to the trace in captured scene (*e.g.*, quantifying each flower's physical height and 1–5 cm height above each).

While recent Vision-Language Models (VLMs) [5, 42, 47] can perform 2D spatial reasoning [11, 18, 78, 89] and even 2D visual trace (*i.e.*, 2D positional sequence) generation [30, 38, 80], they overlook the multi-step nature of this task, particularly the crucial participation of intermediate objects involved in the trace, resulting in suboptimal generation. Moreover, their outputs are mainly in 2D space, lacking 3D space grounding and absolute metric understanding, creating a fundamental chasm between 2D visual trace and 3D *spatial trace*.

To this end, we propose *RoboTracer*, a 3D-aware reasoning VLM that not only acquires precise *3D spatial referring and measuring* via Supervised Fine-tuning (SFT), but also exhibits multi-step metric-grounded reasoning capabilities for *spatial tracing* via reinforcement fine-tuning (RFT). The core of our approach is to introduce a set of metric-sensitive reward functions (*e.g.*, refer-

ring, measuring, scale) during RFT. These rewards supervise the key perceptual objects involved in the trace and offer crucial intermediate evidence for accurate spatial trace generation. In addition, as *3D spatial referring and measuring* require better metric-grounded understanding, we introduce a scale decoder for VLM, supervised by a regression loss on the predicted metric scale factor to enhance metric perception, even from RGB inputs. Moreover, we incorporate a universal spatial encoder with the architectural flexibility to integrate diverse geometric configurations (*e.g.*, camera intrinsics, absolute depth) and further improve the precision of *spatial trace* generation.

To support *RoboTracer*'s training, we present *TraceSpatial*, a large-scale dataset with 4.5M high-quality examples and 30M QA pairs to first learn *3D spatial referring and measuring* in SFT, and further achieve *spatial tracing* with compositional reasoning on both in RFT. It spans diverse indoor, outdoor and tabletop scenes with fine-grained annotations (*e.g.*, precise geometry, object-level spatial referents) and contains a greatly higher proportion (48.2%) of absolute-scale data (14x prior [59]) for metric-grounded understanding. To advance metric-grounded multi-step reasoning capabilities, it provides step-wise annotations of the reasoning process (up to 9 steps). Moreover, it has object-centric and end-effector-centric spatial traces spanning 3 single-arm and dual-arm robot configurations, enhancing generalization and real-world applicability.

We evaluate *RoboTracer* on spatial understanding, measuring, and referring benchmarks, achieving SOTA average success rate of **79.1%**, exceeding Gemini-2.5-Pro by **11%**. It also outperforms all baselines on 2D visual trace benchmarks. To address the lack of benchmarks for spatial tracing, we introduce *TraceSpatial-Bench*, which contains 100 real-world images with manually annotated tasks involving object localization, movement, and placement. Each sample requires metric-grounded, multi-step reasoning (up to 8 steps), with precise start-point masks, end-point 3D bounding boxes, precise geometry annotation, and diverse metrics for fine-grained evaluation. *RoboTracer* still achieves best performance, surpassing Gemini-2.5-Pro by **36%**. Moreover, in Fig. 1 and Sec. 4.5, *RoboTracer* can execute long-horizon, dynamic tasks in cluttered real-world scenes, showing strong generalization across robots (*e.g.*, UR5, G1 humanoid).

Our contributions are summarized as follows: **(1)** We propose *RoboTracer*, a 3D-aware VLM that accepts arbitrary geometric inputs, uses scale supervision, guided by metric-sensitive rewards to achieve spatial tracing. **(2)** We construct *TraceSpatial*, a well-annotated dataset tailored for spatial tracing, enabling both SFT and RFT training, and *TraceSpatial-Bench*, a benchmark that fills the gap in evaluating it. **(3)** Extensive experiments show that *RoboTracer* generalizes well, surpasses baselines in spatial understanding, measuring, referring, tracing, and efficiently controls diverse robots across tasks in real world.

2 Related work

Spatial Reasoning with VLMs. Spatial reasoning refers to the ability to perceive and reason about 3D space, comprising metric-agnostic and -grounded

types. Metric-agnostic reasoning [11,18,58,74,89] captures object-centric properties (*e.g.*, position, orientation) and inter-object relations (*e.g.*, left/right, near/far), whereas metric-grounded reasoning involves precise, absolute-scale measurements [9,12,59] (*e.g.*, distance, depth, size) in the physical world. Compared to metric-agnostic one, metric-grounded reasoning requires better 3D understanding. Despite using 3D modalities, either explicitly [9, 12, 17, 51, 89] (*e.g.*, point clouds, depth maps) or implicitly [16, 25, 72, 86] (*e.g.*, VGGT [69]), existing methods still struggle to reason complex absolute-scale scenes for *3D spatial referring and measuring*. We thus propose a 3D-aware VLM that uses scale supervision and accepts arbitrary geometric inputs to address this gap.

Trace Generation with VLMs for Robotics. Trace enhances manipulation by capturing the spatio-temporal dynamics of objects. Recent advances in VLMs [2, 4, 5, 15, 29, 37, 53, 54, 60, 61, 65, 70, 90, 91] focus on predicting 2D visual traces (*i.e.*, 2D point sequences) and guiding robotic actions in two ways. **(1)** Lift-to-3D [30, 64, 73, 80]: Projects 2D traces into 3D spatial trace using depth maps and camera intrinsics. **(2)** Overlap-on-2D [36,38,87]: Renders the 2D trace onto the image by steering the control policies for action generation. However, 2D traces struggle to fully capture object dynamics in 3D space. Moreover, existing works for 2D visual trace struggle to handle complex spatially constrained instructions and supervise the key perception steps, also required by multi-step 3D spatial tracing, largely due to the lack of datasets. We thus propose a new dataset and benchmark tailored for spatial tracing.

Reinforcement Fine-tuning for VLMs. Reinforcement Fine-tuning (RFT) [7, 55, 56] is a post-training strategy that aligns models with human preferences or specific goals via feedback, complementing SFT [71, 88], which adapts pre-trained models using task-oriented data. Recent advances in VLMs use RFT to improve visual/spatial reasoning [3, 31, 62, 63, 76, 84], grounding [48, 57, 82, 89], segmentation [46], tool-use [27, 85], coding [83], safety [49, 50], and 2D visual trace prediction [28, 46, 80]. However, most approaches remain confined to 2D-relative perception and outcome-based reward, limiting their performance on spatial tracing tasks requiring 3D multi-step metric-grounded reasoning. Therefore, we design metric-sensitive process rewards to supervise key perception steps in the reasoning process to meet the above expectations.

3 Method

3.1 Problem Formulation and Representation Advantages

We formulate *spatial tracing* as predicting an ordered sequence of $6-12$ 3D sparse and discrete keypoints $\tau = \{p_t\}_{t=1}^{T}$ from visual inputs $\mathcal{O}$ (*e.g.*, RGB, RGB-D), optional camera geometry $\mathcal{G}$ (*e.g.*, intrinsics) and a textual instruction $\mathcal{T}$ via a VLM in a open-loop manner. Each point $p_t = (u_t, v_t, d_t)$ comprises a image-plane coordinate (u_t, v_t) and corresponding absolute depth d_t, encoding essential object-related and collision-critical waypoints. The resulting trace τ serves as a

spatial plan for entities (*e.g.*, a robot end-effector or object), and can be integrated with motion planning to predict physically feasible actions via adherence to the trace to follow the instruction. Notably, the instruction encodes both *3D spatial referring and measuring*, often requiring multi-step compositional reasoning. For example, in Fig. 2, to accomplish the task of "*Water flowers from left to right with watering can hovering 1-5 cm above each flower*", requires inferring the 3D positions and heights of all flowers in 3D scenes. These spatial cues provide crucial intermediate evidence for multi-step reasoning, enabling accurate trace generation under spatial constraints at the start, end, and along the path.

Instead of predicting 3D (x, y, z) coordinates directly, we adopt a decoupled (u, v, d) formulation that is trivially convertible to 3D via camera intrinsics. This circumvents the need for VLMs to implicitly learn camera geometry, thereby simplifying training and improving accuracy. Moreover, (u, v, d) points project into lower-dimensional spaces easily: omitting d yields 2D visual traces, and retaining only start/end points yields 3D or 2D spatial referring data. This formulation enhances data reusability, aligns seamlessly with existing 2D datasets [23,36,52] for co-training, thus bolstering multi-task learning performance.

