ROBOAPPROACHBENCH: DIAGNOSING SPATIAL UNDERSTANDING DURING REAL-ROBOT APPROACH

Anonymous authors
Paper under double-blind review

ABSTRACT

Multimodal large language models (MLLMs) are increasingly explored as perception and reasoning modules for embodied agents. Physical approach provides a concrete setting for evaluating their spatial understanding: as a robot moves toward a persistent target, target-relative geometry, ego-motion, and visibility change together. We introduce RoboApproachBench, a real-robot benchmark for diagnosing visual spatial understanding during recorded approaches. The benchmark contains 9,491 questions across 14 tasks, constructed from a corpus of 2,650 real-robot approach trajectories. Its tasks cover goal grounding, target geometry and approach progress, ego-motion, turn-sequence structure, and target visibility and bearing. Models receive task-specific monocular RGB observations and explicitly stated task information, while synchronized sensor measurements and target annotations support reference-answer construction. We evaluate 28 open- and closed-source MLLMs and observe uneven task-level performance. Across 28 MLLMs, peak-turn characterization exceeds temporal localization in 27 of the 28 models, while leading models' strengths in approach ordering and visibility-event recognition coexist with weaker target-relative geometry. Across same-scale Qwen3-VL pairs, Thinking variants improve aggregate scores but exhibit task-specific regressions. Controlled bearing diagnostics on Qwen3.8-Flash yield higher accuracy with post-reappearance than pre-reappearance prefixes, highlighting the importance of observation endpoints when estimating earlier invisible states. RoboApproachBench provides a trajectory-grounded testbed for diagnosing task-specific strengths and limitations in MLLMs' visual spatial understanding during real-robot approaches.

1 INTRODUCTION

Multimodal large language models (MLLMs) show strong visual recognition and multimodal reasoning capabilities (Bai et al., 2025b), motivating their evaluation as perception and reasoning modules for embodied agents (Majumdar et al., 2024; Yang et al., 2025b). For such agents, recognizing task-relevant objects and locating them in the environment are foundational capabilities. Yet an object that is not already within reach must first be physically approached before subsequent interaction can occur. This final approach phase forms a common transition between navigation and object-directed interaction. As the robot translates and turns, its viewpoint and goal-relative geometry continually change; its speed may vary, and the goal may become temporarily unobserved (Kaelbling et al., 1998). Recognizing the goal or recalling a fixed environmental location is therefore only part of the problem: the model must also infer its spatial relationship to the robot at the relevant time.

Reasoning about the approach requires interpreting motion, geometry, and temporal context. The distance traveled by the robot must be distinguished from its net progress toward the goal, while the goal's bearing at a queried time may differ from its last visible bearing because the robot continues to move. Temporary target disappearance introduces an additional question: where is the target relative to the robot when it is no longer visible? These distinctions motivate our central question: which aspects of target-relative geometry, ego-motion, and visibility can MLLMs infer from visual observations of a real-robot approach?

Existing evaluations cover important components of this question. They have examined spatial relations and quantitative geometry (Liu et al., 2023; Fu et al., 2024; Liao et al., 2024), sequential spatial reasoning and belief maintenance (Yang et al., 2025a; Li et al., 2025; Thompson et al., 2025;

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Liao et al., 2026), and interactive task execution (Yang et al., 2025b). Our focus is their organization around physical approach, namely the transition from navigation to object-directed interaction, as a common setting for evaluation. Within this setting, robot motion, target-relative geometry, and visibility changes provide complementary questions about a recorded real-robot process.

To evaluate these questions within a shared data and reference-construction framework, we introduce RoboApproachBench, a real-robot benchmark for diagnosing visual spatial understanding during recorded approaches. It comprises 9,491 questions across 14 tasks and five approach-centric capabilities, derived from a locked corpus of 2,650 real-robot approach trajectories toward human-designated goals. Models receive task-specific monocular RGB observations, questions, and explicitly stated task parameters. Full sensor records and target annotations remain privileged and support reference-answer construction and auditing.

We evaluate 28 MLLMs and identify three diagnostic patterns. First, complementary readouts of the same approach exhibit different performance profiles. Higher approach-ordering scores coexist with weaker target-depth and progress scores among the leading models, while peak-turn characterization exceeds extremum-interval localization in 27 of the 28 evaluated models. Similarly, visibility-event accuracy exceeds lost-target bearing accuracy: Gemini-3.8-Flash reaches 91.33% on loss onset but only 38.00% on unseen-target bearing. Strong turn characterization and visibility-event timing therefore do not establish accurate motion-extremum localization or unseen-target geometry. Second, aggregate gains can conceal task-specific regressions: across four same-scale Qwen3-VL pairs, Thinking variants improve PeakYaw and LossDur but reduce AppOrder and AT-ID accuracy in every pair, with no consistent improvement on LostBear. Third, the benefit of additional observations depends on the observation condition. In fixed-query diagnostics on Qwen3.8-Flash, extending a causal prefix to just before reappearance does not yield a higher point estimate, whereas extending it past reappearance does. Together, these findings motivate task-level spatial diagnosis with explicit observation conditions, rather than relying on aggregate accuracy alone.

Our contributions are threefold:

- Approach-centric task design: We organize spatial diagnosis around real-robot approaches to persistent targets, covering complementary questions about goal grounding, target-relative geometry, ego-motion, turn structure, and visibility.
- Trajectory-grounded benchmark construction: We derive 9,491 questions across 14 tasks from 2,650 real-robot approach trajectories. Synchronized measurements and target annotations support reference-answer construction, while models receive task-specific monocular RGB observations.
- Task-level evaluation and bearing diagnostics: We evaluate 28 MLLMs, identify cross-model differences between motion characterization and localization, and analyze selective gains across paired model variants. Controlled bearing diagnostics further examine observation availability and queried times.

2 Related Work

2.1 Spatial Reasoning in Multimodal Models

Prior work evaluates spatial relations, depth, size, distance, and cross-view geometry from single or multiple images, including VSR, BLINK, 3DSRBench, Q-Spatial, SpatialVLM, SpatialRGPT, MMSI-Bench, and SPAR-Bench (Liu et al., 2023; Fu et al., 2024; Ma et al., 2025; Liao et al., 2024; Chen et al., 2024; Cheng et al., 2024; Yang et al., 2026; Zhang et al., 2025). MindCube, Spatial-DISE, and SpatialTree extend this space to cognitive mapping, capability taxonomies, and agentic competence (Wang et al., 2025a; Huang et al., 2026; Xiao et al., 2025). These works establish broad spatial capabilities but generally evaluate separate views or skills. Our questions are organized around the target-relative geometry, robot motion, and visibility changes of a real-robot approach toward a persistent target.

Table 1: Comparison with direct neighbors. Columns separate model-visible input, answer-side truth, data embodiment, target/state handling, metric/ego-motion coverage, and primary scope. A check indicates explicit coverage in at least one task or subset.

Work	Input	Answer truth	Data	Target/ state	Metric/ ego	Primary scope
STI-Bench (Li et al., 2025)	V	G	R	–	✓	Pose, depth, displacement, speed, and trajectory
VSTemporalI-Bench (Fan et al., 2026)	V	G	M	–	✓	Camera displacement/direction; absolute/relative distance
OSI-Bench (Wu et al., 2026)	V	S	R	–	✓	Relations; static/dynamic metric; camera self-motion
Multi-SpatialMLLM (Xu et al., 2026)	MV	G	M	–	✓	Depth, correspondence, camera/object motion, and size
VLM4D (Zhou et al., 2025)	V	H	R/M	–	✓	4D translation, rotation, viewpoint change, continuity
Dyn-3D (Ding et al., 2026)	V	G	M	–	✓	Counterfactual ego-motion and kinematic trends
OST-Bench (Lin et al., 2025)	Str.	G	R	*state*	✓	Online observer state and agent–object geometry
S^3-Bench (Wei et al., 2026)	Str./Int.	G	M	*state*	✓	Streaming displacement, distance, and temporal queries
REM (Thompson et al., 2025)	V/MV	H	Sim.	*state*	–	Object constancy, counting, and cross-view relations
SpaMEM (Liao et al., 2026)	V	H/G	Sim.	*state*	–	Perception–memory integration for dynamic spatial state
Last-Mile (Wasserman et al., 2023)	RGB-D	G/S	R-rob.	*goal*	✓	Closed-loop goal approach
RoboApproachBench (ours)	Seq.*	S	R-rob.	✓	✓	Approach-centric spatial diagnosis from recorded real-robot trajectories

Notation. V/MV: single/multiple videos; Str./Int.: streaming/interactive; Seq.: task-specific frames. R: real video; R-rob.: real-robot; M: mixed; Sim.: simulated/controlled. Truth: H, human-provided answers; G, geometry; S, synchronized sensors. *state* and *goal* denote general state maintenance and navigation goals, not the persistent approach target used here. Metric/ego indicates explicit coverage. ✓: included; –: not evaluated under these definitions. *Our tasks use chronological sequences, local windows, shuffled frames, or a causal prefix; only lost-target bearing excludes future frames by design.

2.2 Spatial Reasoning from Video and Streaming Observations

Video benchmarks already cover motion and changing evidence. VSI-Bench and STI-Bench evaluate video spatial understanding, with STI-Bench covering scale, pose, displacement, speed, acceleration, orientation, and trajectories (Yang et al., 2025a; Li et al., 2025). VSTemporalI-Bench evaluates camera displacement, direction, and camera–object absolute/relative distance from monocular sequences, while OSI-Bench combines relation, static/dynamic metric, object motion, and camera self-motion using synchronized sensors (Fan et al., 2026; Wu et al., 2026). Multi-SpatialMLLM, VLM4D, and Dyn-3D further cover multi-frame depth/correspondence, 4D motion continuity, and counterfactual ego-motion controls (Xu et al., 2026; Zhou et al., 2025; Ding et al., 2026). Streaming protocols constrain evidence availability: OST-Bench, S^3-Bench, and VSI-SUPER study accumulated observer state, temporal distance queries, recall, and counting (Lin et al., 2025; Wei et al., 2026; Yang et al., 2025c). Thus monocular distance change, ego-motion, and joint spatial–temporal evaluation are prior targets; our distinction is their organization around one real-robot approach with target visibility changes.

2.3 Embodied Spatial Evaluation and State Maintenance

RoboSpatial, ERQA, MV-RoboBench, and CoopSR study reference frames, interaction, cross-view geometry, and multi-robot observations (Song et al., 2025; Gemini Robotics Team, 2025; Feng et al., 2026; Peng et al., 2026). REM evaluates object constancy and relations over trajectories, while SpaMEM separates perception, state updates, and visual-history reasoning (Thompson et al., 2025; Liao et al., 2026). Last-Mile Embodied Visual Navigation studies geometric approach in an RGB-D closed-loop setting (Wasserman et al., 2023); ours diagnoses recorded approaches offline from task-specific monocular RGB. Finally, AGQA-Decomp and QuantiPhy provide prior diagnostics

for cross-question consistency and visual/input faithfulness (Gandhi et al., 2022; Li et al., 2026). These studies further distinguish answer consistency and input faithfulness from task accuracy. Our contribution is a trajectory-grounded diagnosis of target-relative geometry, ego-motion, and visibility changes within one real-robot approach; Table 1 compares this scope with direct neighbors.

3 ROBOAPPROACHBENCH

Physical approach denotes the process of a robot moving toward a goal, potentially through turns and detours. In this setting, we use spatial state to describe the robot's task-relevant geometry and motion, together with the goal's relative geometry and visibility at a given time. Although the goal remains stationary, its relationship to the moving robot can change: traveled path length need not equal net approach progress, and the goal's bearing can change while it is temporarily unobserved. These distinctions make both observer motion and target-relative state necessary for describing the approach process.

3.1 PROBLEM FORMULATION

Task-based spatial diagnosis. We evaluate visual spatial understanding through task-specific questions about recorded physical approaches. Each instance supplies selected monocular RGB frames in task-defined order, a question, answer choices where applicable, and explicit task parameters when needed. Queries concern target geometry, robot motion, turn and visibility events, or target bearing at specified times and intervals. Observation protocols vary by task, including chronological sequences, local windows, shuffled frames, and causal prefixes. Per-task accuracy is the primary evaluation measure; focused diagnostics additionally examine the effects of available observations and queried times.

Approach quantities and diagnostic queries. During physical approach, we query both the robot's motion and its changing spatial relationship to the goal. For visible endpoints $t_a < t_b$, net approach progress is

$$\Delta r^g_{a:b} = r^g_{t_a} - r^g_{t_b},$$

where r^g_t denotes the Euclidean camera-to-goal range. Positive values indicate approach and negative values indicate separation. This quantity is distinct from the path length $L_{a:b}$, which accumulates robot translation over the interval rather than measuring net range reduction. Motion queries characterize translational speed and heading changes through their magnitudes, extrema, and temporal patterns, while turn-sequence queries recover the order of salient turns. Farther-to-nearer ordering examines ordinal approach progression, using source chronology on manually selected monotonic approaches rather than re-sorting measured depths. Visibility-event queries recover when the goal disappears, when it reappears, and how long it remains unobserved. For a closed visibility-loss event with first invisible frame t_ℓ and first subsequent visible frame t_a, the invisible-frame count is $D_{\text{occ}} = t_a - t_\ell$; together with the sequence length T, this determines its percentage bin. Recovering this visibility history is distinct from inferring the goal's bearing during the invisible interval. The latter relates the last visible target to subsequent motion and queries the bearing relative to the robot's forward axis at the requested time. These queries examine complementary aspects of state evolution: geometric change, motion magnitude and temporal structure, and target-relative inference under changing visibility.

3.2 APPROACH-CENTRIC CAPABILITY TAXONOMY

We organize the 14 diagnostic tasks into five approach-centric capability groups according to the queried spatial information and temporal scope. Together, they cover goal grounding, target geometry and approach progress, ego-motion, turn-sequence structure, and target visibility and bearing. This organization provides a task-level profile of model performance across complementary aspects of a recorded approach.

Approach-Goal Grounding uses Approach Target Identification (AT-ID) to infer the approach goal from the visual trajectory, establishing the referent for target-relative queries. Approach-Progress Spatial State Inference examines proximity and its evolution through Target Depth Estimation (T-Depth) for optical-axis depth, Approach Progress Reasoning (AppProg) for Euclidean range change

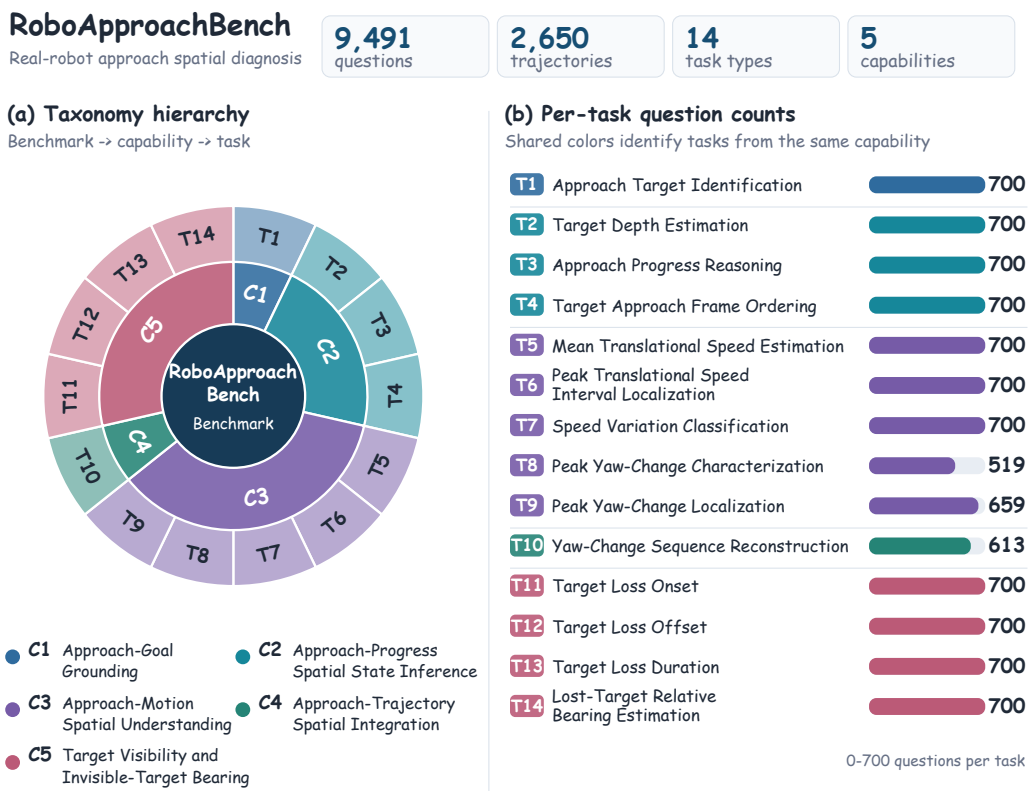


Figure 1: Benchmark organization and question distribution. (a) Five capability groups organize 14 diagnostic tasks spanning goal grounding, target geometry and approach progress, ego-motion, turn-sequence structure, and target visibility and bearing; shared colors identify tasks within each group. (b) The bars report the number of question instances assigned to each task. Their counts sum to 9,491 questions constructed from 2,650 recorded real-robot approach trajectories. Because related tasks can use reference signals from the same recorded process, these task counts do not represent disjoint sets of trajectories.

rather than traveled path length, and Approach Target Frame Ordering (AppOrder) for farther-to-nearer ordering of shuffled monotonic-approach frames.