Our spatial trace, combined with downstream motion planning for robotic control, offers the following advantages: **(1)** Compared to end-to-end VLAs [6, 8,36,39,40,45] that directly predict 6D dense actions, our approach can achieve superior zero-shot generalization without task-specific training, enabled by our generalizable formulation and less-constrained 3D space of this trace. **(2)** As an intermediate representation that delegates spatial reasoning and metric perception from instructions to spatial trace, this trace can handle more spatially and metrically constrained tasks compared to direct VLAs. **(3)** As a sequence of keypoints, spatial trace captures the intermediate execution process more effectively than start or end keypoint-based planning methods.

3.2 RoboTracer

VLM Architecture. Spatial tracing requires a metric-grounded 3D understanding for referring and measuring, yet simply fine-tuning existing VLMs encounters two hurdles: **(1)** insufficient absolute-scale supervision, especially with RGB-only data; **(2)** underuse of absolute-scale geometric cues (*e.g.*, camera intrinsics, depth) to enhance precision. In Fig. 2, we address these issues by introducing a scale decoder and a universal spatial encoder, each aligned with the LLM via projectors, akin to the existing RGB encoder. The scale decoder maps the `<SCALE>` token embedding into a numeric factor, linking scale-invariant representations to absolute metric scales. Rather than classification loss, we use regression loss to supervise it to heighten real-world 3D scale awareness. Moreover, we find that leveraging strong priors from a powerful feed-forward metric 3D geometry model [32] greatly enhances spatial and scale understanding. Building on this model, our universal spatial encoder flexibly integrates optional geometric cues (*e.g.*, camera intrinsics, poses, depth) to refine spatial representations as more geometry becomes available. This modular design enables: **(1)** Flexible

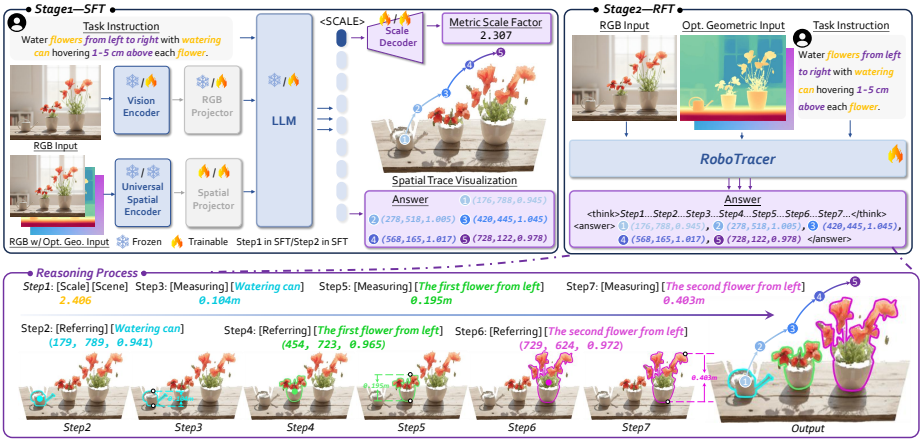

Fig. 2: Overview of RoboTracer. RoboTracer can process RGB images and task instructions, while flexibly integrating various geometric configurations when available to improve spatial precision (enabled by the integrated universal spatial encoder) with real-world scale awareness (powered by the scale decoder with regression loss). After SFT, metric-sensitive rewards in RFT further supervise the key perceptual steps in the trace and offer crucial intermediate evidence for accurate spatial trace generation.

training, leveraging diverse scale-aware annotations in datasets via flexible input augmentation to enrich spatial learning; **(2)** Geometry-adaptive inference, integrating available geometric cues without retraining or architectural changes, and more cues for better performance. See Supp. D.1 for details.

Supervised Fine-tuning. Since general VLMs' 2D-only pretraining limits 3D metric-grounded understanding, we propose a two-step SFT. **(1)** Metric Alignment. In Fig. 2, we align the spatial encoder and scale decoder with LLM by using geometric annotations (*e.g.*, depth, scale) from the *TraceSpatial* dataset (see Sec. 3.3). During this stage, only the projector and scale decoder are updated. **(2)** Metric Enhancement. We freeze the spatial encoder and finetune all other components on *TraceSpatial* and additional instruction-following datasets [43,44,77]. Crucially, we train on both RGB-only and RGB+$\mathcal{X}$ inputs, where $\mathcal{X}$ indicates arbitrary combinations of geometric annotations. This preserves image encoder's general VQA ability while flexibly accommodating diverse geometric configurations without retraining during inference. The SFT loss is defined as:

$$\mathcal{L} = \mathcal{L}_{ntp} + 0.1\| \log(\hat{s}) - \text{stopgrad}(\log(s^*)\|_2^2 \tag{1}$$

where $\mathcal{L}_{\text{ntp}}$ is next-token prediction loss, $\log(\hat{s})$ is predicted scale in logarithmic space, s^* is ground-truth scale. Moreover, *TraceSpatial* contains multi-step data with reasoning processes, providing a "cold start" for subsequent RFT stage. Please check Supp. D.2 for details about SFT training configuration.

Reinforcement Fine-tuning. We use RFT after SFT to improve compositional metric-grounded reasoning using GRPO [56] with multi-step reasoning data from *TraceSpatial*. We first define outcome-based rewards: **(1)** Out-

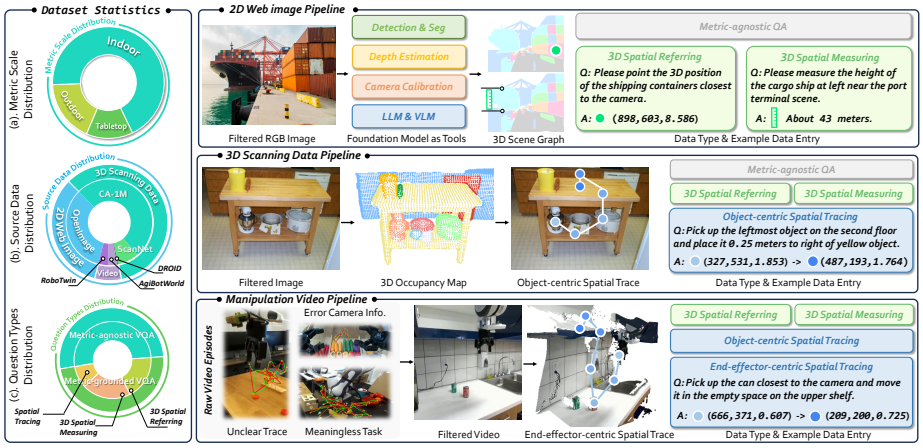

Fig. 3: *TraceSpatial* has 4.5M data samples from 2D, 3D, and video sources, covering outdoor, indoor, and tabletop scenes. It contains metric-agnostic/-grounded QA pairs for 3D spatial referring, measuring, tracing with a detailed reasoning process.

come Format Reward (R_{OF}) for structured outputs; **(2)** Point Reward (R_P) for start/end point consistency between the predicted trajectory (p_1, p_T) in $\hat{\tau} = \{\hat{p}_t\}_{t=1}^{T}$ vs. τ. Formally, $R_P = \frac{1}{2}\big[f(p_1, \hat{p}_1) + f(p_T, \hat{p}_T)\big], f(p, p') = \max\big(0,\ 1 - \|p - p'\|_2^2\big)$. **(3)** Trace Reward ($R_{\text{T}}$) for trajectory-level alignment using a distance metric $d(\tau, \hat{\tau})$: $R_{\text{T}} = \max\big(0,\ 1 - d(\tau, \hat{\tau})\big)$. All (u, v, d) values are normalized to $[0, 1]$, with depth scaled by the scene's maximum depth. However, the outcome-based rewards described above are metric-agnostic and provide no supervision over the key perceptual objects involved in trace generation (*e.g.*, *3D spatial measuring and referring*). To address this, we introduce metric-sensitive process rewards that leverage key-step perception annotations from *TraceSpatial*: **(1)** Process Format Reward (R_{PF}), enforcing the format "[Perception Type] [Target Object]:"; **(2)** Accuracy Reward (R_{Acc}), which applies to steps included in the key-step perception annotations. For each relevant step, we measure the prediction error using a specific metric, according to the perception type (*e.g.*, L1 distance for referring). Notably, this design is order-invariant, allowing flexible step ordering. The final reward sums both outcome-/process-based terms, with process-based terms scaled by 0.25. Fig. 2 shows that RFT-trained model generalizes well to 7-step spatial tracing, progressively resolving complex spatial relations and producing accurate trace. Please check Supp. D.3 for more details.

3.3 TraceSpatial Dataset

Overview. Key features are: **(1) Rich Diversity.** *TraceSpatial* spans outdoor, indoor, and tabletop scenes (Fig. 3 (a)) and includes both object-centric and end-effector-centric spatial traces; the latter captures gripper motions across 3 single-arm and dual-arm robot configurations, fostering model generalization in open-world scenarios. **(2) Multi-Dimensionality.** Beyond metric-agnostic

spatial concepts and relations, the dataset includes 48.2% metric-grouded QA (Fig. 3 (b)). These samples cover 3D spatial measuring and referring, and support multi-step spatial tracing by providing detailed annotations of reasoning process, addressing limitations in existing datasets [30, 80]. **(3) Large Scale.** With 4.5M samples and 30M QA pairs, *TraceSpatial* is the largest dataset for 3D spatial reasoning, supporting bottom-up spatial tracing learning. **(4) Fine-Grained.** Hierarchical object captions, from coarse categories (*e.g.*, "flower") to fine-grained spatial referents (*e.g.*, "the first flower from the left"), enable 3D spatial measuring, referring, and tracing in cluttered scenes. Absolute-scale geometry (*e.g.*, intrinsics, depth) further enriches spatial learning and flexible input augmentation. **(5) High Quality.** Rigorous filtering preserves spatial relevance. From 1.7M OpenImages [34], 466k images remain; from CA-1M [35](2M) and ScanNet [21](190k), 100k and 12k frames are retained based on text-identifiable 3D boxes, respectively. From DROID [33](116k) and AgiBot [20](167k) videos, 20k and 59k episodes are preserved after verifying valid camera poses, coherent tasks, and clean trajectories, respectively. We also manually inspect a subset to verify data quality and iteratively refined the dataset. **(6) Easy Scalability.** Our pipeline scalably integrates 2D images, 3D scans with bounding boxes, and manipulation videos. Even faced with RGB-only datasets in new domains, we can still scale up easily and acquire multi-step reasoning annotations automatically.

Data Recipe. In Fig. 3, we propose a data pipeline that progressively integrates 2D/3D/video sources for general VLMs to perform 3D spatial referring/measuring/tracing. **(1) 2D Web Images** aim to provide basic spatial concepts and broad-scale perception across indoor and outdoor scenes. We filter 1.7M images from OpenImage [34] down to 466K and employ VLM [5] with hierarchical region-captioning to produce fine-grained spatial references, surpassing previous approaches [22, 58]. Object captions serve as nodes for 3D scene graphs, where edges depict spatial relations inferred via object detection, depth, and camera estimation. Using template-/LLM-based methods, we generate metric-agnostic and 3D spatial QA grounded in these captions. **(2) 3D Scanning Datasets** want to arm the model with a focused metric-grounded spatial reasoning of indoor scenes. We thus leverage the richly annotated CA-1M [35] and ScanNet [21]. After fine-grained filtering, we construct 3D scene graphs with more diverse spatial relations, enabled by precise 3D bounding boxes compared to 2D approaches. Moreover, we generate 3D occupancy maps that encode positions, orientations, and metric distances (*e.g.*, "35cm right of the toy") for accurate object-centric spatial trace generation. **(3) Manipulation Videos** provide spatial traces aligned with the embodied manipulation in tabletop settings. While 3D scans enable object-centric tracing, they lack physically plausible manipulations. Hence, we curate real and simulated [14] tabletop videos (from 167k to 59k for AgiBot [20], and from 116k to 24k for DROID [33]) with calibrated cameras, accurate task execution, and clear trajectories. We further leverage VLM [5] to decompose these tasks into subgoals, enabling precise multi-step spatial tracing for single-/dual-arm across 3 robots. Notably, as 3D datasets and simulation videos are all-seeing, we construct multi-step, metric-grounded spatial tracing

Table 1: Performance on spatial understanding benchmarks. Rel., Dep., Dist. denote relation, depth, distance. Top-1/-2 success rate are indicated by **bold**/underlined text.

Method	Input	CV-Bench [68]			$BLINK_{val}$ [26]		RoboSpatial [58]	EmbSpatial [24]
		2D-Rel.	3D-Dep.	3D-Dist.	2D-Rel.	3D-Dep.		
GPT-4o [2]	RGB	84.62	86.50	83.33	82.52	78.23	77.20	63.38
Gemini-2.5-Pro [19]	RGB	93.54	91.00	90.67	**91.61**	87.90	77.24	76.67
NVILA-2B [47]	RGB	70.15	79.67	60.00	67.83	62.10	51.79	47.34
NVILA-8B [47]	RGB	91.54	91.83	90.67	76.92	76.61	59.35	67.72
Qwen-3-VL-4B [67]	RGB	92.31	94.67	87.50	87.71	85.48	79.67	77.01
Qwen-3-VL-8B [67]	RGB	93.85	94.50	90.33	87.41	85.48	77.24	77.86
Molmo 7B-D [23]	RGB	87.69	66.00	61.83	59.44	77.42	58.60	58.74
SpaceVLM-13B [11]	RGB	63.69	66.83	70.17	72.73	62.90	61.00	49.40
RoboBrain 2.0-7B [64]	RGB	96.00	94.83	90.00	79.72	85.48	74.80	74.78
SpatialBot-3B [9]	RGB-D	69.38	77.33	60.83	67.83	67.74	72.36	50.66
RoboTracer-2B-SFT	RGB	96.62	96.00	89.83	83.22	91.94	82.93	70.66
RoboTracer-8B-SFT	RGB	**97.08**	**97.17**	**93.50**	**91.61**	**92.74**	**83.74**	**81.75**

Table 2: Performance on spatial measuring benchmarks. S.N., S.E., R.E. denote Plus, Scannet, Scale Estimation, Refer Estimation. Top-1/-2 success rate (%) are indicated by **bold**/underlined text.

Method	Input	Q-spatial [41]		MSMU [13]	
		Plus	S.N.	S.E.	R.E.
GPT-4o [2]	RGB	31.68	37.06	3.86	2.09
Gemini-2.5-Pro [19]	RGB	56.44	70.00	64.86	48.42
NVILA-2B [47]	RGB	36.90	40.59	40.15	37.37
NVILA-8B [47]	RGB	46.87	44.71	33.98	38.95
Qwen-3-VL-4B [67]	RGB	53.47	70.00	63.71	42.11
Qwen-3-VL-8B [67]	RGB	29.70	56.47	63.32	52.63
Molmo 7B-D [23]	RGB	51.49	63.53	59.85	43.68
SpaceVLM-13B [11]	RGB	25.74	45.29	32.42	31.58
RoboBrain 2.0-7B	RGB	44.55	55.88	69.11	48.42
SpatialBot-3B [9]	RGB-D	20.79	29.41	16.60	22.62
RoboTracer-2B-SFT	RGB	68.32	70.59	78.38	60.00
RoboTracer-8B-SFT	RGB	**73.27**	**78.82**	**83.01**	**70.00**

Table 3: Performance on 2D spatial referring benchmarks. W.2.P., RoboS., RefS.-L. and RefS.-P. denote Where2Place [78], RoboSpatial [58], and Location and Placement parts of RefSpatial-Bench [89], respectively. Top-1/-2 success rate (%) are indicated by **bold**/underlined text.

Model	W.2.P.	RoboS.	RefS.-L.	RefS.-P.
Gemini-2.5-Pro [19]	61.90	40.20	46.96	24.21
Qwen3-VL-4B	66.0	63.11	43.00	**54.55**
Qwen3-VL-8B [67]	64.00	61.48	51.00	42.00
SpaceLLaVA [11]	11.8	16.0	5.82	4.31
RoboPoint [78]	46.80	41.30	22.87	9.27
Molmo-7B [23]	45.00	38.00	21.91	12.85
Molmo-72B [23]	63.80	40.90	45.77	14.74
RoboBrain 2.0-7B [64]	63.59	54.87	36.00	29.00
RoboTracer-2B-SFT	63.00	62.30	49.00	45.00
RoboTracer-8B-SFT	**69.00**	**66.40**	**55.00**	53.00

data, under the assumption that the generated code reflects optimal reasoning, with each line translated into textual form and intermediate results filled into structured formats (*e.g.*, coordinates, distances). See Supp. B for details.