Approach-Motion Spatial Understanding assesses the robot's own motion through Mean Translational Speed Estimation (MeanSpd), Peak Translational Speed Interval Localization (PeakSpd), and Speed Variation Classification (SpdVar), covering mean speed, peak location, and temporal speed patterns. Peak Yaw-Change Characterization (PeakYaw) and Peak Yaw-Change Localization (YawLoc) respectively characterize the direction and magnitude of the largest adjacent-frame turn and locate its interval. Approach-Trajectory Spatial Integration examines the temporal organization of motion through Yaw-Change Sequence Reconstruction (YawSeq), which recovers the ordered sequence of salient left- and right-turn events rather than per-frame yaw.

Target Visibility and Invisible-Target Bearing distinguishes visibility-history recovery from geometric inference during temporary target disappearance. Target Loss Onset (LossOn), Target Loss Offset (LossOff), and Target Loss Duration (LossDur) query disappearance, reappearance, and the intervening invisible span. Lost-Target Relative Bearing Estimation (LostBear) instead queries the goal's bearing relative to the robot's forward axis at an invisible query time, using only the causal prefix ending at that frame; visibility-event tasks may include reappearance.

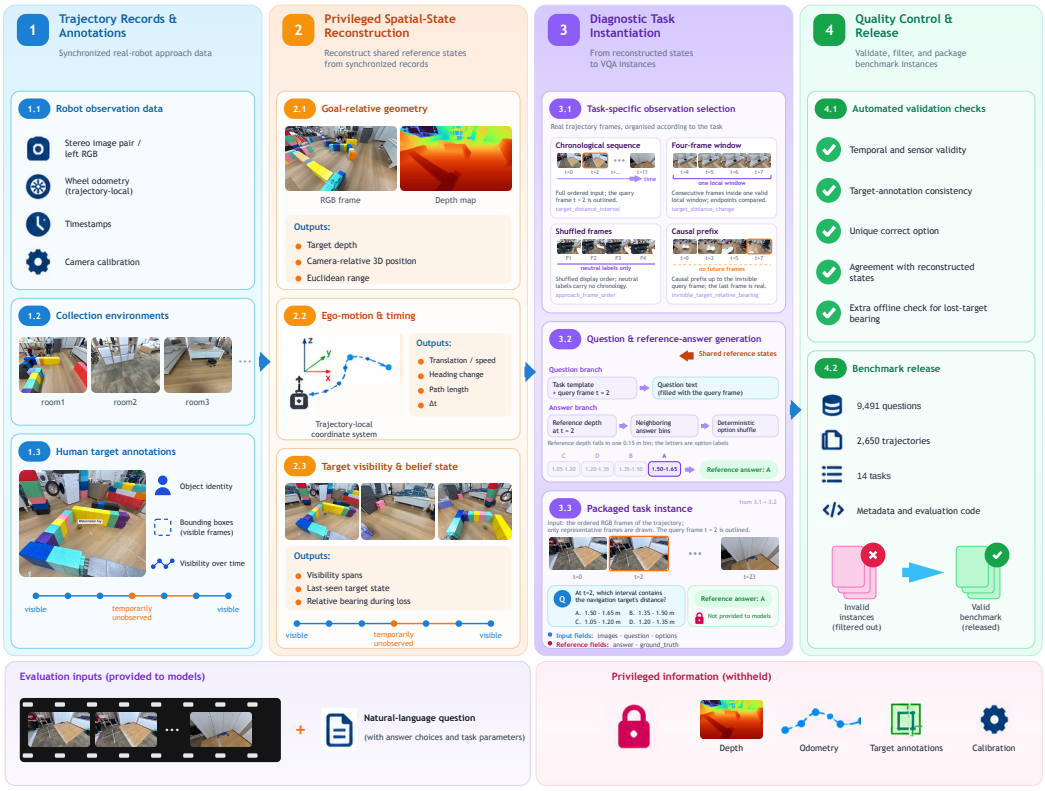


Figure 2: Benchmark construction pipeline. Synchronized records and human target annotations are the inputs; privileged reference reconstruction, diagnostic task instantiation, and quality control produce benchmark instances.

3.3 Trajectory-Grounded Benchmark Construction

We reconstruct task-relevant reference quantities from each real-robot approach trajectory, its synchronized measurements, and target annotations, then generate diagnostic questions about geometry, motion, and visibility. Related tasks can share reference signals from the same recorded process, while following their own observation and answer-construction rules (Fig. 2).

Privileged Reference Reconstruction. Target annotations specify identity, bounding boxes, and visibility. Stereo-derived depth and calibration recover target geometry, while trajectory-local odometry and timestamps yield translation, speed, and heading changes. Reference bearing during temporary loss of visibility combines last-visible geometry with subsequent robot motion. Diagnostic Task Instantiation. Task-specific rules select eligible observations and queried times or intervals, then generate questions and deterministic reference answers, with distractors where applicable. Each instance follows its task's observation protocol. Related tasks share reference sequences: heading changes, for example, support peak-turn characterization, peak localization, and turn-sequence reconstruction. Quality Control. Checks verify record availability, temporal validity, agreement between answers and reconstructed quantities, and a unique reference-matching option for each multiple-choice instance. For the causal lost-target bearing task, reappearance provides an offline geometric consistency check but remains excluded from model input. Instances failing these checks are discarded. Appendix B provides the construction and validation details.

Table 2: Per-task accuracy (%) for 28 MLLMs across RoboApproachBench's 14 tasks, grouped by the five capability categories in Figure 1. Closed-source/API models are evaluated on 300 questions per task (4,200 total), whereas open-source models are evaluated on the full 9,491-question benchmark. Rank orders models by Avg., the unweighted mean of the 14 task accuracies. AppOrder contributes exact-match accuracy, and AT-ID is scored by the frozen reference-based semantic judge. Dark and light shading mark the highest and second-highest scores in each numeric column; ties share shading. Task abbreviations follow Section 3.

			Approach-Goal Grounding	Approach-Progress Spatial State Inference			Approach-Motion Spatial Understanding					Approach-Trajectory Spatial Integration	Target Visibility and Invisible-Target Bearing			
Model	Rank	Avg.	AT-ID	T-Depth	AppProg	AppOrder EM	MeanSpd	PeakSpd	SpdVar	PeakYaw	YawLoc	YawSeq	LossOn	LossOff	LossDur	LostBear
Open-source models																
Qwen3.5-397B-A17B	13	33.19	25.57	27.29	38.43	49.71	18.29	27.29	23.57	45.47	42.94	37.19	31.00	43.29	30.29	24.29
GLM-4.6V-Flash	21	24.79	24.43	23.57	26.43	15.43	27.57	32.29	34.86	40.46	23.82	20.88	26.14	4.86	28.00	18.29
InternVL3.5-30B-A3B	23	23.73	14.57	23.57	26.71	8.71	32.29	29.57	31.71	31.21	24.43	22.68	18.57	16.57	24.00	27.57
InternVL3.5-8B	22	23.75	17.57	27.14	26.71	14.29	32.29	27.57	32.29	27.36	13.66	6.36	22.00	23.71	27.29	34.29
Cosmos-Reason2-8B	20	26.54	22.29	28.29	28.71	14.57	32.71	28.29	18.14	33.33	33.23	22.68	21.71	49.57	24.86	13.14
LLaVA-OneVision-Qwen2-7B	26	19.69	13.00	26.00	25.00	3.43	25.57	29.57	32.14	20.42	21.70	18.43	11.57	6.00	23.29	19.57
Qwen2.5-VL-7B	27	19.19	23.86	26.71	26.71	8.00	32.71	25.14	27.71	26.01	14.57	1.63	2.57	16.29	26.00	10.71
Nemotron-Nano-VL-8B	24	20.16	12.29	20.71	25.00	4.86	27.57	22.71	33.43	26.40	21.40	22.02	12.29	6.43	22.43	24.71
InternVL3-1B	25	19.86	7.14	24.00	24.43	1.14	26.00	31.14	30.14	29.29	19.88	25.12	1.71	6.29	27.71	24.00
Gemma-3n-E2B	28	17.90	1.71	23.57	25.00	3.71	18.29	22.71	33.29	26.01	20.79	19.09	8.86	6.43	19.57	21.57
RoboBrain2.5-8B	18	27.44	33.71	26.71	30.43	20.86	24.00	34.00	30.14	31.60	20.94	31.81	27.71	29.86	22.86	19.57
Qwen3-VL-8B-Instruct	17	28.18	33.29	25.29	29.43	19.29	32.29	32.29	47.29	29.48	20.33	20.39	29.29	35.29	28.29	12.29
Qwen3-VL-8B-Thinking	15	29.54	26.29	17.29	28.43	10.29	30.29	27.29	37.29	54.53	20.33	29.36	34.29	41.29	39.29	17.29
Qwen3-VL-30B-A3B-Instruct	19	26.82	29.29	19.29	30.43	15.29	32.29	26.29	33.29	30.44	15.33	34.42	20.29	40.29	22.29	26.29
Qwen3-VL-30B-A3B-Thinking	16	29.19	23.29	21.29	28.43	13.29	36.29	28.29	38.29	55.49	23.37	27.41	26.29	33.29	37.29	16.29
Qwen3-VL-32B-Instruct	14	33.14	36.29	29.29	30.71	35.29	34.29	35.29	33.29	37.57	18.36	37.36	39.29	54.29	25.29	17.29
Qwen3-VL-32B-Thinking	10	37.05	32.29	30.29	33.43	19.29	35.29	31.29	26.29	60.50	29.44	51.39	50.29	53.29	47.29	18.29
Qwen3-VL-235B-Instruct	11	34.36	35.29	22.29	34.71	35.29	41.29	32.29	32.29	40.46	12.44	31.48	55.29	52.29	28.29	27.29
Qwen3-VL-235B-Thinking	8	39.04	30.29	24.29	32.43	26.29	41.29	30.29	43.29	65.51	35.36	46.33	42.29	55.29	47.29	26.29
Closed-source models (API)																
Gemini-3.8-Flash	1	60.88	41.33	41.33	33.33	81.33	41.33	41.33	46.33	91.33	56.33	86.33	91.33	81.33	81.33	38.00
GPT-5.6-Luna	3	56.38	51.33	28.00	35.33	51.33	38.00	38.00	51.33	61.33	51.33	86.33	96.33	91.33	68.00	41.33
GPT-5.6-Sol	2	57.17	40.00	28.00	32.33	74.67	38.00	34.67	78.33	78.00	63.00	89.33	86.00	55.33	58.00	44.67
Qwen3.8-Max	4	47.47	39.33	30.33	43.33	72.33	28.33	43.33	40.33	65.33	32.33	46.33	83.33	77.33	33.33	29.33
Qwen3.7-Plus	6	40.45	41.33	32.33	40.00	57.33	21.33	31.33	38.33	47.33	22.33	39.33	64.33	69.33	30.33	31.33
Qwen3.7-Flash	9	37.19	35.33	32.33	31.33	54.33	23.33	36.33	25.33	47.33	20.33	26.33	66.33	71.33	27.33	23.33
Qwen3.5-Omni-Plus	7	39.93	37.33	27.33	37.67	56.33	25.33	37.33	29.33	53.33	30.33	30.33	60.33	66.33	40.33	27.33
DeepSeek-V4.1-Flash	12	33.57	28.33	31.33	42.33	32.33	21.33	41.33	17.00	51.33	9.33	36.00	43.67	63.00	24.67	28.00
Kimi-K3	5	42.66	33.33	33.33	38.00	66.33	29.33	36.33	30.33	73.33	23.33	24.33	64.33	67.33	41.33	36.33

4 Evaluation on RoboApproachBench

4.1 Evaluation Setup

We evaluate all models zero-shot using the task-specific input protocols in Section 3. Evaluated models and configurations appear in Appendix D.4; decoding and answer extraction are described in Appendix D.5. The API cohort is evaluated on 300 questions per task, and the open-source cohort on the full benchmark; Table 21 reports evaluation coverage. We report per-task accuracy and Avg., the unweighted mean of all 14 task accuracies, including AppOrder exact match. Figure 3 summarizes capability-level accuracy for representative models and task-macro accuracy for all 28 evaluated models. AT-ID uses a frozen `qwen3.8max` reference-based semantic judge (Appendix F.1). AppOrder uses exact match, with invalid permutations counted as incorrect; missing, repeated, unknown, or extra frame labels are likewise scored as incorrect.

4.2 Main Results

Table 2 shows uneven performance across approach-related spatial tasks. Stronger results on ordering and motion characterization are not accompanied by uniformly high accuracy on target-relative geometry and temporal localization. Gemini-3.8-Flash achieves 81.33% on AppOrder, compared with 41.33% on T-Depth and 33.33% on AppProg. Among the three highest-ranked models, AppOrder ranges from 51.33–81.33% while AppProg remains at 32.33–35.33%, showing that higher approach-ordering scores coexist with lower range-change estimation scores in these models. Farther-to-nearer ordering does not specify target depth at a queried time or the amount of camera-to-target range change over an interval. These readouts therefore assess complementary aspects of the evolving robot–goal relationship.

Motion characterization and temporal localization show a broadly shared performance gap: peak-turn characterization exceeds extremum-interval localization in 27 of the 28 evaluated models (Table 2). Gemini-3.8-Flash, for example, reaches 91.33% on PeakYaw but 56.33% on YawLoc, despite

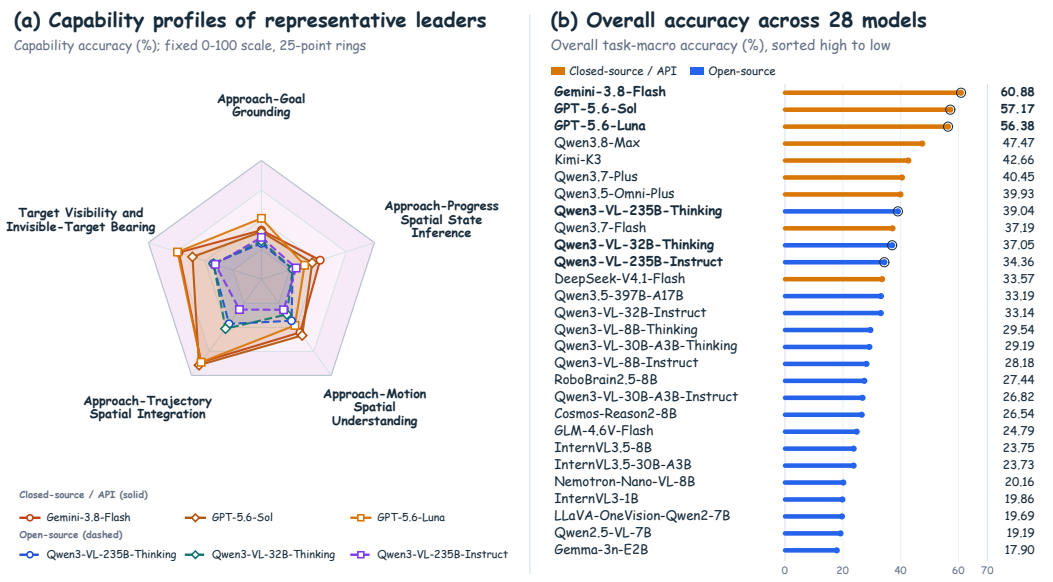


Figure 3: Capability profiles and overall model performance. (a) Five capability-level accuracies for the three highest-overall closed-source/API models and the three highest-overall open-source models. Each axis is the unweighted mean of the tasks assigned to that capability and uses a fixed 0–100 scale. (b) Overall task-macro accuracy for all 28 evaluated models, sorted from high to low and colored by the model groups used in Table 2. Overall is the unweighted mean of the 14 task accuracies, including Target Approach Frame Ordering exact match; it is therefore a task-weighted, rather than capability-weighted, summary.

reaching 91.33% and 81.33% on loss-onset and loss-offset localization. Temporal performance is therefore task-dependent rather than uniformly weak. A correct direction–magnitude category does not uniquely identify the interval of the largest turn, since multiple intervals can share the same category.

Aggregate improvements can mask task-specific regressions. Across four Qwen3-VL Instruct/Thinking pairs at matching scales (8B, 30B-A3B, 32B, and 235B), Thinking variants improve Avg. by 1.36–4.68 percentage points. PeakYaw increases by 22.93–25.05 points and LossDur by 11.00–22.00 points in every pair. However, AppOrder decreases by 2.00–16.00 points and AT-ID by 4.00–7.00 points in every pair, while LostBear shows no consistent improvement. Higher task-macro accuracy therefore does not imply uniform gains in goal grounding, approach ordering, or hidden-target geometry. These comparisons concern model variants under their evaluated configurations, rather than an isolated intervention on inference-time reasoning.

Goal grounding remains an additional limitation: AT-ID reaches at most 51.33% under semantic evaluation. This result concerns identifying the designated approach goal; it does not identify the cause of errors in separately evaluated geometric queries that explicitly name the target. These task-level contrasts describe performance under the respective input and answer protocols; they do not by themselves isolate the causes of errors. The following analyses examine invisible-target bearing and its observation conditions in greater detail (Sections 4.3 and 4.4).

4.3 Visibility Events versus Hidden Target Geometry

The target-loss tasks distinguish the visibility timeline from the geometry of the unseen goal. For Gemini-3.8-Flash, LossOn, LossOff, and LossDur reach 81.33–91.33%, whereas LostBear reaches 38.00% (Table 2). This distinction matters during physical approach: although the goal remains stationary, its robot-relative bearing can change as the robot moves. Recovering when the goal disappears therefore does not establish recovery of its evolving geometry while unseen. These task-level differences identify an uneven capability profile, rather than its cause. LostBear receives only a

Table 3: Bearing diagnostics on Qwen3.8-Flash (accuracy, %). The fixed-query study evaluates 516 questions under four observation conditions; the fixed-observation study evaluates three queried times in each of 64 episodes. The 25% random baseline applies to individual four-option queries.

Diagnostic	Condition	Accuracy (%)
Fixed query, varying observations	Text-only	24.8
	Causal prefix	42.1
	Pre-reappearance prefix	41.1
	Post-reappearance prefix (future-assisted)	47.5
Fixed observations, varying query time	Observation, first-loss query	42.19
	Observation, middle query	39.06
	Observation, last-invisible query	45.31
	All three queries correct (7/64 episodes)	10.94

causal prefix, whereas visibility-event tasks may include reappearance; their marginal accuracies also do not establish incompatible answers within a particular event. Section 4.4 therefore examines how predictions respond to changes in available observations and in the time of the queried state.

4.4 Controlled Diagnostics of Invisible-Target Bearing

We conduct two supplementary bearing diagnostics on Qwen3.8-Flash. The fixed-query study evaluates 516 questions under four observation conditions, while the fixed-observation study queries three times within each of 64 episodes (192 questions). Both adopt a four-cell option set (Appendix C). Table 3 summarizes results for how invisible-target bearing predictions vary with available observations and queried time.

For fixed queries, accuracy rises from 24.8% without images to 42.1% with the causal prefix. Extending the prefix to just before target reappearance yields 41.1%, whereas extending it past reappearance yields 47.5%, a further 6.4 percentage points relative to the pre-reappearance condition. Thus, a longer prefix that still excludes reappearance does not yield a higher point estimate; additional improvement is observed in the condition that includes reappearance. The distinction is not simply between past and future observations, because the pre-reappearance prefix can already extend beyond the queried time. These results motivate specifying both the queried time and the observation endpoint relative to visibility events. The comparison distinguishes observation conditions rather than isolating the effect of target reappearance itself.

Under identical observations, accuracy does not change monotonically across the three queried timestamps. Correct bearing predictions for all three queries are achieved in only seven out of 64 episodes (10.94%). This episode-level joint metric reports correctness across multiple queried times within a single approach, information unavailable from per-query accuracy alone. These diagnostics characterize bearing estimation under controlled observation conditions and are not designed as a direct test of internal state maintenance.

5 Conclusion

We introduce RoboApproachBench, an offline benchmark for diagnosing visual spatial understanding during recorded real-robot approaches to persistent targets. Its 14 tasks cover goal grounding, target-relative geometry and approach progress, ego-motion, turn-sequence structure, and target visibility and bearing, with reference answers constructed from synchronized measurements and target annotations. Evaluation of 28 MLLMs shows that strengths in approach ordering, turn characterization, and visibility-event recognition coexist with weaker target-relative geometry and motion-extremum localization. Paired Qwen3-VL comparisons further show that higher aggregate scores can conceal regressions in goal grounding and approach ordering. Controlled bearing diagnostics distinguish an invisible interval from extending observations within them past target reappearance, while episode-level joint accuracy complements individual-query scores. Together, these results support spatial evaluation that separates queried quantities, task-specific model differences, and available observation conditions. RoboApproachBench provides a trajectory-grounded testbed for characterizing these strengths and limitations in real-robot approach settings.

AI USE STATEMENT

We used generative AI tools to assist with manuscript drafting and polishing and with the retrieval and discovery of related work. These tools also assisted with data preprocessing, code generation and revision, visualization, the computation and aggregation of experimental results, and feedback on the methodology and analysis. The use of a language model for reference-based semantic judging is described in Section 4.1 and Appendix F.1. The authors take full responsibility for the final manuscript, code, data processing, reported results, and conclusions.

REPRODUCIBILITY STATEMENT

Appendix B documents trajectory collection and annotation, reference-answer construction, sampling, and validation. Appendix D specifies the model input and evaluation protocols, including prompt templates and answer parsing. Appendix E describes the benchmark record schema and planned release artifacts, and Appendix F details the reference-based semantic-judging protocol. The benchmark questions and images, construction parameters, generation code, evaluation scripts, and scoring configurations will be publicly released upon acceptance.

REFERENCES

Alibaba Cloud. Model studio vision models. Alibaba Cloud Model Studio documentation, 2026. URL https://help.aliyun.com/en/model-studio/vision-model/. Accessed: 2026-09-19.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025a.

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*, 2025b.

Boyuan Chen, Zhuo Xu, Sean Kirmani, Brian 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, 2024.

An-Chieh Cheng, Hongxu Yin, Yang Fu, Qiushan Guo, Ruihan Yang, Jan Kautz, Xiaolong Wang, and Sifei Liu. SpatialRGPT: Grounded spatial reasoning in vision-language models. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-4293.

DeepSeek-AI. Deepseek-V4.1-Flash: Pushing the limits of KV cache compression. *arXiv preprint arXiv:2609.19969*, 2026. URL https://arxiv.org/abs/2609.19969.

Jiayu Ding, Zhuodong Liu, Lei Zhang, Manyu Xiong, Hongbo Jin, Haoran Tang, Hongbo Zhang, Changen Zhu, and Wenbo Xing. Dyn-3D: Unveiling and resolving ego-motion ambiguity in vision-language models. *arXiv preprint arXiv:2609.01059*, 2026. URL https://arxiv.org/abs/2609.01059.

Zhiwen Fan, Jian Zhang, Renjie Li, Junge Zhang, Runjin Chen, Hezhen Hu, Kevin Wang, et al. VLM-3R: Vision-language models augmented with instruction-aligned 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Fan_VLM-3R_Vision-Language_Models_Augmented_with_Instruction-Aligned_3D_Reconstruction_CVPR_2026_paper.html.

Zhiyuan Feng, Zhaolu Kang, Qijie Wang, Zhiying Du, Jiongrui Yan, Shubin Shi, Chengbo Yuan, Huizhi Liang, Yu Deng, Qixiu Li, Rushuai Yang, Arctanx An, Leqi Zheng, Weijie Wang, Shawn Chen, Sicheng Xu, Yaobo Liang, Jiaolong Yang, and Baining Guo. Seeing across views: Benchmarking spatial reasoning of vision-language models in robotic scenes. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=jXDZJAfRZB.

Paul Foster, Zhenghong Sun, Jong Jin Park, and Benjamin Kuipers. VisAGGE: Visible angle grid for glass environments. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2213–2220, 2013. doi: 10.1109/ICRA.2013.6630875.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. BLINK: Multimodal large language models can see but not perceive. In *Computer Vision – ECCV 2024*, volume 15081 of *Lecture Notes in Computer Science*, pp. 148–166. Springer, 2024. doi: 10.1007/978-3-031-73337-6_9.

Mona Gandhi, Mustafa Omer Gul, Eva Prakash, Madeleine Grunde-McLaughlin, Ranjay Krishna, and Maneesh Agrawala. Measuring compositional consistency for video question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. URL https://arxiv.org/abs/2204.07190.

Rui Gao, Jisun Park, Xiaohang Hu, Seungjun Yang, and Kyungeun Cho. Reflective noise filtering of large-scale point cloud using multi-position LiDAR sensing data. *Remote Sensing*, 13(16):3058, 2021. doi: 10.3390/rs13163058.

Gemini Robotics Team. Gemini robotics: Bringing ai into the physical world. *arXiv preprint arXiv:2503.20020*, 2025.

Gemma Team. Gemma 3n model card. Google DeepMind model card, 2025. URL https://ai.google.dev/gemma/docs/gemma-3n/model_card. Accessed: 2026-09-10.

Google DeepMind. Gemini 3.8 Flash model card. Google DeepMind model card, 2026. URL https://deepmind.google/models/model-cards/gemini-3-8-flash/. Accessed: 2026-09-19.

Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2 edition, 2004. doi: 10.1017/CBO9780511811685. URL https://doi.org/10.1017/CBO9780511811685.

Xinmiao Huang, Qisong He, Zhenglin Huang, Boxuan Wang, Zhuoyun Li, Guangliang Cheng, Yi Dong, and Xiaowei Huang. Spatial-DISE: A unified benchmark for evaluating spatial reasoning in vision-language models. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=bMINsPQpME.

Leslie Pack Kaelbling, Michael L. Littman, and Anthony R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101(1–2):99–134, 1998. doi: 10.1016/S0004-3702(98)00023-X. URL https://doi.org/10.1016/S0004-3702(98)00023-X.

Kimi Team. Kimi K3: Open frontier intelligence. *arXiv preprint arXiv:2607.24653*, 2026. URL https://arxiv.org/abs/2607.24653.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. LLaVA-OneVision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024. doi: 10.48550/arXiv.2408.03326.

Puyin Li, Tiange Xiang, Ella Mao, Shirley Wei, Xinye Chen, Adnan Masood, Li Fei-Fei, and Ehsan Adeli. QUANTIPHY: A quantitative benchmark evaluating physical reasoning abilities of vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Puyin_QUANTIPHY_A_Quantitative_Benchmark_Evaluating_Physical_Reasoning_Abilities_of_Vision-Language_CVPR_2026_paper.html.

Yun Li, Yiming Zhang, Tao Lin, XiangRui Liu, Wenxiao Cai, Zheng Liu, and Bo Zhao. STI-Bench: Are MLLMs ready for precise spatial-temporal world understanding? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5622–5632, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Li_STI-Bench_Are_MLLMs_Ready_for_Precise_Spatial-Temporal_World_Understanding_ICCV_2025_paper.html.

Chih-Ting Liao, Xi Xiao, Chunlei Meng, Zhangquan Chen, Yitong Qiao, Weilin Zhou, Tianyang Wang, Xu Zheng, and Xin Cao. SpaMEM: Benchmarking dynamic spatial reasoning via perception-memory integration in embodied environments. *arXiv preprint arXiv:2604.22409*, 2026. URL https://arxiv.org/abs/2604.22409.

Yuan-Hong Liao, Rafid Mahmood, Sanja Fidler, and David Acuna. Reasoning paths with reference objects elicit quantitative spatial reasoning in large vision-language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 17028–17047. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.emnlp-main.947. URL https://aclanthology.org/2024.emnlp-main.947/.

Jingli Lin, Chenming Zhu, Runsen Xu, Xiaohan Mao, Xihui Liu, Tai Wang, and Jiangmiao Pang. OST-Bench: Evaluating the capabilities of MLLMs in online spatio-temporal scene understanding. *arXiv preprint arXiv:2507.07984*, 2025.

Fangyu Liu, Guy Emerson, and Nigel Collier. Visual spatial reasoning. *Transactions of the Association for Computational Linguistics*, 11:635–651, 2023. doi: 10.1162/tacl_a_00566. URL https://aclanthology.org/2023.tacl-1.37/.

Wufei Ma, Haoyu Chen, Guofeng Zhang, Yu-Cheng Chou, Jieneng Chen, Celso M. de Melo, and Alan Yuille. 3DSRBench: A comprehensive 3d spatial reasoning benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6924–6934, 2025.

Arjun Majumdar, Anurag Ajay, Xiaohan Zhang, Pranav Putta, Sriram Yenamandra, Mikael Henaff, Sneha Silwal, Paul McVay, Oleksandr Maksymets, Sergio Arnaud, Karmesh Yadav, Qiyang Li, Ben Newman, Mohit Sharma, Vincent Berges, Shiqi Zhang, Pulkit Agrawal, Yonatan Bisk, Dhruv Batra, Mrinal Kalakrishnan, Franziska Meier, Chris Paxton, Alexander Sax, and Aravind Rajeswaran. OpenEQA: Embodied question answering in the era of foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16488–16498, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Majumdar_OpenEQA_Embodied_Question_Answering_in_the_Era_of_Foundation_Models_CVPR_2024_paper.html.

Junhong Min, Youngpil Jeon, Jimin Kim, and Minyong Choi. S^2M^2: Scalable stereo matching model for reliable depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 26729–26739, 2025. doi: 10.1109/ICCV51701.2025.02481.

NVIDIA. Llama-3.1-Nemotron-Nano-VL-8B-V1 model card. NVIDIA NIM, 2025. URL https://build.nvidia.com/nvidia/llama-3.1-nemotron-nano-vl-8b-v1/modelcard. Accessed: 2026-09-10.

NVIDIA. Cosmos Reason 2 documentation. NVIDIA Cosmos documentation, 2026. URL https://docs.nvidia.com/cosmos/latest/reason2/index.html. Accessed: 2026-09-10.

OpenAI. GPT-5.6: Frontier intelligence that scales with your ambition. OpenAI, 2026. URL https://openai.com/index/gpt-5-6/. Accessed: 2026-09-19.

Kunyu Peng, Zhikun Zhou, Kailun Yang, Di Wen, Ruiping Liu, Yufan Chen, Junwei Zheng, Hao Shi, Yi Zhou, M. Saquib Sarfraz, Danda Pani Paudel, and Luc Van Gool. Seeing together: Multi-robot cooperative egocentric spatial reasoning with multimodal large language models. *arXiv preprint arXiv:2605.18431*, 2026. URL https://arxiv.org/abs/2605.18431.

Qwen Team. Qwen3.5-397B-A17B model card. Hugging Face model repository, 2026a. URL https://huggingface.co/Qwen/Qwen3.5-397B-A17B. Accessed: 2026-09-19.

Qwen Team. Qwen3.5-Omni technical report. *arXiv preprint arXiv:2604.15804*, 2026b. doi: 10.48550/arXiv.2604.15804. URL https://arxiv.org/abs/2604.15804.

Qwen Team. Qwen3.8-Max: A new bar for coding and cowork. Qwen official blog, 2026c. URL https://qwen.ai/blog?id=qwen3.8. Accessed: 2026-09-19.

Daniel Scharstein and Heiko Hirschmüller. Middlebury stereo evaluation — version 3. Online benchmark, 2015. URL https://vision.middlebury.edu/stereo/eval3/. Accessed: 2026-09-24.

Daniel Scharstein, Heiko Hirschmüller, York Kitajima, Greg Krathwohl, Nera Nešić, Xi Wang, and Porter Westling. High-resolution stereo datasets with subpixel-accurate ground truth. In *Pattern Recognition (GCPR)*, volume 8753 of *Lecture Notes in Computer Science*, pp. 31–42. Springer, 2014. doi: 10.1007/978-3-319-11752-2_3.

Chan Hee Song, Valts Blukis, Jonathan Tremblay, Stephen Tyree, Yu Su, and Stan Birchfield. RoboSpatial: Teaching spatial understanding to 2d and 3d vision-language models for robotics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Huajie Tan, Enshen Zhou, Zhiyu Li, Yijie Xu, Yuheng Ji, Xiansheng Chen, et al. RoboBrain 2.5: Depth in sight, time in mind. *arXiv preprint arXiv:2601.14352*, 2026. doi: 10.48550/arXiv.2601.14352.

Jacob Thompson, Emiliano Garcia-Lopez, and Yonatan Bisk. REM: Evaluating LLM embodied spatial reasoning through multi-frame trajectories. In *Proceedings of the Conference on Language Modeling*, 2025. URL https://arxiv.org/abs/2512.00736.

Qineng Wang, Baiqiao Yin, Pingyue Zhang, Jianshu Zhang, Kangrui Wang, Zihan Wang, Jieyu Zhang, Keshigeyan Chandrasegaran, Han Liu, Ranjay Krishna, Saining Xie, Jiajun Wu, Fei-Fei Li, and Manling Li. MindCube: Spatial mental modeling from limited views. *arXiv preprint arXiv:2506.21458*, 2025a. URL https://arxiv.org/abs/2506.21458.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. InternVL3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025b.

Justin Wasserman, Karmesh Yadav, Girish Chowdhary, Abhinav Gupta, and Unnat Jain. Last-mile embodied visual navigation. In *Proceedings of the 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pp. 666–678. PMLR, 2023. URL https://proceedings.mlr.press/v205/wasserman23a.html.

Yuxi Wei, Wei Huang, Qirui Chen, Lu Hou, and Xiaojuan Qi. See, remember, explore: A benchmark and baselines for streaming spatial reasoning. *arXiv preprint arXiv:2603.23864*, 2026.

Mingrui Wu, Zhaozhi Wang, Fangjinhua Wang, Jiaolong Yang, Marc Pollefeys, and Tong Zhang. From indoor to open world: Revealing the spatial reasoning gap in MLLMs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16789–16799, 2026. URL https://arxiv.org/abs/2512.19683.

Yuxi Xiao, Longfei Li, Shen Yan, Xinhang Liu, Sida Peng, Yunchao Wei, Xiaowei Zhou, and Bingyi Kang. SpatialTree: How spatial abilities branch out in MLLMs. *arXiv preprint arXiv:2512.20617*, 2025. URL https://arxiv.org/abs/2512.20617.

Runsen Xu, Weiyao Wang, Hao Tang, Xingyu Chen, Xiaodong Wang, Fu-Jen Chu, Matt Feiszli, and Kevin J. Liang. Multi-SpatialMLLM: Multi-frame spatial understanding with multi-modal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Xu_Multi-SpatialMLLM_Multi-Frame_Spatial_Understanding_with_Multi-Modal_Large_Language_Models_CVPR_2026_paper.html.

Jihan Yang, Shusheng Yang, Anjali W. Gupta, Rilyn Han, Fei-Fei Li, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10632–10643, 2025a. URL https://openaccess.thecvf.com/content/CVPR2025/html/Yang_Thinking_in_Space_How_Multimodal_Large_Language_Models_See_Remember_CVPR_2025_paper.html.