4 Experiments

Model Configuration. We adopt NVILA [47] (2B/8B) as base model and apply SFT. Due to computational limits, we only perform RFT on 2B model. Since the model can accept arbitrary geometric inputs, it defaults to using only images when input types are unspecified. Please check Supp. D for details.
Robotic Evaluation Configuration. We ensure physical feasibility and smoothness by motion planning via adherence to the spatial trace, collision avoidance, and joint limits with a smoothness cost. We also correct VLM-generated spatial traces before use to satisfy physical constraints. Please see Supp. E.8 for details.

4.1 Spatial Understanding and Measuring

We evaluate our SFT model on public spatial understanding benchmarks, including CV-Bench [68], BLINK [26] (validation set), RoboSpatial [58] (configuration

Table 4: Performance on the 2D visual trace benchmarks. RMSE denotes Root Mean Square Error. Top-1/-2 scores are shown by **bold**/underlined text.

Model	ShareRobot-Bench [30]			VABench-V [79]		
	Discrete Fréchet Distance ↓	Hausdorff Distance ↓	RMSE ↓	Discrete Fréchet Distance ↓	Hausdorff Distance ↓	RMSE ↓
Qwen3-VL-4B [67]	0.3808	0.3294	0.2204	0.2792	0.2528	0.2037
Qwen3-VL-8B [67]	0.3922	0.3411	0.2328	0.2741	0.2549	0.2021
MolmoAct [36]	0.7764	0.7764	0.6771	0.8136	0.8136	0.6877
HAMSTER [38]	0.4365	0.3919	0.3554	0.2124	0.2045	0.1825
RoboBrain 2.0-3B [64]	0.1974	0.1853	0.1380	0.3642	0.2820	0.2445
RoboBrain 2.0-7B [64]	0.1669	0.1575	0.1250	0.3289	0.2604	0.2237
Embodied-R1-3B [81]	0.3426	0.3002	0.2388	0.3028	0.2588	0.2129
RoboTracer-2B-SFT	0.1605	0.1544	0.1114	0.1664	0.1551	0.1237
RoboTracer-8B-SFT	**0.1449**	**0.1384**	**0.0966**	**0.1494**	**0.1367**	**0.1065**

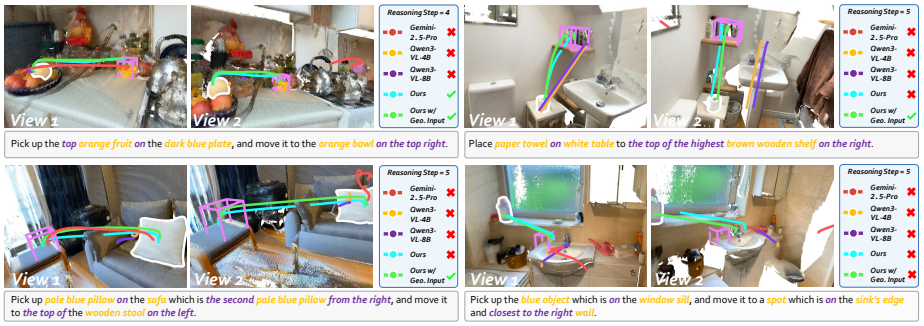

Fig. 4: *TraceSpatial-Bench* Results. White masks denote ground-truth 3D starting region; pink 3D boxes mark correct end regions. Despite similar 2D projections (left views of each case), our model yields more accurate spatial traces than strong general VLMs, which often produce floating or colliding traces due to inaccurate depth estimation. Leveraging richer geometric cues further improves performance.

subset), and EmbSpatial [24]. We also assess spatial measuring performance on Q-Spatial [41] and MSMU [13]. Please refer to Supp. E.2 and E.3 for details.

SFT learns strong spatial understanding and metric measuring. In Tab. 1 and Tab. 2, *RoboTracer*-8B-SFT trained solely on *TraceSpatial* surpasses all baselines with an average success rate of 85.7%, even outperforming Gemini-2.5-Pro by 8.58% and base model NVILA-8B by 20.3%. Notably, we find that its improvements are more pronounced in 3D-related and measurement tasks compared to 2D tasks (23.6% *vs.*14.7%), showing that our architectural design and curated dataset during SFT enhance the model's 3D spatial and scale awareness.

4.2 2D Spatial Referring and Visual Trace

We evaluate 2D spatial referring (Where2Place [78], RoboSpatial [58], RefSpatial-Bench [89]) and visual trace benchmarks (ShareRobot-Bench [30] (end-effector-centric), VABench-V [79]) (object-centric). See Supp. E.4 and E.5 for details.

Table 5: Performance on *TraceSpatial-Bench*. R., I., D. denote RGB image, intrinsics, absolute depth. N. means adding noise to depth input. We report success rate (SR).

Model	Input	Process Reward	2D Start SR (%) ↑	2D End SR (%) ↑	3D Start SR (%) ↑	3D End SR (%) ↑	Overall SR (%) ↑
Gemini-2.5-Pro [19]	RGB	-	31	33	9	16	3
Qwen3-VL-4B [67]	RGB	-	64	16	43	24	4
Qwen3-VL-8B [67]	RGB	-	60	21	47	22	6
MolmoAct [36]	RGB	-	4	15	-	-	-
Magma [75]	RGB	-	28	17	-	-	-
RoboBrain 2.0-7B [64]	RGB	-	46	23	-	-	-
Embodied-R1-3B [81]	RGB	-	63	13	-	-	-
RoboRefer-2B (finetuned)	R.D.	-	48	43	60	51	28
LEO-7B (finetuned)	PointCloud	-	46	39	58	48	27
RoboTracer-2B-SFT	RGB	-	56	44	63	52	31
RoboTracer-2B-SFT	R.I.D.	-	59	45	75	56	38
RoboTracer-RFT	RGB	✗	61	46	64	54	33
RoboTracer-RFT	RGB	✓	63	47	73	55	39
RoboTracer-RFT	R.I.	✓	64	48	73	56	40
RoboTracer-RFT (2D pipeline)	R.I.D.	✓	66	51	75	58	42
RoboTracer-RFT	R.I.D. w/ N.	✓	68	53	76	59	44
RoboTracer-RFT	R.I.D.	✓	**69**	**53**	**78**	**61**	**45**

Decoupled formulation makes multi-task learning easy. Our *RoboTracer* achieves best performance on both 2D spatial referring (see Tab. 3) and visual trace (see Tab. 4) benchmarks. While *TraceSpatial* focuses on 3D metric-grounded data without explicitly targeting 2D tasks, our decoupled point formulation enables seamless projection of *TraceSpatial* into the data formats required for such 2D tasks (see Sec. 3.1). Moreover, in Tab. 9 (ID G & H), modeling (u, v, d) surpasses direct 3D (x, y, z) modeling. We attribute this to **(1)** our dataset reuse via dimensionality reduction at various levels (*e.g.*, 3D to 2D, point sequences to individual points), **(2)** stronger alignment with existing 2D datasets for co-training, thereby improving multi-task learning performance.

4.3 Multi-Step Spatial Tracing

To evaluate complex multi-step, metric-grounded spatial tracing, we introduce *TraceSpatial-Bench*, a challenging real-world benchmark focusing on cluttered indoor/tabletop scenes. The dataset comprises 100 images with precise geometric annotations (*e.g.*, camera geometry, absolute depth, and 3D occupancy). Each sample requires 3˘8 reasoning steps, specifying both the manipulated object's start mask and its end 3D box. We assess performance in 2D and 3D; a trajectory succeeds if its start and end positions are correct and all intermediate paths remain collision-free. See Supp. E.6 for details. We present our analyses below.

RFT enables better multi-step metric-guided reasoning. In Tab. 5, *RoboTracer*-RFT outperforms all baselines on 3D metrics, exceeding Gemini-2.5-Pro by 36%. We find that while these VLMs perform well on 2D referring and tracing, they fall short in 3D spatial tracing due to their limited metric depth understanding, often producing traces that float or collide with objects. Moreover, *RoboTracer*-RFT also outperforms the baselines (*e.g.*, LEO, RoboRefer) with 3D

Table 6: Performance on general benchmarks. Top-1 scores are shown by **bold** text.