Rui Yang, Hanyang Chen, Junyu Zhang, Mark Zhao, Cheng Qian, Kangrui Wang, Qineng Wang, Teja Venkat Koripella, Marziyeh Movahedi, Manling Li, Heng Ji, Huan Zhang, and Tong Zhang. EmbodiedBench: Comprehensive benchmarking multi-modal large language models for vision-driven embodied agents. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 70576–70631. PMLR, 2025b. URL https://proceedings.mlr.press/v267/yang25f.html.

Shusheng Yang, Jihan Yang, Pinzhi Huang, Ellis Brown, Zihao Yang, Yue Yu, Shengbang Tong, Zihan Zheng, Yifan Xu, Muhan Wang, Daohan Lu, Rob Fergus, Yann LeCun, Fei-Fei Li, and Saining Xie. Cambrian-S: Towards spatial supersensing in video. *arXiv preprint arXiv:2511.04670*, 2025c.

Sihan Yang, Runsen Xu, Yiman Xie, Sizhe Yang, Mo Li, Jingli Lin, Chenming Zhu, Xiaochen Chen, Haodong Duan, Xiangyu Yue, Dahua Lin, Tai Wang, and Jiangmiao Pang. MMSI-Bench: A benchmark for multi-image spatial intelligence. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=gHRoX4vXm3.

Z.ai. GLM-4.6V-Flash model card. Hugging Face model card, 2025. URL https://huggingface.co/zai-org/GLM-4.6V-Flash. Accessed: 2026-09-19.

Jiahui Zhang, Yurui Chen, Yueming Xu, Ze Huang, Jilin Mei, Chunhui Chen, Yanpeng Zhou, Yu-Jie Yuan, Xinyue Cai, Guowei Huang, Xingyue Quan, Hang Xu, and Li Zhang. From flatland to space: Teaching vision-language models to perceive and reason in 3d. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-5180. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/e3a0db7c0a191854c176af1d20cdec80-Abstract-Datasets_and_Benchmarks_Track.html. Datasets and Benchmarks Track.

Shijie Zhou, Alexander Vilesov, Xuehai He, Ziyu Wan, Shuwang Zhang, Aditya Nagachandra, Di Chang, Dongdong Chen, Xin Eric Wang, and Achuta Kadambi. VLM4D: Towards spatiotemporal awareness in vision language models. *arXiv preprint arXiv:2508.02095*, 2025.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, et al. InternVL3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025. doi: 10.48550/arXiv.2504.10479.

A Overview and Organization

This appendix provides the implementation details needed to reproduce RoboApproachBench questions and evaluation requests. We first describe trajectory eligibility, the privileged signals used for deterministic ground-truth reconstruction, and the construction rules grouped under the five diagnostic capabilities that instantiate the fourteen task types. We then specify thresholds, discretization, distractor construction, stable sampling, versioning, and automated validation. The second half documents the prompt contract used at inference time, including the shared system instruction, task templates, decoding settings, and deterministic answer parsing. We conclude with the benchmark record schema, representative rendered inputs, the complete semantic-judge prompt, and release information.

The benchmark deliberately separates model-visible evidence from construction signals. The model receives selected monocular RGB observations, neutral frame labels, the question, and answer choices when applicable. Odometry, timestamps, stereo depth, camera calibration, target bounding boxes, source paths, and all reconstructed ground-truth fields remain privileged construction data. The locked benchmark contains 9,491 questions constructed from a corpus of 2,650 source trajectories, covering fourteen task types and five capability groups. Most task types contain 700 questions; Peak Yaw-Change Characterization, Peak Yaw-Change Localization, and Yaw-Change Sequence Reconstruction contain 519, 659, and 613 questions, respectively.

Table 4: Benchmark census. The trajectory count refers to the locked source corpus; object vocabulary size counts distinct annotated target descriptions.

Quantity	Count
Questions	9,491
Source trajectories	2,650
Questions per source trajectory (mean)	3.58
Rooms	3
Task types / capability groups	14 / 5
Distinct annotated target descriptions	507
Bounding-box annotation files inspected	39,275

The 14-task benchmark retains the locked corpus of 2,650 source trajectories. The mean is therefore $9{,}491/2{,}650 = 3.58$ questions per source trajectory; this ratio does not assert that every source trajectory contributes a retained question. For the object vocabulary, we read the bounding-box annotations associated with these trajectories, remove surrounding whitespace from nonempty object labels, and deduplicate identical label strings across frames and trajectories. The resulting 507 entries preserve the annotated object category and attributes such as color; this is the target-description vocabulary used for the reported object count. Repeated views of the same description do not increase the count.

B Question Construction and Benchmark Generation

B.1 Data Collection, Annotation, and Trajectory Eligibility

Trajectory collection. We collect real-robot approach trajectories in indoor environments through keyboard teleoperation. For each recording, a human operator guides the robot toward a designated static target. Depending on the scene layout and the presence of obstacles, the robot approaches the target directly or executes turns before completing the approach. Human data collectors additionally categorize the trajectories according to their motion characteristics to support subsequent task-specific question generation.

Trajectory review and target annotation. Human annotators review each image sequence frame by frame, checking image quality, whether the complete approach has been recorded, and whether data acquisition was interrupted. Only successfully completed approach runs that pass this review are retained in the versioned trajectory eligibility list used by all question generators. Annotators record the designated target's semantic identity and annotate its bounding box in each visible frame,

maintaining a consistent target reference throughout the sequence. Target visibility is recorded frame by frame, with explicit invisibility distinguished from missing annotations. These annotations provide the target reference and image-level observations required by the automated construction pipeline.

Task-specific trajectory selection. Question generators select eligible trajectories according to the manually assigned categories and the construction requirements of each task. In particular, Approach Target Frame Ordering uses only trajectories categorized as monotonic approaches, for which chronological order corresponds to farther-to-nearer progression toward the designated target. The reference ordering therefore follows source chronology rather than a re-sorting of measured depths. Monotonicity is a selection criterion for this task, not an assumption imposed on the entire benchmark. Other tasks draw from trajectory categories compatible with their respective requirements, allowing different aspects of the recorded approaches to support complementary diagnostic questions.

B.2 Shared Trajectory Representation and Privileged States

This subsection fixes the notation for trajectories, adjacent segments, target observations, visibility, and the privileged record, and it is self-contained: every symbol used by the construction rules below is introduced here.

Trajectory, frames, and adjacent segments. An approach trajectory toward a static goal g is

$$\mathcal{T}_g = \{(I_t, \tau_t)\}_{t\in\mathcal{F}}, \qquad \mathcal{F} = \{0, \dots, T-1\},$$

where I_t is the monocular RGB observation from the fixed left camera at timestamp τ_t, and $\mathcal{F}$ indexes all T source frames. Adjacent motion segments are indexed by $\mathcal{S} = \{0, \dots, T-2\}$, so segment t joins frames t and $t+1$.

Privileged record. The synchronized privileged record is

$$\mathcal{Z}(\mathcal{T}_g) = \big(\mathcal{X}, \mathcal{M}^g, \mathcal{O}^g, \mathcal{C}\big), \qquad \begin{aligned} \mathcal{X} &= \{(x_t, y_t, \theta_t, \tau_t)\}_{t\in\mathcal{F}}, \\ \mathcal{M}^g &= \{(b_t^g, z_t^g)\}_{t\in\mathcal{V}}, \\ \mathcal{O}^g &= \{o_t^g\}_{t\in\mathcal{F}}, \\ \mathcal{C} &= (\mathsf{K}, {}^{B}\mathsf{T}_{C}). \end{aligned}$$

Here (x_t, y_t, θ_t) is the trajectory-local planar pose with heading θ_t, b_t^g is the annotated target bounding box, z_t^g is its optical-axis stereo depth, $o_t^g \in \{0, 1\}$ is the visibility indicator, $\mathcal{V} = \{t \in \mathcal{F} : o_t^g = 1\}$ is the set of visible frames, K holds the fixed left-camera intrinsics, and ${}^{B}\mathsf{T}_{C}$ is the camera-to-base extrinsic transform. An explicitly empty bounding-box annotation denotes target invisibility and is distinguished from a missing annotation record.

Stereo-derived depth. We estimate disparity from synchronized, rectified stereo image pairs using S^2M^2-XL (+2Mtr) (Min et al., 2025) and convert the estimates to metric optical-axis depth using the stereo calibration. For each visible target, the depth at the center of its annotated bounding box defines z_t^g. Right-camera images, disparity estimates, and reconstructed depth are used only for reference construction and validation; evaluated models receive left-camera RGB observations only.

Ego-motion quantities. For adjacent frames t and $t+1$, the planar displacement, elapsed time, translational speed, and wrapped heading change are

$$\begin{aligned} \Delta s_t &= \sqrt{(x_{t+1} - x_t)^2 + (y_{t+1} - y_t)^2}, \quad \Delta\tau_t = \tau_{t+1} - \tau_t, \\ v_t &= \frac{\Delta s_t}{\Delta\tau_t}, \qquad\qquad\qquad\qquad \Delta\theta_t = \mathrm{wrap}(\theta_{t+1} - \theta_t), \qquad t \in \mathcal{S}, \end{aligned}$$

with $\mathrm{wrap}(\cdot)$ mapping an angle into $(-\pi, \pi]$. The sign convention is fixed across releases: positive heading change denotes a left turn and negative heading change denotes a right turn. The trajectory-level mean speed and the peak translation and rotation segments are

$$\bar{v} = \frac{\sum_{t\in\mathcal{S}} \Delta s_t}{\sum_{t\in\mathcal{S}} \Delta\tau_t}, \qquad t_v^\star = \arg\max_{t\in\mathcal{S}} v_t, \qquad t_\theta^\star = \arg\max_{t\in\mathcal{S}} |\Delta\theta_t|.$$

Mean Translational Speed Estimation queries $\bar{v}$; Peak Translational Speed Interval Localization queries $t_v^\star$; and the two peak yaw tasks characterize and localize $\Delta\theta_{t_\theta^\star}$.

Target back-projection and camera-to-target range. For a visible target, let $\mathsf{c}_t^g = (u_t^g, v_t^g)$ be the center of its annotated bounding box. Back-projecting its optical-axis depth under the calibrated pinhole-camera model (Hartley & Zisserman, 2004) gives

$$\mathsf{p}_{C,t}^g = z_t^g \mathsf{K}^{-1} \begin{bmatrix} u_t^g \\ v_t^g \\ 1 \end{bmatrix}, \qquad r_t^g = \|\mathsf{p}_{C,t}^g\|_2.$$

Target Depth Estimation queries the optical-axis depth z_t^g, whereas Approach Progress Reasoning uses the Euclidean camera-to-target range. For visible endpoints $t_a < t_b$, the signed range change and the traveled path length are

$$\Delta r_{a:b}^g = r_{t_a}^g - r_{t_b}^g, \qquad L_{a:b} = \sum_{t=t_a}^{t_b-1} \Delta s_t.$$

Positive $\Delta r_{a:b}^g$ indicates progress toward the goal and negative $\Delta r_{a:b}^g$ indicates increasing separation. The traveled path length $L_{a:b}$ is never the reference answer for this task, even when the query discloses it.

World-frame transform and lost-target bearing. The final visible target point is expressed in the trajectory-local world frame through the camera, robot-base, and trajectory-world transforms:

$$\begin{bmatrix} \mathsf{p}_W^g \\ 1 \end{bmatrix} = {}^W\mathsf{T}_{B(t_r)}\, {}^B\mathsf{T}_C \begin{bmatrix} \mathsf{p}_{C,t_r}^g \\ 1 \end{bmatrix},$$

where t_r is the last visible frame before a closed occlusion episode and ${}^W\mathsf{T}_{B(t_r)}$ is the robot-base pose at t_r in the trajectory-local world frame. Let (p_x^g, p_y^g) denote the planar components of p_W^g. Its bearing relative to the robot at an occluded query frame t is

$$\phi_t = \mathrm{wrap}\big(\mathrm{atan2}(p_y^g - y_t, p_x^g - x_t) - \theta_t\big).$$

Positive and negative values place the target to the left and right of the robot's forward axis, and $|\phi_t|$ determines the angular category using the bins in Table 6. The model receives only the causal prefix ending at t; later observations, including target reappearance, are excluded from its input and used only for offline consistency auditing. These privileged quantities define the reference answer and audit fields, but they are never inserted into the model-visible message.

Model-visible queries and reference answers. For task k, let $(s_1, \ldots, s_m)$ index the selected observations in their task-specified display order, and let q_k include the question, answer choices where applicable, and any explicitly stated task parameters. The model prediction and reference answer are

$$\hat{a}_k = f(I_{s_1}, \ldots, I_{s_m}, q_k), \qquad a_k = \rho_k(\phi_k(\mathcal{Z}(\mathcal{T}_g)); \xi_k).$$

Here ϕ_k extracts the diagnostic quantity, ξ_k specifies task parameters, and ρ_k maps the quantity to the reference answer. The displayed observations may form a chronological sequence, a local window, a shuffled set, or a causal prefix, as specified by the task. Privileged records remain hidden, although the question may explicitly state a threshold, path length, or time interval (Appendix D.3).

B.3 Capability-Aligned Construction and the Fourteen Task Types

The fourteen task types are instantiated by deterministic construction rules grouped under the five diagnostic capabilities introduced in Section 3.2. This organization aligns benchmark construction with the capability taxonomy while allowing each task to be regenerated and validated independently.

The locked benchmark contains all fourteen task types.

B.4 Sampling, Thresholds, and Discretization

The generators use fixed thresholds and intervals so that the answer space is stable across models. Table 6 lists the complete numeric boundaries and the task-specific endpoint conventions, together with exact frame-index answer rules.

- Peak Yaw-Change Characterization combines left/right direction with three magnitude bins: at most 30°, strictly between 30° and 50°, and at least 50°. These give six answer categories; there is no separate no-turn category.
- A Yaw-Change Sequence Reconstruction event is significant only when its absolute angle is strictly greater than the turn threshold $\epsilon_\theta = 15^\circ$, that is, when $|\Delta\theta_t| > \epsilon_\theta$. Consecutive same-direction significant segments are merged before the sequence answer is formed, so the answer reports the temporally ordered sequence of salient turn events rather than every adjacent-frame rotation.
- Mean Translational Speed Estimation uses $0.05\,\mathrm{m/s}$ bins. In Speed Variation Classification, let $t_{\max} = \arg\max_{t\in\mathcal{S}} v_t$ and $t_{\min} = \arg\min_{t\in\mathcal{S}} v_t$ over the segment speeds defined in Appendix B.2. A trajectory receives the stable label when the global speed spread does not exceed $\epsilon_v = 0.1\,\mathrm{m/s}$; otherwise $t_{\max} < t_{\min}$ yields an accelerate–decelerate pattern and $t_{\min} < t_{\max}$ yields a decelerate–accelerate pattern. This rule describes the time order of the global extrema only. It does not assert segment-wise monotonicity, and a trajectory may violate it locally while still receiving the corresponding global-extrema label.
- Target Depth Estimation intervals have width $0.15\,\mathrm{m}$ and are anchored at zero; the evaluated option ladder starts at $0.15\,\mathrm{m}$. Equality with a lower boundary belongs to that interval; equality with an upper boundary belongs to the next interval.
- Approach Progress Reasoning compares the Euclidean camera-to-target range at the first and fourth frames of a four-frame observation. For the evaluated cohort, $\Delta r = r_{t_a} - r_{t_b} \geq 0$, where positive values denote a reduction in range. The four answer bins are $[0, 0.10)$, $[0.10, 0.20)$, $[0.20, 0.30)$, and $[0.30, \infty)$ m; each lower boundary is inclusive and each finite upper boundary is exclusive.
- For Approach Target Frame Ordering, source indices begin at zero. Trajectories with at most 15 source frames use stride 2; longer trajectories use stride 3. An off-grid final frame is not forcibly appended. Sampled frames are shuffled with a stable per-trajectory seed and neutral labels are assigned after shuffling.
- Lost-Target Relative Bearing Estimation discretizes the absolute bearing into 25° bins paired with left/right; the displayed option set uses the two side bins 50°–75° and 75°–100°, giving four answer cells.

These values are construction parameters rather than model inputs. They are recorded with the dataset tag and generator summaries so that a frozen paper snapshot can be distinguished from later releases.

B.5 Frame Selection and Reference Ordering

For a source trajectory containing T chronologically indexed frames, as defined in Appendix B.2, define the sampling stride $h(T)$ and the sampled source-index set $\mathcal{P}$ by

$$h(T) = \begin{cases} 2, & T \leq 15, \\ 3, & T > 15, \end{cases} \qquad \mathcal{P} = \{0, h, 2h, \ldots, \lfloor (T-1)/h \rfloor h\}. \tag{1}$$

Thus the sampling interval is a fixed number of source-frame indices, rather than a fixed wall-clock duration. No off-grid final frame is appended. At least two selected frames are required. The fixed base seed is 20260804. For each trajectory, SHA-256 hashes the base seed and the identity string consisting of the task name, trajectory identifier, and `frame-order` suffix; its first 16 bytes, interpreted as a big-endian integer, seed the random shuffle. If the shuffle equals the original order, the first two entries are swapped. Neutral display labels are assigned after shuffling. The reference answer lists these labels in ascending source-time order. Since this task uses only trajectories manually categorized as monotonic approaches, the ascending source-time order provides the farther-to-nearer reference; the reference is therefore the recorded source chronology, not an ordering derived from re-sorting measured depths, and monotonicity is a selection criterion for this task rather than a condition imposed on the rest of the benchmark.

B.6 Multiple-Choice Option Construction

For a multiple-choice item, the generator first computes the semantic answer from the privileged trajectory record, then forms the task-specific option set. Options for four task types are drawn from

Table 5: Capability-aligned construction map for the fourteen task types.

Capability	Task types	Privileged inputs	Core construction rule
Approach-Goal Grounding	Approach Target Identification	Human target boxes and labels	Require a unique, trajectory-consistent designated goal and render its semantic identity as the reference answer.
Approach-Progress Spatial State Inference	Target Depth Estimation Approach Progress Reasoning Approach Target Frame Ordering	Target geometry, stereo depth, odometry, timestamps, and camera intrinsics	Recover optical-axis depth and range change; sample and permute chronological frames for farther-to-nearer reconstruction.
Approach-Motion Spatial Understanding	Mean Translational Speed Estimation Peak Translational Speed Interval Localization Speed Variation Classification Peak Yaw-Change Characterization Peak Yaw-Change Localization	Trajectory-local poses, yaw, and timestamps	Compute adjacent translation and rotation, aggregate speed, locate extrema, and map magnitude and temporal structure to task-specific answers.
Approach-Trajectory Spatial Integration	Yaw-Change Sequence Reconstruction	Trajectory-local yaw and timestamps	Extract salient signed turn events, merge consecutive same-direction events, and preserve their temporal order.
Target Visibility and Invisible-Target Bearing	Target Loss Onset Target Loss Offset Target Loss Duration Lost-Target Relative Bearing Estimation	Visibility annotations, target geometry, odometry, and camera-to-base extrinsics	Instantiate closed visibility episodes, use causal prefixes for bearing, and reserve later reappearance for validation.

Table 6: Explicit answer intervals and temporal boundaries. Scalar bins are specified with their endpoint convention. Temporal answers identify source frame indices or adjacent-frame pairs rather than discretized seconds.

Task	Boundaries, units, and answer rule
Target Depth Estimation	Optical-axis depth in meters. The admissible 0.15 m option ladder has boundaries 0.15, 0.30, 0.45, 0.60, 0.75, 0.90, 1.05, 1.20, 1.35, 1.50, 1.65, 1.80, 1.95, 2.10, 2.25, 2.40, 2.55, 2.70, 2.85, 3.00, 3.15. Bins are $[0.15k, 0.15(k+1))$ for $k \in \{1, \ldots, 20\}$.
Approach Progress Reasoning	Compare Euclidean camera-to-target range at the first and fourth frames of the four-frame input, with $\Delta r = r_{t_a} - r_{t_b}$ in meters. In the evaluated cohort, $\Delta r \geq 0$ and the four bins are $[0, 0.10)$, $[0.10, 0.20)$, $[0.20, 0.30)$, and $[0.30, \infty)$. Lower bounds are inclusive; finite upper bounds are exclusive.
Mean Translational Speed Estimation	Meters per second: $[0, 0.05]$, $(0.05, 0.10]$, $(0.10, 0.15]$, $(0.15, 0.20]$, $(0.20, 0.25]$. Zero is included in the first bin; the upper boundary belongs to its bin. The higher 0.25–0.30 m/s bin and items whose reference speed falls in it are excluded from this cohort.
Lost-Target Relative Bearing Estimation	Absolute bearing in degrees, discretized in 25° bins paired with left/right; the displayed option set is the four cells $[50, 75)$ and $[75, 100)$ on each side. A positive signed bearing is left, as is zero; a negative bearing is right.
Target Loss Duration	Exact invisible-frame count $D_{\mathrm{occ}} = t_a - t_\ell$, where t_ℓ is the first invisible frame and t_a is the first subsequent visible frame. The count is paired with a percentage bin for $p = 100 D_{\mathrm{occ}}/T$. Boundaries (%) are 0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100. Bins are $[5k, 5(k+1))$; a closed visibility episode has $0 < p < 100$.
Peak Translational Speed Interval Localization / Peak Yaw-Change Localization	Adjacent source-frame pairs $(j, j+1)$, one for every integer $j \in \{0, \ldots, T-2\}$. The answer identifies the pair containing the maximum translational speed or maximum absolute yaw change. The pair notation names two frames; it is not an open continuous-time interval. A tie selects the earliest pair.
Target Loss Onset / Target Loss Offset	Exact frame indices in $\{0, \ldots, T-1\}$; no temporal binning. Loss onset is the first invisible frame and loss offset is the first subsequent visible frame.

a shared answer ladder: target depth, peak-speed interval, peak-yaw interval, and target-loss duration. For the bin- and count-based ladders, a consecutive window containing the reference bin is selected uniformly among its feasible starting positions. The two interval-localization tasks instead pair the reference adjacent-frame interval with its two immediate neighbours and one further interval, so their candidates are anchored on the reference interval and need not be contiguous. Lost-target bearing uses a fixed question-independent four-cell ladder that pairs a side with a magnitude range, so the same four cells appear in every item. AppProg instead compares the Euclidean camera-to-target range at the first and fourth frames of a four-frame observation. Let $\Delta r = r_{t_a} - r_{t_b}$, so positive values indicate a shorter range. Its fixed option set comprises a reduction below 0.10 m and three further reduction bins: $[0.10, 0.20)$, $[0.20, 0.30)$, and at least 0.30 m. The evaluated 700-instance cohort is balanced across the four reference bins (175 instances each), as well as across the four correct-option letters (175 each). The bearing ladder orders left and right categories across the forward axis.

The other four choice tasks use fixed answer sets or sequence-derived distractors. Mean-speed estimation displays the same five 0.05 m/s bins in every item, from 0.00–0.05 through 0.20–0.25 m/s. The higher 0.25–0.30 m/s bin and items with that reference answer are excluded; the retained items are sampled across the five reference bins without changing their boundaries. Speed-variation classification uses three fixed categories, and peak-yaw characterization uses six direction–magnitude combinations. Turn-sequence reconstruction combines the reference sequence with an explicit no-turn option and three distinct edited sequences.

Option order is shuffled with a fixed seed, and the answer letter is assigned after shuffling. Each record retains the semantic answer in `answer_text`, the option-independent `answer_value` when applicable, and the privileged fields needed to audit the construction.

The four non-choice task types continue to use their dedicated time-index, frame-ordering, or semantic-identity output contracts. Construction rules for all ten choice tasks are summarized in Table 16.

B.7 Determinism and Reproducibility

Question sampling, frame shuffling, and option ordering use fixed seeds. Question identifiers are unique, and each record retains the reference answer, task parameters, and privileged ground truth needed for independent auditing. The evaluation freezes question wording and answer spaces. Model-specific output adapters preserve the semantic task. The released benchmark artifact records the prompt, parser, and evaluated-model configuration used for each run so that scoring can be reproduced.

B.8 Validation and Quality Control

Candidate records pass automated validation before release. The checks verify whitelist membership, source-file availability, finite timestamps, valid segment chains, image-path existence, and task-specific requirements such as a unique target annotation or a closed visibility event. For multiple-choice records, the validator checks option-label order, text uniqueness, answer-label alignment, and membership of the ground truth in the declared interval or threshold. For frame-order records, it checks that every displayed frame appears exactly once in the answer and that the shuffle is not the identity permutation. For invisible-target bearing, it additionally checks world-coordinate consistency of the independently estimated target position, while keeping later reappearance outside the model input.

The final dataset is also checked for valid JSON, shard-size limits, unique question IDs, and consistency between `answer`, `answer_text`, and `answer_value`.

B.9 Reliability of Sensor-Derived References

Stereo-derived reference geometry. We use S^2M^2-XL (+2Mtr) (Min et al., 2025) to estimate dense disparity from synchronized, rectified stereo pairs and convert it to metric optical-axis depth using stereo calibration. These estimates provide pseudo-ground-truth geometry for reference-answer construction. For each visible target, depth is sampled at the center of its annotated bounding box, as

specified in Appendix B.2. Right-camera images, disparity estimates, and reconstructed depth remain privileged construction data; evaluated MLLMs receive only left-camera RGB observations.

Quantitative validation. We evaluate S^{2}M^2-XL (+2Mtr) on the 15 full-resolution stereo pairs in the Middlebury v3 training split (Scharstein et al., 2014; Scharstein & Hirschmüller, 2015) using the dataset-provided ground truth, with FP16 inference and three refinement iterations. Figure 4 reports disparity endpoint error (EPE) over all valid ground-truth pixels and the non-occluded subset, stratified by ground-truth depth. For S^{2}M^2-XL (+2Mtr), the bin-level EPE remains below 1.2 pixels over all valid pixels and below 0.7 pixels over non-occluded pixels across the five plotted depth intervals spanning 0.5–10 m. The first and last intervals each contain pixels from only one stereo pair and should therefore be interpreted as scene-specific observations.

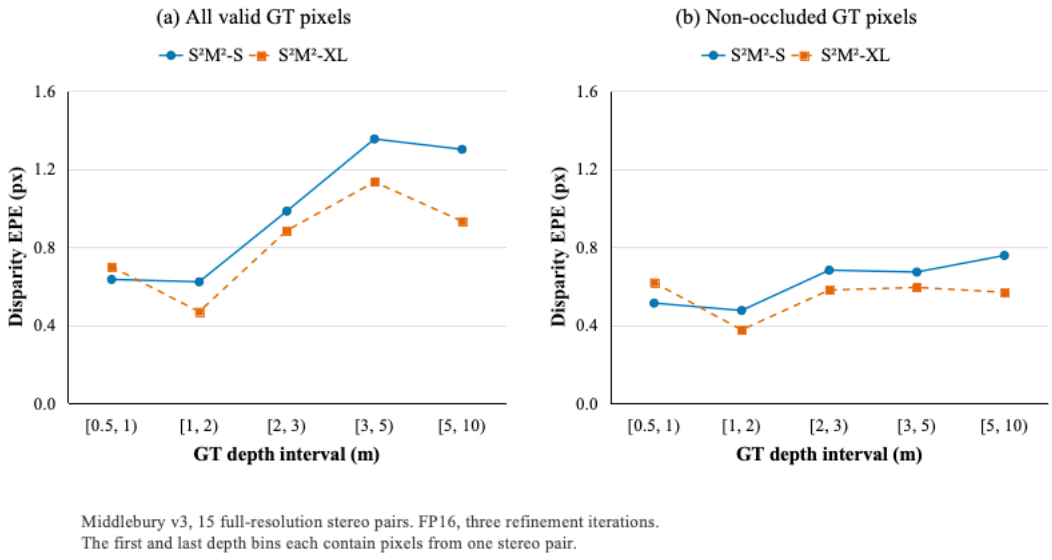


Figure 4: Disparity endpoint error (EPE) for S^{2}M^2-S and S^{2}M^2-XL (+2Mtr) on Middlebury v3 across five ground-truth depth intervals, shown for all valid pixels (left) and non-occluded pixels (right).

Scope and sensing limitations. Dense, image-aligned reconstruction, together with quantitative validation and the geometric consistency checks described in Appendix B.8, supports our use of stereo-derived reference geometry. LiDAR is also subject to material-dependent sensing limitations: transparent glass can transmit laser pulses, while mirrors and highly specular surfaces, including polished metal, can redirect returns and cause missing measurements or reflection artifacts (Foster et al., 2013; Gao et al., 2021). LiDAR acquisition therefore does not automatically guarantee complete, artifact-free geometry. Our reconstructed references remain pseudo-ground truth; the present evaluation neither establishes superiority over LiDAR nor specifically validates accuracy on transparent or reflective materials.

B.10 Trajectory-Local Pose References

During teleoperated data collection, the onboard localization pipeline fuses measurements from an LD19 2D LiDAR and a Bosch BMI270 IMU. The resulting trajectory-local planar poses (x_t, y_t, θ_t) are recorded throughout each approach and associated with RGB frames by timestamp. Reference construction uses these recorded pose estimates rather than commanded poses; the pose records remain privileged and are not provided to the evaluated models.

The pose-derived motion references depend on changes within a trajectory rather than its absolute placement in a global map. Writing $\mathsf{p}_t = (x_t, y_t)^\top$, the adjacent-frame translation and heading changes are

$$\Delta\mathsf{p}_t = \mathsf{p}_{t+1} - \mathsf{p}_t, \qquad \Delta\theta_t = \mathrm{wrap}(\theta_{t+1} - \theta_t).$$

As defined in Appendix B.2, the displacement magnitude $\Delta s_t = \|\Delta \mathsf{p}_t\|_2$ and elapsed time determine segment speed, while the signed heading changes determine turn magnitudes, peak-turn intervals, and salient turn sequences.

To distinguish shared offsets from errors in these relative quantities, consider

$$\mathsf{p}_t = \mathsf{p}_t^\star + \mathsf{b} + \boldsymbol{\epsilon}_t, \qquad \theta_t \equiv \theta_t^\star + b_\theta + \epsilon_{\theta,t} \pmod{2\pi},$$

where starred quantities denote the underlying true poses in the same coordinate frame, b and b_θ are time-invariant offsets, and $\boldsymbol{\epsilon}_t$ and $\epsilon_{\theta,t}$ are time-varying estimation errors. The corresponding increment errors satisfy

$$\Delta \mathsf{p}_t - \Delta \mathsf{p}_t^\star = \boldsymbol{\epsilon}_{t+1} - \boldsymbol{\epsilon}_t,$$
$$\mathrm{wrap}\big(\Delta\theta_t - \Delta\theta_t^\star\big) = \mathrm{wrap}\big(\epsilon_{\theta,t+1} - \epsilon_{\theta,t}\big).$$

Thus, a shared position offset cancels in the translation difference, and a shared heading offset cancels in the heading difference. The relevant uncertainty is therefore the error in relative motion over the intervals used for reference construction, including its propagation through speed and path-length calculations.

For lost-target bearing, the pose-dependent quantity is the relative transform between the last visible frame and the invisible query frame, combined with the last-visible target geometry and camera-to-base calibration (Appendix B.2). The relevant motion interval can therefore span multiple frames. Appendix B.11 relates perturbations in the resulting reference quantities to answer-boundary margins and the separation between competing extrema.

B.11 Discretization and Reference-Label Stability

The option intervals in Tables 6 and 16 coarsen continuous reference quantities. For a scalar reference value q, let $m(q)$ be its distance to the nearest boundary that changes its answer. If the total perturbation of q is strictly smaller than $m(q)$, its binned answer is unchanged. The bin width alone is insufficient for this test: an item arbitrarily close to a boundary may change labels under a much smaller perturbation. Table 7 applies this criterion to the measurement-derived multiple-choice tasks. Here e_z, $e_{r,t}$, $e_{\theta,t}$, e_v, and e_ϕ denote item-level absolute error bounds on optical-axis depth, endpoint range, pose yaw, mean speed, and lost-target bearing, respectively; the table states sufficient conditions, not measured bounds on these errors. We use e_{seg} for a uniform absolute error bound on each segment speed.

Tables 8 to 11 report observed ground-truth margins for target depth, endpoint range change, mean translational speed, and lost-target bearing. In the T-Depth, MeanSpd, and LostBear tables, A–E identify correct-answer letters after option shuffling; they are not fixed numeric bins.

For Approach Progress Reasoning, we compute $m(\Delta r) = \min_{b \in \{0.10, 0.20, 0.30\}\,\mathrm{m}} |\Delta r - b|$. The first bin's margin is measured to 0.10 m; the separate condition that Δr remain nonnegative still applies.

Mean-speed questions contain at least 8 frames, 10.32 frames on average, and at most 32 frames. With fixed timestamps, the definition $\bar{v} = L/D$, where $L = \sum_t \Delta s_t$ and $D = \sum_t \Delta \tau_t$, converts a speed-bin margin $m(\bar{v})$ into an allowable accumulated path-length perturbation of $m(\bar{v})D$; D is computed from timestamps rather than frame count. At the smallest reported margin, 2.0160 cm/s, an 8 s elapsed trajectory has a 16.128 cm path-length margin. Accordingly, the relevant position-related uncertainty is the perturbation accumulated from the displacement increments used to compute L, rather than a final-position error alone.

The three turn tasks use adjacent-frame heading differences rather than absolute heading, so a shared yaw offset cancels. Peak Yaw-Change Localization compares change magnitudes rather than angular rates or binned angles; Yaw-Change Sequence Reconstruction uses a 15° salience threshold and turn directions rather than magnitude bins. Let η_θ denote a hypothetical uniform bound on the absolute error in each wrapped heading change. A gap greater than $2\eta_\theta$ between the largest and second-largest change magnitudes preserves the peak-turn location. Turn-sequence membership is preserved when every segment's magnitude is more than η_θ from the 15° salience threshold, with the directions of retained turns unchanged. These are sufficient stability conditions, not measured error bounds.

Table 7: Answer granularity and sufficient conditions for reference-label stability. The conditions use each item's margin to the nearest answer-changing boundary; nominal interval width by itself does not certify stability.