Model	MME_{test}	MMBench_{dev}	OK-VQA	POPE
NVILA-2B [47]	1547	**78.63**	64.9	81.96
RoboTracer-2B-SFT	**1751**	77.62	**65.22**	**82.52**

Table 7: Performance on RoboTwin 2.0 [14] hard tasks. We report the success rate (%) compared to end-to-end VLA and VLM-based models. Gray rows indicate unseen tasks not present in *TraceSpatial*. *RoboTracer* demonstrates strong generalization.

Task	End-to-End Policy					Vision-Language Model		Ours
	ACT	DP	DP3	RDT	π_0	Qwen3-VL-8B	Gemini-2.5-Pro	RoboTracer-2B
Place A2B Left	0	0	2	1	1	0	2	**84**
Move Playingcard Away	0	0	3	11	22	0	5	**94**
Click Alarmclock	4	5	14	12	11	0	0	**79**
Place Burger Fries	0	0	18	27	4	0	0	**99**
...								
Place Container Plate	1	0	1	17	45	0	0	**52**
Stack Blocks Two	0	0	0	2	1	0	0	**33**
Place Empty Cup	0	0	1	7	11	0	0	**85**
Place Object Stand	0	0	0	5	11	0	0	**38**
Seen Avg. Success Rate	0.9	0.4	3.7	6.3	6.6	0	0.7	**75.4**
Unseen Avg. Success Rate	0.1	0	0.6	4.7	11.6	0	0.3	**44.4**
Total Avg. Success Rate	0.6	0.2	2.5	5.7	8.6	0	0.5	**64.0**

input after being finetuned by *TraceSpatial*. This shows that *RoboTracer*-RFT leverages learned 3D spatial knowledge and compositional reasoning to generate more accurate spatial traces. Fig. 4 further shows complex spatial tracing cases from *TraceSpatial-Bench* with model comparisons.

Multi-step processes enable better reasoning with robustness. In Tab. 5 and Tab. 9 (ID D), we find that metric-grounded reasoning relies more on multi-step process RFT training than high-quality external 3D inputs (*e.g.*, depth map) from spatial encoder equipped in SFT. Specifically, RFT without spatial encoder, initialized from SFT, slightly underperforms RFT with spatial encoder but largely exceeds SFT. Moreover, RFT model remains robust under noisy geometric input, showing its robustness without the high-quality external features.

Accurate geometry refines metric reasoning. In Tab. 5, using more precise geometric cues greatly improves performance, yield up to 6% absolute gain. This suggests that explicit, readily available geometry in embodied settings can further improve metric reasoning, rather than relying solely on implicit learning of VLM to understand them via RGB-only input. Moreover, our model supports more geometry input to enhance itself without retraining, broadening its applicability. Fig. 4 shows the generated traces with different geometric inputs.

4.4 Public vision-language Benchmarks

Joint training preserves common-sense knowledge. In Tab. 6, our model performs on par with or slightly surpasses the base model. This benefit stems from our use of *TraceSpatial* for RGB and RGB+$\mathcal{X}$ (See Sec. 3.2) joint training, enriched by general visual instruction datasets.

Table 8: Real-world robot evaluation of spatially and metrically constrained tasks. MolmoAct is the end-to-end VLA baseline. RoboRefer is the keypoint-based baseline.

Spatially and Metrically Constrained Tasks	Success Rate(%) ↑ MolmoAct [36]	RoboRefer [89]	MolmoAct (finetuned)	**Ours**
Pick the rightmost hamburger and place it on the keyboard in front of the laptop without collisions	0.00	0.00	40.00	**60.00**
Water flowers from right to left with watering can hovering 1-5 cm above each flower.	0.00	0.00	10.00	**30.00**

Fig. 5: Real-World Evaluation. The blue trace denotes predicted spatial trace in 2D, and the blue dot marks current target. *RoboTracer* can generate spatial traces whose start and end points all satisfy spatial constraints in cluttered and dynamic scenes.

4.5 Simulator and Real-world Evaluation

Spatial trace offers superior zero-shot generalization. In Tab. 7, we evaluate our model on 19 Robotwin 2.0 [14] hard tasks involving clustered scenes. Among them, 12 tasks are seen and 7 are unseen in the *TraceSpatial* dataset. Unlike task-specific end-to-end VLA baselines trained on each task, our model, without any task-specific training, outperforms the best baseline by 32.8% in unseen tasks, showing strong generalization. Moreover, current powerful VLMs struggle to generate spatial traces, highlighting the importance of our model design and dataset contribution. Please check Supp. E.7 for details.

Spatial tracing is critical for spatially and metrically constrained tasks. In Tab. 8, only our method can handle long-horizon tasks requiring multi-step, metric-grounded spatial tracing in cluttered, dynamic scenes compared to VLA-based (*e.g.*, MolmoAct [36]) and keypoint-based (*e.g.*, RoboRefer [89]) method. These tasks demand precise identification and placement of objects under evolving spatial constraints, while continuously avoiding collisions. In Fig. 5, integrating *RoboTracer* with motion planning and Code-as-Monitor [92] enables rapid updates at 1.5 Hz. Thus, when the rightmost hamburger is moved, the robot adapts by grasping the newly rightmost one, reaching over the large doll and the computer screen to place it on the keyboard. Notably, this embodiment-agnostic spatial trace can also be executed by G1 humanoid, enabling even more complex, long-horizon tasks such as flower-watering (Fig. 1). See Supp. E.8 for details.

RoboTracer serves as a practical tool to generate action data. *RoboTracer* equipped with motion planning and Code-as-Monitor [92] can generate action data for spatially and metrically tasks. In Tab. 8, after finetuning with successful episode and employing generated spatial traces as auxiliary supervision, we improve MolmoAct's 3D reasoning and spatial performance (success rate raised from 0% to 25%), showing *RoboTracer*'s application for data generation.

Table 9: Ablation Study on data recipe, spatial encoder, scale supervision with regression (Reg.) and next-token prediction (N.T.P) loss, point formulation in Q-Spatial, ShareRobotBench(S.R.B.) and *TraceSpatial-Bench*(T.S.B). We report success rate (SR)/Discrete Fréchet Distance (DFD) of 2B model.

ID	Data Recipe			Spatial	Scale	Point	Model	Q-Spatial	S.R.B	T.S.B.
	2D	3D	Video	Encoder	Supervision	Formulation	Type	SR(%) ↑	DFD ↓	SR (%) ↑
A	✗	✓	✓	✓	Reg.	(u, v, d)	SFT	51.49	0.1723	27
B	✓	✗	✓	✓	Reg.	(u, v, d)	SFT	33.52	0.1809	19
C	✓	✓	✗	✓	Reg.	(u, v, d)	SFT	63.40	0.4376	24
D	✓	✓	✓	✗	Reg.	(u, v, d)	RFT	-	-	36
E	✓	✓	✓	✓	✗	(u, v, d)	SFT	53.47	0.1693	24
F	✓	✓	✓	✓	N.T.P.	(u, v, d)	SFT	57.43	0.1671	26
G	✓	✓	✓	✓	Reg.	(x, y, z)	SFT	68.32	0.2426	30
H	✓	✓	✓	✓	Reg.	(u, v, d)	SFT	**69.61**	**0.1605**	31
I	✓	✓	✓	✓	Reg.	(u, v, d)	RFT	-	-	**39**

4.6 Ablation Study

Data recipe is crucial for SFT training. Tab. 9 shows that combining 2D, 3D, and video data yields optimal performance. 2D/3D data offer critical scale supervision for both indoor and outdoor scenes; their absence degrades metric-grounded Q-spatial [41] accuracy. Video data enables gripper motion learning across diverse robot configurations, improving end-effector-centric performance on ShareRobotBench [30]. Moreover, 3D data and simulated videos increase spatial instruction diversity and metric-grounded reasoning, crucial for *TraceSpatial-Bench*. This three-way data synergy is thus key to SFT training.

Dataset construction pipeline has scalability. Our dataset pipeline scales easily to RGB-only datasets in new domains, enabling automatic multi-step reasoning annotations. We re-annotate *TraceSpatial* via 2D web image pipeline and a gripper detector [52] without ground-truth metadata. Tab. 5 shows comparable performance of newly RFT-trained model, demonstrating its scalability.

Universal spatial encoder improves 3D reasoning. We fine-tune NVILA-2B [47] on *TraceSpatial* without universal spatial encoder, followed by RFT. In Tab. 9 (ID D & I), we find that the spatial encoder enhances metric-grounded reasoning and greatly improves multi-step spatial tracing. This is mainly due to: **(1)** encoder's prior 3D knowledge that facilitates implicit 3D learning, **(2)** cumulative reasoning across steps, amplifying the utility of spatial cues.

Scale regression supervision boosts metric awareness. In Tab. 9 (ID E & F & H), we compare supervising the metric scale factor using regression loss, next-token prediction loss, and no supervision. The regression-based model performs best. While textual (next-token) supervision of offers slight gains, it remains inferior to regression. We observe that: **(1)** pure next-token prediction supervision—whether or not the scale factor is included as output—demands extensive data to improve scale awareness as shown in recent work [10]; **(2)** explicitly predicting the scale factor, particularly for RGB-only inputs (as in our

RGB/RGB+$\mathcal{X}$ mixed training), forces models to learn scale information without relying on auxiliary geometry, thereby bolstering capability.

Metric-sensitive reward advances the accuracy. In Tab. 5, integrating process rewards boosts overall success rate by 4% compared to purely outcome-based methods. Notably, using only these metric-agnostic outcome-based rewards yields smaller improvements on 3D metrics and overall success rates compared to the gains observed on 2D metrics, relative to SFT. This highlights that leveraging *TraceSpatial*'s key step annotations to supervise metric-grounded step-wise perception enables more accurate traces in complex spatial relations.

5 Conclusion

In this paper, we propose *RoboTracer*, a novel 3D-aware VLM that addresses spatial tracing via multi-step, metric-grounded reasoning on both 3D spatial referring and measuring. In detail, we empower the model to flexibly process arbitrary geometric inputs for precision, employ scale supervision to enhance scale awareness, and guide it with metric-sensitive rewards to improve its reasoning. We also present *TraceSpatial*, a large-scale well-designed dataset for SFT and RFT training, with *TraceSpatial-Bench*, a benchmark tailored to evaluate spatial tracing. Extensive experiments show the effectiveness of *RoboTracer* and highlight its potential for a broad range of robotic applications.

Limitation and Future Work. Despite the strong performance of our 3D spatial-trace prediction, we rely on motion planning to recover the 6D end-effector pose for robotic control, which is less effective for rotation-rich manipulation where orientation constraints are critical. Dense spatial traces may handle such tasks, which is a potential area for further research to explore.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (62132001, 62476011), the Beijing Natural Science Foundation (L252218, L252060), and the Fundamental Research Funds for the Central Universities.

References

1. Abdolmaleki, A., Abeyruwan, S., Ainslie, J., Alayrac, J.B., Arenas, M.G., Balakrishna, A., Batchelor, N., Bewley, A., Bingham, J., Bloesch, M., et al.: Gemini robotics 1.5: Pushing the frontier of generalist robots with advanced embodied reasoning, thinking, and motion transfer. arXiv preprint arXiv:2510.03342 (2025)
2. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv (2023)
3. An, J., Zhu, Y., Li, Z., Zhou, E., Feng, H., Huang, X., Chen, B., Shi, Y., Pan, C.: Agfsync: Leveraging ai-generated feedback for preference optimization in text-to-image generation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 1746–1754 (2025)

4. Anthropic, A.: The claude 3 model family: Opus, sonnet, haiku. Claude-3 Model Card **1**, 1 (2024)
5. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., et al.: Qwen2. 5-vl technical report. arXiv (2025)
6. Bai, S., Lyu, J., Zhou, W., Li, Z., Wang, D., Xing, L., Zhao, X., Wang, P., Wang, Z., Chi, C., et al.: Latent reasoning vla: Latent thinking and prediction for vision-language-action models. arXiv preprint arXiv:2602.01166 (2026)
7. Bai, Y., Jones, A., Ndousse, K., Askell, A., Chen, A., DasSarma, N., Drain, D., Fort, S., Ganguli, D., Henighan, T., et al.: Training a helpful and harmless assistant with reinforcement learning from human feedback. arXiv (2022)
8. Black, K., Brown, N., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., Groom, L., Hausman, K., Ichter, B., et al.: π_0: A vision-language-action flow model for general robot control. arXiv preprint arXiv:2410.24164 (2024)
9. Cai, W., Ponomarenko, I., Yuan, J., Li, X., Yang, W., Dong, H., Zhao, B.: Spatialbot: Precise spatial understanding with vision language models. ICRA (2025)
10. Cai, Z., Yeh, C.F., Xu, H., Liu, Z., Meyer, G., Lei, X., Zhao, C., Li, S.W., Chandra, V., Shi, Y.: Depthlm: Metric depth from vision language models. arXiv preprint arXiv:2509.25413 (2025)
11. Chen, B., Xu, Z., Kirmani, S., Ichter, B., Sadigh, D., Guibas, L., Xia, F.: 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 (2024)
12. Chen, P., Lou, Y., Cao, S., Guo, J., Fan, L., Wu, Y., Yang, L., Ma, L., Ye, J.: Sd-vlm: Spatial measuring and understanding with depth-encoded vision-language models. arXiv preprint arXiv:2509.17664 (2025)
13. Chen, P., Lou, Y., Cao, S., Guo, J., Fan, L., Wu, Y., Yang, L., Ma, L., Ye, J.: Sd-vlm: Spatial measuring and understanding with depth-encoded vision-language models (2025), https://arxiv.org/abs/2509.17664
14. Chen, T., Chen, Z., Chen, B., Cai, Z., Liu, Y., Li, Z., Liang, Q., Lin, X., Ge, Y., Gu, Z., et al.: Robotwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation. arXiv preprint arXiv:2506.18088 (2025)
15. Chen, Z., Shi, Z., Lu, X., He, L., Qian, S., Yin, Z., Ouyang, W., Shao, J., Qiao, Y., Lu, C., et al.: Rh20t-p: A primitive-level robotic dataset towards composable generalization agents. arXiv preprint arXiv:2403.19622 (2024)
16. Chen, Z., Zhang, M., Yu, X., Luo, X., Sun, M., Pan, Z., Feng, Y., Pei, P., Cai, X., Huang, R.: Think with 3d: Geometric imagination grounded spatial reasoning from limited views. arXiv preprint arXiv:2510.18632 (2025)
17. Cheng, A.C., Fu, Y., Chen, Y., Liu, Z., Li, X., Radhakrishnan, S., Han, S., Lu, Y., Kautz, J., Molchanov, P., et al.: 3d aware region prompted vision language model. arXiv preprint arXiv:2509.13317 (2025)
18. Cheng, A.C., Yin, H., Fu, Y., Guo, Q., Yang, R., Kautz, J., Wang, X., Liu, S.: Spatialrgpt: Grounded spatial reasoning in vision language models. NeurIPS (2024)
19. Comanici, G., Bieber, E., Schaekermann, M., Pasupat, I., Sachdeva, N., Dhillon, I., Blistein, M., Ram, O., Zhang, D., Rosen, E., et al.: Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. arXiv preprint arXiv:2507.06261 (2025)
20. contributors, A.W.C.: Agibot world colosseum. https://github.com/OpenDriveLab/AgiBot-World (2024)
21. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: CVPR (2017)