Task	Answer granularity	Sufficient item-level condition
Target Depth Estimation	0.15 m depth intervals	$m(z_t^g) > e_z$. A hypothetical 0.01 m depth-error bound would protect only items more than 0.01 m from a depth boundary.
Approach Progress Reasoning	0.10 m steps at 0.10, 0.20, and 0.30 m	$m(\Delta r) > e_{r,t_a} + e_{r,t_b}$, with Δr also remaining nonnegative. If each endpoint *range* error were at most 0.01 m, a margin greater than 0.02 m would suffice.
Mean Translational Speed Estimation	0.05 m/s intervals	$m(\bar{v}) > e_v$; the pose-position error must first be propagated through displacement and elapsed time to bound e_v.
Speed Variation Classification	0.10 m/s speed-spread threshold and the temporal order of global speed extrema	If every segment-speed error is at most e_{seg}, the spread stays on the same side of 0.10 m/s when its threshold margin exceeds $2e_{\text{seg}}$. For a non-stable label, the gaps to the competing maximum and minimum segments must also exceed $2e_{\text{seg}}$ to preserve their temporal order.
Peak Yaw-Change Characterization	Left/right direction and magnitude boundaries at 30° and 50°	The peak segment must remain the maximum, and its signed yaw change must stay on the same side of the direction and magnitude boundaries.
Peak Yaw-Change Localization	Adjacent-frame pair with the largest $\lvert\Delta\theta_t\rvert$; no angular bin	The gap between the largest and second-largest $\lvert\Delta\theta_t\rvert$ must exceed the sum of their perturbation bounds. A shared yaw offset cancels in each adjacent-frame difference.
Yaw-Change Sequence Reconstruction	Turns with $\lvert\Delta\theta_t\rvert > 15^\circ$, merged by direction	Each segment must remain on the same side of 15°; the directions of retained turns must remain unchanged. Their temporal order is then unchanged.
Lost-Target Relative Bearing Estimation	The four fixed cells 50°–75° and 75°–100°, on each side	$m(\phi_t) > e_\phi$, including the left/right boundary. The bearing error also depends on depth back-projection and both robot poses.
Peak Speed Localization	Adjacent-frame index, not a speed bin	The gap between the two largest segment speeds must exceed the sum of their error bounds.

Table 8: Target Depth Estimation: ground-truth distance to the nearest depth-bin boundary, grouped by correct-answer letter.

Correct answer	Minimum margin (cm)	Median margin (cm)
A	3.10	5.50
B	3.10	5.20
C	3.10	5.20
D	3.10	5.15

Table 9: Approach Progress Reasoning (700 questions): ground-truth distance to the nearest range-change bin boundary, grouped by reference bin.

Reference bin	Minimum margin (cm)	Median margin (cm)
$[0, 0.10)$ m	2.330	6.222
$[0.10, 0.20)$ m	2.011	3.944
$[0.20, 0.30)$ m	2.005	4.207
$[0.30, \infty)$ m	2.076	7.198

Table 10: Mean Translational Speed Estimation: ground-truth distance to the nearest speed-bin boundary, grouped by correct-answer letter.

Correct answer	Minimum margin (cm/s)	Median margin (cm/s)
A	2.0291	2.2820
B	2.0439	2.2206
C	2.0395	2.3215
D	2.0160	2.3215
E	2.0318	2.2516

Table 11: Lost-Target Relative Bearing Estimation: ground-truth distance to the nearest bearing-bin boundary, grouped by correct-answer letter.

Correct answer	Minimum margin (°)	Median margin (°)
A	5.11	9.43
B	5.12	8.51
C	5.16	8.65
D	5.20	7.93

B.12 Text-Only Ablation

To assess whether question and option text provide a shortcut to the reference answer, we evaluate text-only copies of the ten multiple-choice task types. Each copy preserves the question, options, reference answer, and scoring rule of its image-based counterpart while omitting all images. We evaluate Qwen3.8-Flash zero-shot with temperature zero and disabled thinking. All 6,691 questions yield scored responses. The AppProg cohort contains 700 questions, balanced across its four reference bins and correct-option letters; the other task counts are unchanged. The chance level for each question is $1/k$, where k is its number of options; task and overall chance levels average this quantity over the corresponding questions.

Table 12 shows 20.65% aggregate accuracy against a 24.25% nominal chance level. T-Depth (23.14%) and YawLoc (23.37%) lie near their 25% random baseline; MeanSpd (25.29%) is numerically in the same range, although its five-option random baseline is 20%. PeakYaw (11.75% versus 16.67%) and YawSeq (10.93% versus 20%) instead fall below chance. In the absence of images, the question text systematically steers the model toward wrong answers rather than eliciting random guesses: PeakYaw selects a turn of at most 30° in 332 of 519 cases, although only 73 references have that magnitude; YawSeq selects "no salient turn" in 87 of 613 cases, although none has that reference answer.

Eight of ten tasks do not exceed their nominal chance levels, offering no evidence of a general text-only answer shortcut. AppProg (25.71% versus 25%) and MeanSpd (25.29% versus 20%) are the only above-chance point estimates; the AppProg result is close to its nominal baseline. The ablation therefore does not establish that every task is free of text-based cues.

C Controlled Bearing Diagnostics

These analyses examine answers to invisible-target bearing queries under controlled changes in available observations and query times. In the initial fixed-query design, distractor windows were anchored at the ground-truth value, making the correct option's position predictable and allowing the *text-only* condition to exceed the random baseline without visual evidence. For the Lost-Target Relative Bearing diagnostic, the revised design removes this prior by decoupling the option set from the ground truth. This appendix describes the observation conditions, option construction, and a worked question–option–answer example.

Table 12: Text-only multiple-choice diagnostic on Qwen3.8-Flash. Questions, options, and reference answers match the image-based evaluation; no images are supplied. Accuracy uses exact match, and chance is averaged from $1/k$ per question.

Task ID	n	k	Correct	Accuracy (%)	Random (%)
`T2_Target_Depth_Estimation`	700	4	162	23.14	25.00
`T3_Approach_Progress_Reasoning`	700	4	180	25.71	25.00
`T5_Mean_Translational_Speed_Estimation`	700	5	177	25.29	20.00
`T6_Peak_Translational_Speed_Interval_Localization`	700	3–4	127	18.14	25.02
`T7_Speed_Variation_Classification`	700	3	188	26.86	33.33
`T8_Peak_Yaw-Change_Characterization`	519	6	61	11.75	16.67
`T9_Peak_Yaw-Change_Localization`	659	4	154	23.37	25.00
`T10_Yaw-Change_Sequence_Reconstruction`	613	5	67	10.93	20.00
`T13_Target_Loss_Duration`	700	4	152	21.71	25.00
`T14_Lost-Target_Relative_Bearing_Estimation`	700	4	114	16.29	25.00
Overall	6,691	—	1,382	20.65	24.25

C.1 Task Types and Observation Conditions

The fixed-query diagnostic evaluates Lost-Target Relative Bearing under four observation conditions. For a given question, all conditions share the wording, reference answer, and option set; only the supplied observations vary. The text-only condition receives no images. The image-bearing conditions provide prefixes of the recorded sequence with different endpoints relative to the fixed query time:

- Lost-Target Relative Bearing Estimation (`invisible_target_relative_bearing`, 516 questions $\times$ 4 conditions $=$ 2,064 records): image-bearing conditions use a fixed query at an invisible frame t and vary in how far the observation sequence extends. `text_only` supplies no images; `causal_prefix` ends at the query frame and is the causal prefix; `pre_reappearance_prefix` extends to the last frame before the target reappears; and `post_reappearance_prefix` extends past the reappearance. The query asks for the target's bearing relative to the robot's forward axis at t.

C.2 Leak-Free Option Construction

For a multiple-choice item over k options, the option set F is *decoupled* from the ground truth g when (i) the sampling of F does not depend on g, and (ii) the ground truth is conditionally uniform given the option set, $P(G = g \mid F) = 1/k$. The fixed-query protocol uses a fixed, question-independent four-cell bearing ladder, samples equally from its four reference classes, and balances answer-letter assignment per question. This gives uniform reference-class and answer-letter marginals within each of its four observation conditions. The same-observation diagnostic in Appendix C.5 uses the same four semantic cells but chooses one option order per episode and holds it fixed across its three queries. Its answer letters are balanced by query stage; its reference-class frequencies follow the selected trajectories.

Both diagnostics retain questions whose reference bearing falls in one of these $50°$–$100°$ side bins. Restricting the ladder to these four cells avoids an empty-bin cue and makes the answer a side $\times$ magnitude judgment.

Table 13: Four-cell option ladder used by both bearing diagnostics. Option order is assigned per question for the fixed-query study and per episode for the same-observation study.

Task type	Option-construction rule (English rendering)
Lost-Target Relative Bearing Estimation	"left, about $50°$–$75°$"; "left, about $75°$–$100°$"; "right, about $50°$–$75°$"; "right, about $75°$–$100°$".

C.3 Question–Option–Answer Examples

The example below is an English rendering of a benchmark record; the released prompt is written in Chinese. It shows the question, the fixed option ladder, and the reference answer, with the correct option in bold.

Lost-Target Relative Bearing Estimation (`text_only` condition). *Question.* "Observe the image sequence in its original chronological order. The navigation target 'brown cloth' is invisible at t=17. Regardless of whether the input contains additional observations after t=17, judge the target's direction and angular range relative to the robot's forward axis at t=17; do not substitute the timestamp of the last image for the query timestamp." *Options.* A. left, about $75°$–$100°$; B. right, about $50°$–$75°$; C. right, about $75°$–$100°$; D. left, about $50°$–$75°$. *Answer.* `D` (left, about $50°$–$75°$).

C.4 Evaluation Protocol

The diagnostic is evaluated on Qwen3.8-Flash under a zero-shot protocol with a temperature of zero. Accuracy is the percentage of exact matches with the reference option label. Because the text-only condition shares the question and option set with image-bearing conditions but receives no images, paired comparisons use the exact McNemar test on per-question disagreements, which is more sensitive than the single-sample binomial test against the 25% random baseline.

One caveat affects how the text-only condition should be read. On a balanced option set, a model that collapses to a constant answer also scores exactly $1/k$, which is indistinguishable from "no information" on accuracy alone. We therefore report two additional diagnostics per condition: the maximum share of a single predicted *content* (after mapping each predicted letter back to its option text, since letters are re-randomized per question), and the accuracy broken down by ground-truth class. A condition is flagged as degenerate when one predicted content accounts for at least 80% of answers or when one ground-truth class is answered nearly always and the others nearly never. Degenerate text-only conditions should be read as $acc - 1/k$ rather than acc itself.

C.5 Same Observation, Different Query Times

A further diagnostic examines answers to different query times under identical observations. We retain 64 occlusion episodes, one per trajectory, with three distinct invisible query times: first loss, an intermediate frame, and the last invisible frame before reappearance. The three questions in each episode share byte-identical observations and one ordering of the four options in Table 13; only the query time changes. Each reference bearing is more than $5°$ from the nearest option-bin boundary. Correct answer letters are balanced separately at each query stage (16 per letter), while the four reference classes occur 32, 40, 66, and 54 times in the order shown in Table 13.

Table 14: Same-observation, different-query-time diagnostic on Qwen3.8-Flash (64 episodes). Confidence intervals are 95% Wilson intervals.

Measure	Value (%)	95% CI
Accuracy, first-loss query	42.19	30.9–54.4
Accuracy, middle query	39.06	28.1–51.3
Accuracy, last-invisible query	45.31	33.7–57.4
All three queries correct in the same episode	10.94	—

Seven of the 64 episodes have all three queries answered correctly (10.94%). This joint accuracy reports an episode-level result alongside the per-query accuracies. The fixed four-cell option set replaces the earlier ground-truth-anchored distractors. Reference-class frequencies are nevertheless unequal, and the model selects option A in 86 of 192 image-bearing queries. On the matched text-only questions, two runs score $41/192$ and $42/192$, versus $81/192$ with images; the earlier observation that images reduce accuracy does not hold for this cohort.

D Prompt and Evaluation Contract

D.1 Prompt Architecture

The evaluator does not construct a separate hand-written prompt for each model. It loads a versioned JSON prompt specification and renders one task-specific request for each benchmark record. A rendered request consists of a shared system instruction, a task-specific user template, runtime substitutions, and the ordered visual content. The templates use five placeholders:

- `{question}`: the question string stored in the benchmark record;
- `{options}`: newline-separated option labels and texts, or a no-options marker;
- `{image_count}`: the number of images included in the request;
- `{image_labels}`: labels in the order used by the evaluator; and
- `{question_type}`: the canonical task identifier.

The evaluator validates that every task type has an answer mode and a user template before inference. The exact prompt file remains the version-controlled source of truth; this appendix documents its semantics and provides a paper-readable rendering of the shared instruction.

D.2 Shared System Instruction

The shared system instruction is rendered in Chinese in the current prompt file. Its paper-readable English rendering is:

> *You are a vision-language model specialized in evaluating motion understanding from consecutive first-person images of a mobile robot. The images come from a fixed left camera. Internally compare stable scene structures, vanishing points, parallax, perspective, and target-scale change; distinguish robot translation, camera rotation, and local dynamic objects. Do not treat exposure changes, motion blur, or a single moving object as evidence of robot motion. Temporal tasks must use the supplied frame labels exactly, and multiple-choice tasks must follow option content rather than option position. Use only the current question, options, and images, and do not guess unobserved content. You may analyze internally, but the final response must contain no reasoning, explanation, Markdown, or code block—only the answer form requested by the task-specific output contract.*

This instruction directs the model toward stable geometric cues while prohibiting privileged metadata, filename-based shortcuts, and verbose outputs that complicate deterministic scoring.

D.3 Task-Specific Prompt Contract

The following table summarizes the fourteen task templates used in the evaluation snapshot. The frozen prompt specification and its hashes will accompany the public release.

The table makes the information asymmetry explicit. In particular, `target_distance_interval` uses the complete sequence to establish target identity but asks for one specified time, whereas `invisible_target_relative_bearing` excludes the future reappearance by design. `approach_frame_order` is the only task whose displayed images are deliberately permuted; its neutral labels do not reveal source chronology.

Table 15: Task-specific prompt focus, input ordering, and output contract.

Task ID	Prompt focus	Input ordering / output mode
T1_Approach_Target_Identification	Identify the one static object consistently approached by the robot.	Chronological / Free text
T2_Target_Depth_Estimation	Locate the named target using the sequence but estimate depth at one specified t.	Chronological / Choice
T3_Approach_Progress_Reasoning	Compare endpoint target range over a four-frame window using intermediate frames for identity and trend.	Four-frame window / Choice
T4_Approach_Target_Frame_Ordering	Restore target-approach progression from farthest to nearest.	Shuffled sampled frames / Frame sequence
T5_Mean_Translational_Speed_Estimation	Estimate whole-trajectory translation while discounting pure rotation.	Chronological / Choice
T6_Peak_Translational_Speed_Interval_Localization	Localize the adjacent pair with the strongest translational motion.	Chronological / Choice
T7_Speed_Variation_Classification	Apply the 0.1 m/s stability threshold, then compare the temporal order of global speed extrema.	Chronological / Choice
T8_Peak_Yaw-Change_Characterization	Find the largest adjacent-frame ego-turn; predict left/right direction and one of three magnitude bins.	Chronological / Choice
T9_Peak_Yaw-Change_Localization	Compare all adjacent pairs and localize the largest absolute turn.	Chronological / Choice
T10_Yaw-Change_Sequence_Reconstruction	Retain significant turns above 15°, merge same-direction runs, and output their order.	Chronological / Choice
T11_Target_Loss_Onset	Identify the first invisible frame of a specified closed loss event.	Chronological / Time index
T12_Target_Loss_Offset	Identify the first frame in which the target becomes visible again.	Chronological / Time index
T13_Target_Loss_Duration	Count the closed invisible interval and match its percentage range.	Chronological / Choice
T14_Lost-Target_Relative_Bearing_Estimation	Estimate target bearing at an invisible query frame from the causal observation prefix.	Causal prefix / Choice

Option construction for ten task types. Table 16 covers four shared answer ladders, five fixed answer sets, and one sequence-derived option set. Every displayed candidate is anchored on the reference answer. The two bin- and count-based ladders take a feasible consecutive window that contains the reference bin, whereas the two interval-localization tasks (`T6`, `T9`) take the reference interval together with its immediate neighbours and one further interval; lost-target bearing instead uses a fixed four-cell ladder that is identical for every item. The controlled-context diagnostic in Appendix C.2 uses separate option sets.

For all ten rows, the generator validates option-text uniqueness, membership of the reference value, and consistency among the option label, `answer`, and `answer_text`.

D.4 Evaluated Models and Configurations

We evaluate 28 MLLMs, covering 19 open-source models and nine closed-source API models. The open-source group includes Qwen2.5-VL-7B Bai et al. (2025b); the Qwen3-VL Instruct/Thinking variants at 8B, 30B-A3B, 32B, and 235B scales Bai et al. (2025a); InternVL3-1B Zhu et al. (2025); InternVL3.5-8B and InternVL3.5-30B-A3B Wang et al. (2025b); Gemma-3n-E2B Gemma Team (2025); GLM-4.6V-Flash Z.ai (2025); Nemotron-Nano-VL-8B NVIDIA (2025); LLaVA-OneVision-Qwen2-7B Li et al. (2024); RoboBrain2.5-8B Tan et al. (2026); Cosmos-Reason2-8B NVIDIA (2026); and Qwen3.5-397B-A17B. The 397B model uses locally deployed open weights Qwen Team (2026a). The API group comprises Gemini-3.8-Flash Google DeepMind (2026), GPT-5.6-Luna and GPT-5.6-Sol OpenAI (2026), DeepSeek-V4.1-Flash DeepSeek-AI (2026), Qwen3.7-Plus and Qwen3.7-Flash Alibaba Cloud (2026), Qwen3.8-Max Qwen Team (2026c), Qwen3.5-Omni-Plus Qwen Team (2026b), and Kimi-K3 Kimi Team (2026). Provider documentation citations identify the model families; the evaluated endpoint identifiers and inference settings are specified in the evaluation configuration.

Qwen3.8-Flash is additionally used for the supplementary diagnostic experiments reported in this appendix. It is accessed through the DashScope API using a DashScope API key and is not part of the 28-model leaderboard cohort. Its supplementary inference protocol is summarized in Table 17.

Table 16: Option construction for the ten multiple-choice tasks. The reference answer is computed before the option set; option labels follow a seeded shuffle.

Task ID	Options	Option set and selection rule	Reference and boundary details
T2_Target_Depth_Estimation	4	Four consecutive 0.15 m optical-axis depth bins; the window is sampled from feasible positions that contain the reference bin.	The reference interval is left-closed and right-open, with lower bound from 0.15 to 3.00 m on the admissible ladder.
T3_Approach_Progress_Reasoning	4	The same four fixed endpoint-range-reduction categories appear in every item: below 0.10 m, $[0.10, 0.20)$ m, $[0.20, 0.30)$ m, and at least 0.30 m.	Compare Euclidean camera-to-target range at the first and fourth frames of a four-frame observation; $\Delta r = r_{t_a} - r_{t_b} \geq 0$ in this cohort. Bins are lower-inclusive and upper-exclusive at finite boundaries.
T5_Mean_Translational_Speed_Estimation	5	All five fixed-width 0.05 m/s bins from 0.00–0.05 through 0.20–0.25 m/s appear in every item.	The rarely occupied 0.25–0.30 m/s bin and items with that reference bin are excluded. Retained items are sampled across the five reference bins.
T6_Peak_Translational_Speed_Interval_Localization	3–4	The maximum-speed interval on the $t = 0, \ldots, T-2$ ladder, its two immediately neighbouring intervals, and one further interval drawn from the remaining ones.	Each option names an adjacent-frame interval $[t, t+1]$; when the reference interval lies at either end of the ladder only one neighbour exists, and three options are presented.
T7_Speed_Variation_Classification	3	Fixed choices: speed basically unchanged, accelerate then decelerate, or decelerate then accelerate.	The stable class applies when maximum minus minimum segment speed is at most 0.10 m/s; otherwise the order of the speed extrema sets the answer.
T8_Peak_Yaw-Change_Characterization	6	Fixed product of left/right turn and three magnitude categories.	For the largest absolute adjacent-frame yaw change, the magnitude bins are $\lvert\Delta\theta\rvert \leq 30^\circ$, $30^\circ < \lvert\Delta\theta\rvert < 50^\circ$, and $\lvert\Delta\theta\rvert \geq 50^\circ$.
T9_Peak_Yaw-Change_Localization	4	The maximum-turn interval on the $t = 0, \ldots, T-2$ ladder, its two immediately neighbouring intervals, and one further interval drawn from the remaining ones.	Each option names an adjacent-frame interval $[t, t+1]$; items whose reference interval lies at a ladder end are completed with a further interval, so every item presents exactly four options.
T10_Yaw-Change_Sequence_Reconstruction	5	The reference turn sequence, one "no obvious turn" choice, and three distinct sequences edited from the reference.	Turns exceed 15° in magnitude; seeded edits flip turn directions, reverse turn order, or add/remove a turn event.
T13_Target_Loss_Duration	4	Four consecutive invisible-frame counts from 1 through the trajectory length T, in a window containing the reference count.	Each count n is paired with the five-percentage-point band containing $100n/T\%$; the count and band form one option.
T14_Lost-Target_Relative_Bearing_Estimation	4	The same four fixed bearing cells appear in every item: left and right at 50°–75° and at 75°–100° from the forward axis.	Each cell pairs a side with a 25° magnitude range. The option set is question-independent and its cells need not be consecutive on a single ladder.

Table 17: Supplementary Qwen3.8-Flash evaluation protocol.

Model	Qwen3.8-Flash
Access route	DashScope API (authenticated with a DashScope API key)
Evaluation scope	Supplementary text-only and context diagnostics in this appendix; excluded from the 28-model leaderboard
Inference	Zero-shot; temperature = 0; thinking disabled where specified for the diagnostic

Evaluation is zero-shot and uses no task-specific fine-tuning. Each model receives the task-specific left-camera RGB observations and neutral frame labels defined by Appendix D.3, and privileged state is never provided. Deployment settings are specified in the evaluation configuration. `qwen3.8max` also serves as the frozen semantic judge for AT-ID Qwen Team (2026c).

D.5 Decoding, Structured Output, and Answer Extraction

Inference uses each model's evaluated decoding configuration. Finite-label tasks use structured constraints where supported, whereas free-text identity and frame ordering use deterministic answer extraction.

The parser reads a JSON `answer` field when present. A choice answer is accepted only when it matches an option label or exactly matches an option text; time-index answers are normalized to canonical `t=N` form. For AppOrder, the parser also extracts Chinese numbered-frame labels, `Frame N`, or `#N` from plain text or a list. A frame sequence is valid only when its length equals the reference length, every label occurs once, and its label set exactly matches the displayed set. Missing, repeated, unknown, or extra labels are format errors; a well-formed sequence whose order differs from the reference is a content error rather than a format error. The two are recorded separately and are both scored as incorrect.

Target-identity responses are Unicode-normalized before being passed to the frozen semantic judge. The official score uses the judge's binary decision; case-insensitive literal equality does not determine the score.

D.6 Rendered Examples for the Four Output Modes

Choice output. `Task: target_distance_interval; Input: 24 ordered images labelled t=0,...,t=23; Question: At t=2, which interval contains the navigation target's distance?` The options are `A. 1.50–1.65 m; B. 1.35–1.50 m; C. 1.05–1.20 m; D. 1.20–1.35 m.` The required output is `{"answer":"A"}`. Depth, target boxes, calibration, and source paths are hidden.

Time-index output. `Task: target_first_loss_time; Input: ordered images labelled t=0,...,t=23; Question: At which t does the target first become invisible?` The required output is `{"answer":"t=N"}`; the numeric value is filled by the task instance and normalized to this canonical form by the parser.

Free-text output. `Task: navigation_target_identity; Question: What object is the robot navigating toward? Provide one concise object name.` The required output is a single JSON answer such as `{"answer":"beige paper cup"}`. The English rendering is shown only to document the output contract; the benchmark prompt may use Chinese text.

Frame-sequence output. `Task: approach_frame_order; Input: eight images displayed under neutral labels Frame 1,...,Frame 8; Question: Return every label from farthest to nearest.` Under the revised text-sequence adapter, the required output is plain text, for example `Frame 3 -> Frame 1 -> Frame 2 -> ...`; every displayed label must occur exactly once. JSON-answer and plain-text outputs are parsed by the same frame-label normalizer; Gemma is scored using its completed plain-text sequence evaluation.

E Data Schema and Release Reproducibility

E.1 Benchmark Record Schema

Each benchmark question is a JSON object. The core fields are summarized below. Fields under `ground_truth` may be nested differently by task type, but their purpose is consistent: they preserve the exact construction value and the metadata needed for offline auditing.

Table 18: Core fields in a benchmark question record and their visibility to the evaluated model.

Field	Role	Model-visible?
`question_id`	Globally unique item identifier.	No
`question_type`	Canonical task identifier used to select the template; its raw value is not included in the prompt.	No
`trajectory_name`	Source trajectory identity.	No
`images`	Ordered image records and task-specific display metadata; only image content and neutral labels are exposed.	Image content and labels only
`question`	Natural-language question.	Yes
`options`	Shuffled option labels and texts for choice tasks.	Yes when applicable
`answer`, `answer_text`, `answer_value`	Official answer fields and position-independent structured value.	No
`ground_truth`	Exact metric, temporal, visibility, and audit fields.	No
`dataset_tag` and eligibility metadata	Release and provenance information.	No

The model-visible projection is a strict subset of the benchmark record. In particular, source image paths and all fields exposing privileged sensor records are removed from the request payload.

E.2 Reproducibility Checklist

Upon acceptance, we will publicly release the questions and images, construction parameters, generation code, and evaluation scripts. The release will include:

- the versioned whitelist and benchmark summary;
- all generator source files and their command-line defaults;
- the prompt JSON and prompt version;
- global seeds and dataset tags;
- sample manifests used for figures or control experiments;
- model profiles and decoding parameters;
- the answer parser and scoring implementation;
- the frozen result manifest and evaluated question identifiers;
- the derived `per_question_scores.jsonl`;
- the AppOrder prompt or adapter version used for every rerun, especially the revised Gemma text-sequence run; and
- SHA-256 signatures for the frozen prompt, configuration, and sample manifest.

This metadata is important because a changed prompt, option order, or parser can alter scores without changing model weights.

F Semantic Judging and Release

F.1 Reference-Based Semantic Judge

The semantic metric is restricted to `navigation_target_identity`; the other thirteen task types retain their task-specific deterministic scoring. The judge is a textual-equivalence evaluator: it receives the question, reference object name, and candidate response, but never the image sequence, trajectory metadata, privileged state, or evaluated-model identity. It therefore cannot solve the visual question on behalf of the evaluated MLLM.

For each target-identity record, the protocol passes the Chinese question, the reference object name (category together with its annotated color attributes), and the model's raw response to `qwen3.8max`. The frozen judge configuration uses a version-controlled prompt, temperature zero, a 128-token limit, disabled thinking, and JSON-object response mode. The required schema is `{"correct": true/false, "reason": "..."}`. Each request allows up to three attempts.

The reference preserves the annotated category and color description. The judge is instructed to extract an answer field from JSON when present, remove common prefixes and terminal punctuation, and decompose the reference into a core object category and optional attributes. A candidate is correct only when its core category matches the reference. A missing color is accepted; an explicit color is accepted when it is exact or an ordinary-language near-color. A clearly different color family or object category is incorrect. Broad scene descriptions that fail to name the designated target are not rewarded.

Table 19: Semantic-judge protocol for free-form target identity. The judge sees text only; visual evidence remains exclusively with the evaluated MLLM.

Component	Frozen specification
Inputs	Question, ground-truth target name, and raw candidate response
Judge	`qwen3.8max`; no images
Decision	Same core object category and absent, exact, or everyday near-color attribute
Output	JSON boolean plus a short reason; temperature 0; thinking disabled
Versioning	Frozen judge prompt and hashed configuration signature
Headline policy	The frozen semantic-judge decision is the official Approach Target Identification score used in the main table, aggregates, rankings, and target-grounding analysis.

Synonyms, hypernyms, and shortened object names are evaluated by the LLM under this same-category rule. There is no separate synonym dictionary, hypernym acceptance list, or abbreviation whitelist: the judge determines whether the candidate still identifies the reference object. Missing color is explicitly permitted by the prompt; a conflicting color family is rejected. The same frozen prompt and judge configuration are used for every model. The prompt is submitted as a single user-role message; no additional system-role message is supplied. The three fields below are replaced with the question, reference name, and model response before submission.

Frozen semantic-judge prompt (English translation).

> You are a rigorous answer evaluator. Determine whether the model's object name refers to the same object as the reference answer and whether its attribute descriptions are correct.
>
> [Question] `{question}`
> [Reference answer] `{gt_answer}`
> [Model output] `{prediction}`
>
> Evaluation procedure.
> 1. Preprocessing.
>
> - If the output is JSON, extract the content of its answer field.

- Remove prefixes such as “The answer is,” “This is,” and “Answer:”.
- Remove terminal punctuation.

2. Parse the reference answer. Identify the core object category (for example, “paper cup”) and any additional attributes (for example, the color “beige”).

3. Apply the decision rules.

Correct: the object category must match the reference (differences in function words such as “of” do not matter), and the color must satisfy at least one of the following: (a) no color is mentioned, which is an allowed omission; (b) the color matches exactly; or (c) the color is from a nearby color family or is interchangeable in ordinary usage. For example, beige, light yellow, apricot, light brown, and ivory may be treated as nearby colors. The practical criterion is whether an ordinary speaker would use the two color terms interchangeably to describe the same object.

Incorrect: the object category does not match the reference, or the colors clearly belong to different families (for example, “beige” versus “black,” “red,” “blue,” or “green”).

Return only the following JSON format, with no additional text. For color judgments, the reason must state whether the colors are treated as nearby.

```
{"correct": true/false,
 "reason": "brief explanation"}
```

The following table is an English rendering of eight saved judge records. Reference and candidate strings are translated from the original Chinese answers; the decisions and rationales are preserved in meaning. These examples illustrate actual decisions rather than an additional hand-written scoring rule.

Table 20: English rendering of recorded semantic-judge examples. The judge receives text only; visual evidence remains exclusively with the evaluated MLLM.

Case	Reference	Candidate	Decision	Rationale
Synonym	Black garment	Black clothing	Correct	The object category is the same (“clothing” and “garment” are synonyms), and the color matches exactly (“black”).
Abbreviation	Silver laptop	Silver notebook	Correct	The object category is the same; “notebook” is a common abbreviation for “laptop” in this context, and the color matches exactly (“silver”).
Broader name	Mineral-water bottle	Water bottle	Correct	“Water bottle” refers to the same object category as “mineral-water bottle” and is a general hypernym; no conflicting color is introduced.
Color omitted	Beige paper cup	Paper cup	Correct	The core object category matches, and the candidate omits the color; missing color is allowed.
Approximate color	Beige cup	White mug	Correct	The object category matches (“mug” and “cup”), and white and beige are treated as nearby colors in ordinary usage.
Category insufficient	Yellow beverage bottle	Yellow small bottle	Incorrect	The category is not sufficiently specific: “small bottle” is not a synonym or reasonable abbreviation for “beverage bottle.”
Wrong category	Pink storage box	Wooden table	Incorrect	The categories conflict: the reference is a storage box, whereas the candidate is a table.
Wrong color	Black paper cup	White paper cup	Incorrect	The category matches, but black and white clearly belong to different color families.

F.2 Evaluation Coverage

Table 21: Evaluation coverage for the 14-task benchmark. NAttempted is the number of scheduled model–question evaluations; NScored and NExcluded partition it. No-response/network failures are retried until a valid answer is scored, so counts report post-retry outcomes. Failure rate and coverage use NAttempted as denominator. Closed-source models used 300 questions per task (4,200 total), whereas open-source models used the full 9,491-question benchmark.

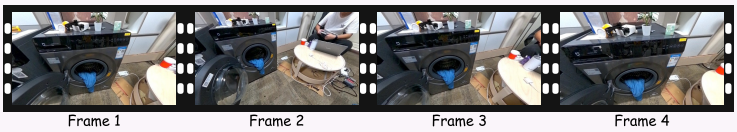
Model	N attempted	N scored	N excluded	No response/ network	Parse/ format	Failure rate (%)	Coverage (%)
Open-source models							
Qwen3.5-397B-A17B	9491	9491	0	0	0	0.00	100.00
GLM-4.6V-Flash	9491	9491	0	0	0	0.00	100.00
InternVL3.5-30B-A3B	9491	9491	0	0	0	0.00	100.00
InternVL3.5-8B	9491	9491	0	0	0	0.00	100.00
Cosmos-Reason2-8B	9491	9491	0	0	0	0.00	100.00
LLaVA-OneVision-Qwen2-7B	9491	9491	0	0	0	0.00	100.00
Qwen2.5-VL-7B	9491	9491	0	0	0	0.00	100.00
Nemotron-Nano-VL-8B	9491	9491	0	0	0	0.00	100.00
InternVL3-1B	9491	9491	0	0	0	0.00	100.00
Gemma-3n-E2B	9491	9491	0	0	0	0.00	100.00
RoboBrain2.5-8B	9491	9491	0	0	0	0.00	100.00
Qwen3-VL-8B-Instruct	9491	9491	0	0	0	0.00	100.00
Qwen3-VL-8B-Thinking	9491	9491	0	0	0	0.00	100.00
Qwen3-VL-30B-A3B-Instruct	9491	9491	0	0	0	0.00	100.00
Qwen3-VL-30B-A3B-Thinking	9491	9491	0	0	0	0.00	100.00
Qwen3-VL-32B-Instruct	9491	9491	0	0	0	0.00	100.00
Qwen3-VL-32B-Thinking	9491	9491	0	0	0	0.00	100.00
Qwen3-VL-235B-Instruct	9491	9491	0	0	0	0.00	100.00
Qwen3-VL-235B-Thinking	9491	9491	0	0	0	0.00	100.00
Closed-source models							
Gemini-3.8-Flash	4200	4200	0	0	0	0.00	100.00
GPT-5.6-Luna	4200	4200	0	0	0	0.00	100.00
GPT-5.6-Sol	4200	4200	0	0	0	0.00	100.00
Qwen3.8-Max	4200	4200	0	0	0	0.00	100.00
Qwen3.7-Plus	4200	4200	0	0	0	0.00	100.00
Qwen3.7-Flash	4200	4200	0	0	0	0.00	100.00
Qwen3.5-Omni-Plus	4200	4200	0	0	0	0.00	100.00
DeepSeek-V4.1-Flash	4200	4200	0	0	0	0.00	100.00
Kimi-K3	4200	4200	0	0	0	0.00	100.00

Note. Questions with no response/network errors are retried until a valid answer is scored; the table reports final post-retry outcomes.

G FOURTEEN TASK EXAMPLES

This section presents one example for each of the fourteen task types in RoboApproachBench. The English renderings preserve the question semantics, answer choices, units, and temporal indices where specified. Each panel shows all supplied RGB frames in task-input order, with at most four frames per row. The accompanying sample manifest records the available selection and image provenance.