22. Daxberger, E., Wenzel, N., Griffiths, D., Gang, H., Lazarow, J., Kohavi, G., Kang, K., Eichner, M., Yang, Y., Dehghan, A., et al.: Mm-spatial: Exploring 3d spatial understanding in multimodal llms. arXiv (2025)
23. Deitke, M., Clark, C., Lee, S., Tripathi, R., Yang, Y., Park, J.S., Salehi, M., Muennighoff, N., Lo, K., Soldaini, L., et al.: Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 91–104 (2025)
24. Du, M., Wu, B., Li, Z., Huang, X.J., Wei, Z.: Embspatial-bench: Benchmarking spatial understanding for embodied tasks with large vision-language models. In: ACL (Volume 2: Short Papers) (2024)
25. Fan, Z., Zhang, J., Li, R., Zhang, J., Chen, R., Hu, H., Wang, K., Qu, H., Wang, D., Yan, Z., et al.: Vlm-3r: Vision-language models augmented with instruction-aligned 3d reconstruction. arXiv preprint arXiv:2505.20279 (2025)
26. Fu, X., Hu, Y., Li, B., Feng, Y., Wang, H., Lin, X., Roth, D., Smith, N.A., Ma, W.C., Krishna, R.: Blink: Multimodal large language models can see but not perceive. In: ECCV (2024)
27. Han, Y., Chi, C., Zhou, E., Rong, S., An, J., Wang, P., Wang, Z., Sheng, L., Zhang, S.: Tiger: Tool-integrated geometric reasoning in vision-language models for robotics. arXiv preprint arXiv:2510.07181 (2025)
28. Huang, C.P., Wu, Y.H., Chen, M.H., Wang, Y.C.F., Yang, F.E.: Thinkact: Vision-language-action reasoning via reinforced visual latent planning. arXiv preprint arXiv:2507.16815 (2025)
29. Ji, Y., Liu, Y., Tan, H., Huang, X., Huang, F., Xu, Y., Chi, C., Zhao, Y., Lyu, H., Co, P., et al.: Prm-as-a-judge: A dense evaluation paradigm for fine-grained robotic auditing. arXiv preprint arXiv:2603.21669 (2026)
30. Ji, Y., Tan, H., Shi, J., Hao, X., Zhang, Y., Zhang, H., Wang, P., Zhao, M., Mu, Y., An, P., et al.: Robobrain: A unified brain model for robotic manipulation from abstract to concrete. arXiv preprint arXiv:2502.21257 (2025)
31. Kang, L., Song, X., Zhou, H., Qin, Y., Yang, J., Liu, X., Torr, P., Bai, L., Yin, Z.: Viki-r: Coordinating embodied multi-agent cooperation via reinforcement learning. arXiv preprint arXiv:2506.09049 (2025)
32. Keetha, N., Müller, N., Schönberger, J., Porzi, L., Zhang, Y., Fischer, T., Knapitsch, A., Zauss, D., Weber, E., Antunes, N., et al.: Mapanything: Universal feed-forward metric 3d reconstruction. arXiv preprint arXiv:2509.13414 (2025)
33. Khazatsky, A., Pertsch, K., Nair, S., Balakrishna, A., Dasari, S., Karamcheti, S., Nasiriany, S., Srirama, M.K., Chen, L.Y., Ellis, K., et al.: Droid: A large-scale in-the-wild robot manipulation dataset. arXiv preprint arXiv:2403.12945 (2024)
34. Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., Kamali, S., Popov, S., Malloci, M., Kolesnikov, A., et al.: The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. IJCV (2020)
35. Lazarow, J., Griffiths, D., Kohavi, G., Crespo, F., Dehghan, A.: Cubify anything: Scaling indoor 3d object detection. arXiv preprint arXiv:2412.04458 (2024)
36. Lee, J., Duan, J., Fang, H., Deng, Y., Liu, S., Li, B., Fang, B., Zhang, J., Wang, Y.R., Lee, S., et al.: Molmoact: Action reasoning models that can reason in space. arXiv preprint arXiv:2508.07917 (2025)
37. Li, B., Zhang, Y., Guo, D., Zhang, R., Li, F., Zhang, H., Zhang, K., Zhang, P., Li, Y., Liu, Z., Li, C.: LLaVA-onevision: Easy visual task transfer. Transactions on Machine Learning Research (2025)

38. Li, Y., Deng, Y., Zhang, J., Jang, J., Memmel, M., Yu, R., Garrett, C.R., Ramos, F., Fox, D., Li, A., et al.: Hamster: Hierarchical action models for open-world robot manipulation. arXiv preprint arXiv:2502.05485 (2025)
39. Li, Z., Chi, C., Wei, Y., Zhu, B., Huang, T., Sun, Z., Peng, Y., Wang, P., Wang, Z., Liu, F., et al.: Do you have freestyle? expressive humanoid locomotion via audio control. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 956–965 (2026)
40. Li, Z., Chi, C., Zhu, B., Wei, Y., Bai, S., Ji, Y., Peng, Y., Huang, T., Wang, P., Wang, Z., et al.: Robomirror: Understand before you imitate for video to humanoid locomotion. arXiv preprint arXiv:2512.23649 (2025)
41. Liao, Y.H., Mahmood, R., Fidler, S., Acuna, D.: Reasoning paths with reference objects elicit quantitative spatial reasoning in large vision-language models (2024), https://arxiv.org/abs/2409.09788
42. Liu, F., Emerson, G., Collier, N.: Visual spatial reasoning. Transactions of the Association for Computational Linguistics (2023)
43. Liu, F., Lin, K., Li, L., Wang, J., Yacoob, Y., Wang, L.: Mitigating hallucination in large multi-modal models via robust instruction tuning. In: ICLR (2024)
44. Liu, H., Li, C., Li, Y., Lee, Y.J.: Improved baselines with visual instruction tuning. In: CVPR (2024)
45. Liu, M., Zhou, E., Chi, C., Han, Y., Rong, S., Chen, L., Wang, P., Wang, Z., Zhang, S.: Sapave: Towards active perception and manipulation in vision-language action models for robotics. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 37164–37174 (2026)
46. Liu, Y., Peng, B., Zhong, Z., Yue, Z., Lu, F., Yu, B., Jia, J.: Seg-zero: Reasoning-chain guided segmentation via cognitive reinforcement. arXiv (2025)
47. Liu, Z., Zhu, L., Shi, B., Zhang, Z., Lou, Y., Yang, S., Xi, H., Cao, S., Gu, Y., Li, D., et al.: Nvila: Efficient frontier visual language models. arXiv (2024)
48. Liu, Z., Sun, Z., Zang, Y., Dong, X., Cao, Y., Duan, H., Lin, D., Wang, J.: Visual-rft: Visual reinforcement fine-tuning. arXiv (2025)
49. Lu, X., Chen, Z., Hu, X., Zhou, Y., Zhang, W., Liu, D., Sheng, L., Shao, J.: Is-bench: Evaluating interactive safety of vlm-driven embodied agents in daily household tasks. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 40, pp. 35680–35688 (2026)
50. Lu, X., Zhou, Y., Chen, Z., Wang, R., Sima, B., Zhou, E., Sheng, L., Liu, D., Shao, J.: Homeguard: Vlm-based embodied safeguard for identifying contextual risk in household task. arXiv preprint arXiv:2603.14367 (2026)
51. Mao, Y., Zhong, J., Fang, C., Zheng, J., Tang, R., Zhu, H., Tan, P., Zhou, Z.: Spatiallm: Training large language models for structured indoor modeling. arXiv preprint arXiv:2506.07491 (2025)
52. Niu, D., Sharma, Y., Biamby, G., Quenum, J., Bai, Y., Shi, B., Darrell, T., Herzig, R.: Llarva: Vision-action instruction tuning enhances robot learning. arXiv preprint arXiv:2406.11815 (2024)
53. Qin, Y., Shi, Z., Yu, J., Wang, X., Zhou, E., Li, L., Yin, Z., Liu, X., Sheng, L., Shao, J., et al.: Worldsimbench: Towards video generation models as world simulators. arXiv preprint arXiv:2410.18072 (2024)
54. Qin, Y., Zhou, E., Liu, Q., Yin, Z., Sheng, L., Zhang, R., Qiao, Y., Shao, J.: Mp5: A multi-modal open-ended embodied system in minecraft via active perception. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16307–16316 (2024)

55. Rafailov, R., Sharma, A., Mitchell, E., Manning, C.D., Ermon, S., Finn, C.: Direct preference optimization: Your language model is secretly a reward model. NeurIPS (2023)
56. Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y., Wu, Y., et al.: Deepseekmath: Pushing the limits of mathematical reasoning in open language models. arXiv (2024)
57. Shen, H., Liu, P., Li, J., Fang, C., Ma, Y., Liao, J., Shen, Q., Zhang, Z., Zhao, K., Zhang, Q., et al.: Vlm-r1: A stable and generalizable r1-style large vision-language model. arXiv preprint arXiv:2504.07615 (2025)
58. Song, C.H., Blukis, V., Tremblay, J., Tyree, S., Su, Y., Birchfield, S.: Robospatial: Teaching spatial understanding to 2d and 3d vision-language models for robotics. CVPR (2025)
59. Sun, P., Lang, S., Wu, D., Ding, Y., Feng, K., Liu, H., Ye, Z., Liu, R., Liu, Y.H., Wang, J., et al.: Spacevista: All-scale visual spatial reasoning from mm to km. arXiv preprint arXiv:2510.09606 (2025)
60. Tan, H., Chi, C., Chen, X., Ji, Y., Zhao, Z., Hao, X., Lyu, Y., Cao, M., Zhao, J., Lyu, H., et al.: Roboos-next: A unified memory-based framework for lifelong, scalable, and robust multi-robot collaboration. arXiv preprint arXiv:2510.26536 (2025)
61. Tan, H., Hao, X., Chi, C., Lin, M., Lyu, Y., Cao, M., Liang, D., Chen, Z., Lyu, M., Peng, C., et al.: Roboos: A hierarchical embodied framework for cross-embodiment and multi-agent collaboration. arXiv preprint arXiv:2505.03673 (2025)
62. Tan, H., Ji, Y., Hao, X., Lin, M., Wang, P., Wang, Z., Zhang, S.: Reason-rft: Reinforcement fine-tuning for visual reasoning. arXiv (2025)
63. Tan, H., Zhou, E., Li, Z., Xu, Y., Ji, Y., Chen, X., Chi, C., Wang, P., Jia, H., Ao, Y., et al.: Robobrain 2.5: Depth in sight, time in mind. arXiv preprint arXiv:2601.14352 (2026)
64. Team, B.R., Cao, M., Tan, H., Ji, Y., Chen, X., Lin, M., Li, Z., Cao, Z., Wang, P., Zhou, E., et al.: Robobrain 2.0 technical report. arXiv preprint arXiv:2507.02029 (2025)
65. Team, G., Anil, R., Borgeaud, S., Alayrac, J.B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A.M., Hauth, A., Millican, K., et al.: Gemini: a family of highly capable multimodal models. arXiv (2023)
66. Team, G.R., Abeyruwan, S., Ainslie, J., Alayrac, J.B., Arenas, M.G., Armstrong, T., Balakrishna, A., Baruch, R., Bauza, M., Blokzijl, M., et al.: Gemini robotics: Bringing ai into the physical world. arXiv (2025)
67. Team, Q.: Qwen3 technical report (2025), https://arxiv.org/abs/2505.09388
68. Tong, P., Brown, E., Wu, P., Woo, S., IYER, A.J.V., Akula, S.C., Yang, S., Yang, J., Middepogu, M., Wang, Z., et al.: Cambrian-1: A fully open, vision-centric exploration of multimodal llms. NeurIPS (2024)
69. Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: Vggt: Visual geometry grounded transformer. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 5294–5306 (2025)
70. Wang, Y., Ji, Y., Liu, Y., Zhou, E., Yang, Z., Tian, Y., Qin, Z., Liu, Y., Tan, H., Chi, C., et al.: Towards cross-view point correspondence in vision-language models. arXiv preprint arXiv:2512.04686 (2025)
71. Wei, J., Bosma, M., Zhao, V.Y., Guu, K., Yu, A.W., Lester, B., Du, N., Dai, A.M., Le, Q.V.: Finetuned language models are zero-shot learners. arXiv (2021)
72. Wu, D., Liu, F., Hung, Y.H., Duan, Y.: Spatial-mllm: Boosting mllm capabilities in visual-based spatial intelligence. arXiv preprint arXiv:2505.23747 (2025)

73. Xu, R., Zhang, J., Guo, M., Wen, Y., Yang, H., Lin, M., Huang, J., Li, Z., Zhang, K., Wang, L., et al.: A0: An affordance-aware hierarchical model for general robotic manipulation. arXiv preprint arXiv:2504.12636 (2025)
74. Yang, G., Zhang, T., Hao, H., Wang, W., Liu, Y., Wang, D., Chen, G., Cai, Z., Chen, J., Su, W., et al.: Vlaser: Vision-language-action model with synergistic embodied reasoning. arXiv preprint arXiv:2510.11027 (2025)
75. Yang, J., Tan, R., Wu, Q., Zheng, R., Peng, B., Liang, Y., Gu, Y., Cai, M., Ye, S., Jang, J., et al.: Magma: A foundation model for multimodal ai agents. arXiv (2025)
76. Yang, Y., He, X., Pan, H., Jiang, X., Deng, Y., Yang, X., Lu, H., Yin, D., Rao, F., Zhu, M., et al.: R1-onevision: Advancing generalized multimodal reasoning through cross-modal formalization. arXiv (2025)
77. Yu, L., Poirson, P., Yang, S., Berg, A.C., Berg, T.L.: Modeling context in referring expressions. In: ECCV (2016)
78. Yuan, W., Duan, J., Blukis, V., Pumacay, W., Krishna, R., Murali, A., Mousavian, A., Fox, D.: Robopoint: A vision-language model for spatial affordance prediction for robotics. arXiv preprint arXiv:2406.10721 (2024)
79. Yuan, Y., Cui, H., Chen, Y., Dong, Z., Ni, F., Kou, L., Liu, J., Li, P., Zheng, Y., Hao, J.: From seeing to doing: Bridging reasoning and decision for robotic manipulation. arXiv preprint arXiv:2505.08548 (2025)
80. Yuan, Y., Cui, H., Huang, Y., Chen, Y., Ni, F., Dong, Z., Li, P., Zheng, Y., Hao, J.: Embodied-r1: Reinforced embodied reasoning for general robotic manipulation. arXiv preprint arXiv:2508.13998 (2025)
81. Yuan, Y., Cui, H., Huang, Y., Chen, Y., Ni, F., Dong, Z., Li, P., Zheng, Y., Hao, J.: Embodied-r1: Reinforced embodied reasoning for general robotic manipulation (2025), https://arxiv.org/abs/2508.13998
82. Zhan, Y., Zhu, Y., Zheng, S., Zhao, H., Yang, F., Tang, M., Wang, J.: Vision-r1: Evolving human-free alignment in large vision-language models via vision-guided reinforcement learning. arXiv (2025)
83. Zhang, J., Chen, K., Lu, Z., Zhou, E., Yu, Q., Zhang, J.: Prune4web: Dom tree pruning programming for web agent. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 40, pp. 34710–34718 (2026)
84. Zhang, J., Huang, J., Yao, H., Liu, S., Zhang, X., Lu, S., Tao, D.: R1-vl: Learning to reason with multimodal large language models via step-wise group relative policy optimization. arXiv (2025)
85. Zhang, Z., Wu, Y., Jia, L., Wang, Y., Zhang, Z., Li, Y., Ran, B., Zhang, F., Sun, Z., Yin, Z., et al.: Think3d: Thinking with space for spatial reasoning. arXiv preprint arXiv:2601.13029 (2026)
86. Zheng, D., Huang, S., Li, Y., Wang, L.: Learning from videos for 3d world: Enhancing mllms with 3d vision geometry priors. arXiv preprint arXiv:2505.24625 (2025)
87. Zheng, R., Liang, Y., Huang, S., Gao, J., Daumé III, H., Kolobov, A., Huang, F., Yang, J.: Tracevla: Visual trace prompting enhances spatial-temporal awareness for generalist robotic policies. arXiv preprint arXiv:2412.10345 (2024)
88. Zhou, C., Liu, P., Xu, P., Iyer, S., Sun, J., Mao, Y., Ma, X., Efrat, A., Yu, P., Yu, L., et al.: Lima: Less is more for alignment. NeurIPS (2023)
89. Zhou, E., An, J., Chi, C., Han, Y., Rong, S., Zhang, C., Wang, P., Wang, Z., Huang, T., Sheng, L., et al.: Roborefer: Towards spatial referring with reasoning in vision-language models for robotics. arXiv preprint arXiv:2506.04308 (2025)

90. Zhou, E., Qin, Y., Yin, Z., Huang, Y., Zhang, R., Sheng, L., Qiao, Y., Shao, J.: Minedreamer: Learning to follow instructions via chain-of-imagination for simulated-world control. arXiv preprint arXiv:2403.12037 (2024)
91. Zhou, E., Qin, Y., Yin, Z., Shi, Z., Huang, Y., Zhang, R., Sheng, L., Shao, J.: Chain-of-imagination for reliable instruction following in decision making. In: 2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 10010–10017. IEEE (2025)
92. Zhou, E., Su, Q., Chi, C., Zhang, Z., Wang, Z., Huang, T., Sheng, L., Wang, H.: Code-as-monitor: Constraint-aware visual programming for reactive and proactive robotic failure detection. arXiv preprint arXiv:2412.04455 (2024)