Case 1: Target Approach Frame Ordering



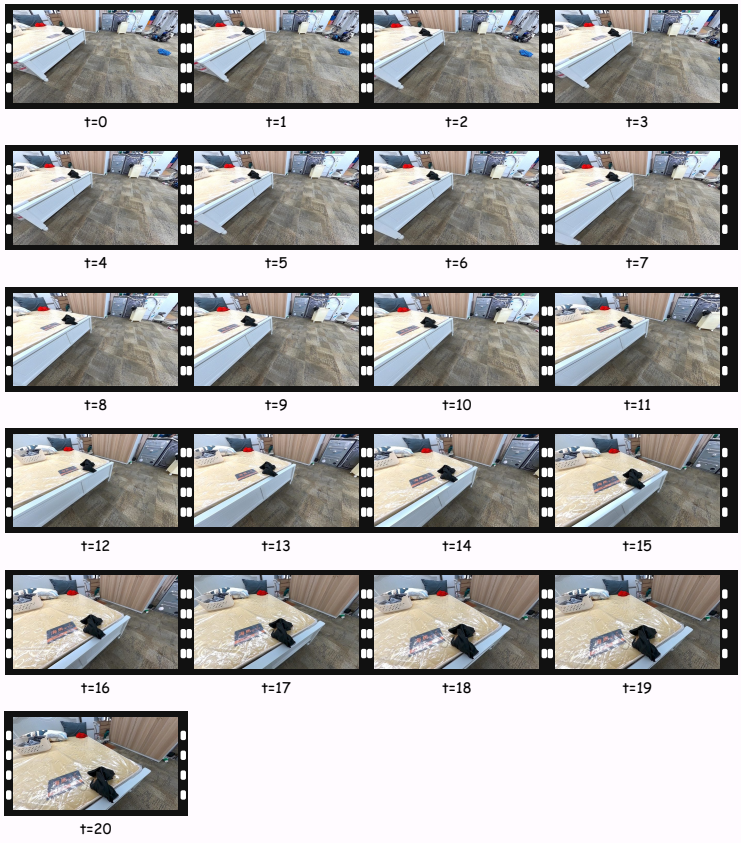


Question: These 7 images come from the same robot approach trajectory toward a designated object, but their order has been randomly shuffled. The images are numbered in the current display order as Frame 1, Frame 2, ... , Frame 7. Based on the target's apparent size, perspective, and changes in robot-target distance, reorder the displayed frame IDs from farthest to nearest relative to the target. Output only the complete sequence (e.g., Frame 3 -> Frame 1 -> Frame 2).

Answer: Frame 7 -> Frame 2 -> Frame 5 -> Frame 3 -> Frame 1 -> Frame 4 -> Frame 6

Figure 5: Complete task example for Approach Target Frame Ordering.

Case 2: Mean Translational Speed Estimation



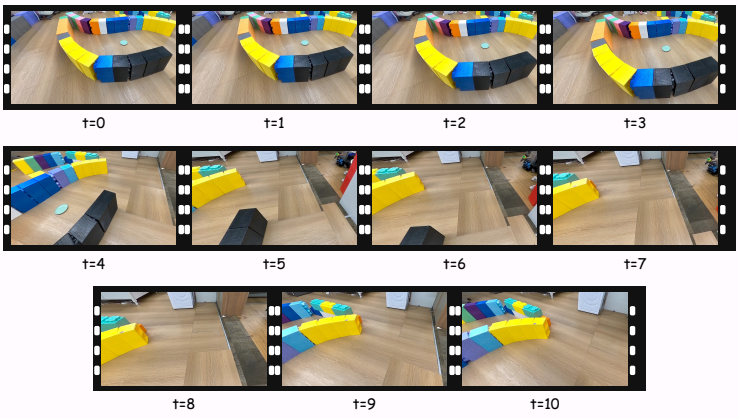
Question: Given an approximately 1.1 s interval between adjacent frames, observe the complete motion sequence in temporal order. In which range does the trajectory's mean translational speed lie?

Options:
(A) 0.10-0.15 m/s
(B) 0.00-0.05 m/s
(C) 0.05-0.10 m/s
(D) 0.20-0.25 m/s
(E) 0.15-0.20 m/s

Answer: (C) 0.05-0.10 m/s

Figure 6: Complete task example for Mean Translational Speed Estimation.

Case 3: Lost-Target Relative Bearing Estimation



Question: Observe the images from t=0 to t=10 in their original trajectory order. The three questions for this occlusion event use exactly the same images and options. The navigation target, the 'cyan plate,' is invisible at query time t=10. Determine its direction and angular range relative to the robot's forward direction at t=10; do not assume the query refers to the last image.

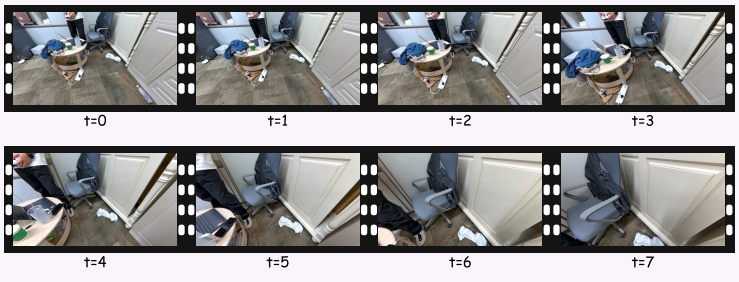
Options:

(A) The target is to the right, approximately 75-100 degrees from the forward direction.
(B) The target is to the left, approximately 75-100 degrees from the forward direction.
(C) The target is to the left, approximately 50-75 degrees from the forward direction.
(D) The target is to the right, approximately 50-75 degrees from the forward direction.

Answer: (B) The target is to the left, approximately 75-100 degrees from the forward direction.

Figure 7: Complete task example for Lost-Target Relative Bearing Estimation.

Case 4: Approach Target Identification



Question: Observe the complete image sequence in temporal order from t=0. What object is the robot navigating toward? Enter only the object name.

Answer: White towel

Figure 8: Complete task example for Approach Target Identification.

Case 5: Speed Variation Classification



Question: Given an approximately 1.1 s interval between adjacent frames, treated as constant, observe the images in temporal order from T0 and determine the overall speed change before reaching the destination. If the difference between the maximum and minimum speeds is no more than 0.1 m/s, classify the speed as essentially constant; otherwise, choose the description matching the order of the maximum and minimum speeds.

Options:
(A) Accelerate, then decelerate, and finally reach the destination.
(B) Decelerate, then accelerate, and finally reach the destination.
(C) The maximum-minimum speed difference is at most 0.1 m/s; the speed is essentially constant.

Answer: (C) The maximum-minimum speed difference is at most 0.1 m/s; the speed is essentially constant.

Figure 9: Complete task example for Speed Variation Classification.

Case 6: Approach Progress Reasoning



Question: Observe the four frames in temporal order. Comparing only Frames 1 and 4, which option best describes the change in the straight-line distance from the camera to the navigation target, "white glove"?

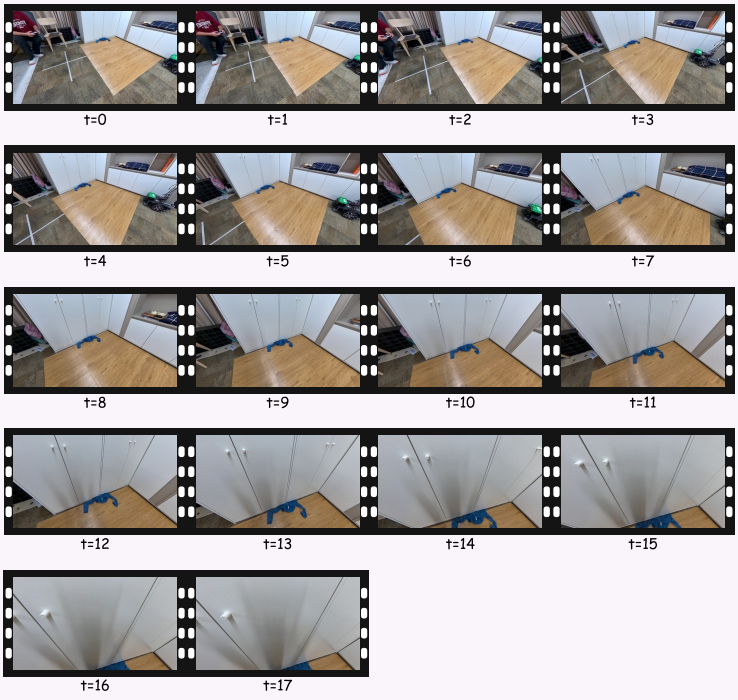
Options:

(A) The target distance decreases by at least 0.20 m but less than 0.30 m.
(B) The target distance decreases by at least 0.30 m.
(C) The target distance decreases by at least 0.10 m but less than 0.20 m.
(D) The target distance decreases by less than 0.10 m.

Answer: (A) Decrease by at least 0.20 m but less than 0.30 m.

Figure 10: Complete task example for Approach Progress Reasoning.

Case 7: Target Depth Estimation



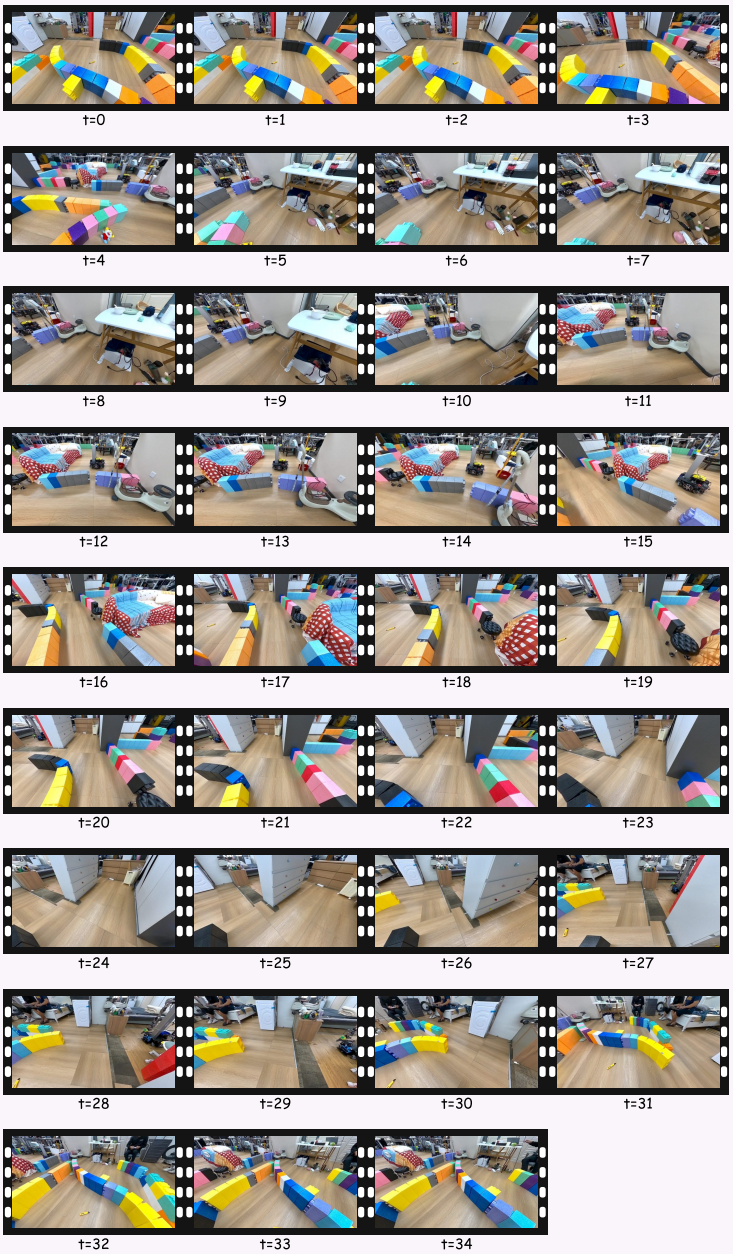
Question: Observe the complete image sequence in temporal order from t=0. At t=9, approximately which range describes the distance from the blue garment to the camera?

Options:
(A) 1.50-1.65 m
(B) 1.65-1.80 m
(C) 1.20-1.35 m
(D) 1.35-1.50 m

Answer: (D) 1.35-1.50 m

Figure 11: Complete task example for Target Depth Estimation.

Case 8: Target Loss Onset

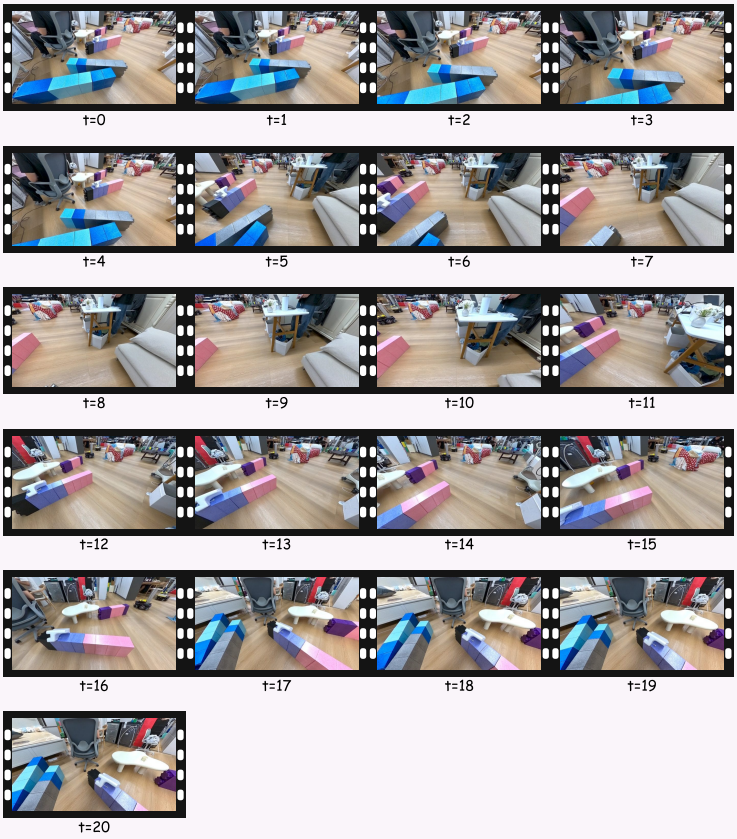


Question: Observe the complete image sequence in temporal order from t=0. During the yellow utility knife's third temporary disappearance from view, at which frame does it first become invisible? Answer only t and the number, e.g., t=3.

Answer: t=29

Figure 12: Complete task example for Target Loss Onset.

Case 9: Target Loss Duration



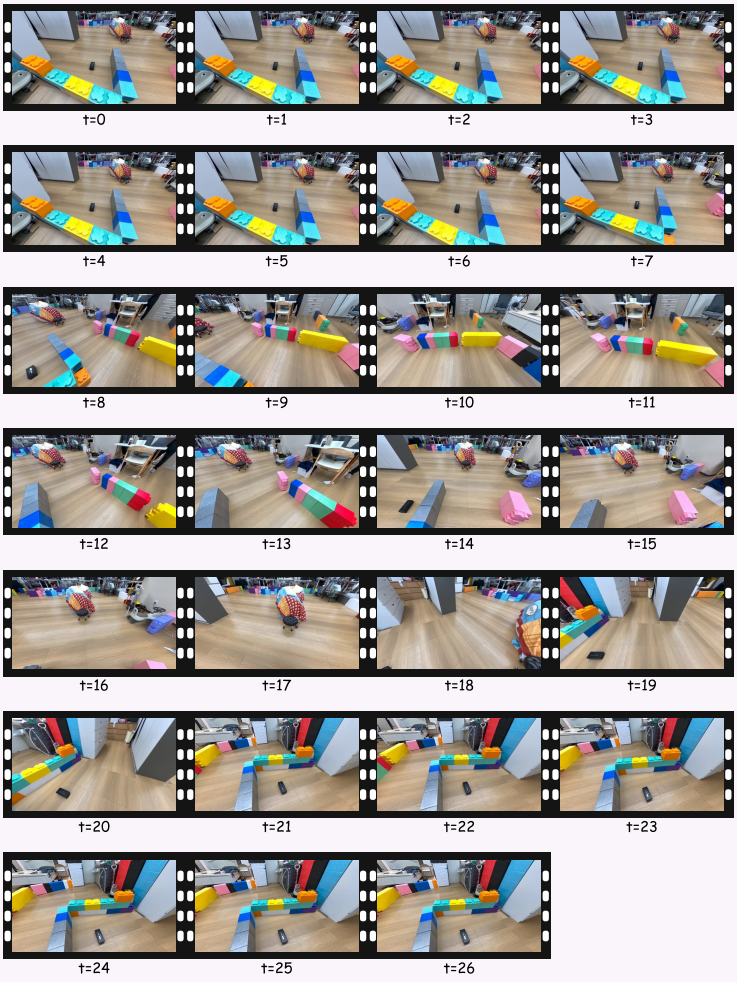
Question: Observe the complete image sequence in temporal order from t=0. Between the white foam foot mat's second disappearance and reappearance, how many consecutive frames is it invisible, and approximately what fraction of the full trajectory does this represent?

Options:
(A) 3 consecutive frames, approximately 10-15 percent of the trajectory.
(B) 2 consecutive frames, approximately 5-10 percent of the trajectory.
(C) 4 consecutive frames, approximately 15-20 percent of the trajectory.
(D) 1 consecutive frame, approximately 0-5 percent of the trajectory.

Answer: (D) 1 consecutive frame, approximately 0-5 percent of the trajectory.

Figure 13: Complete task example for Target Loss Duration.

Case 10: Target Loss Offset

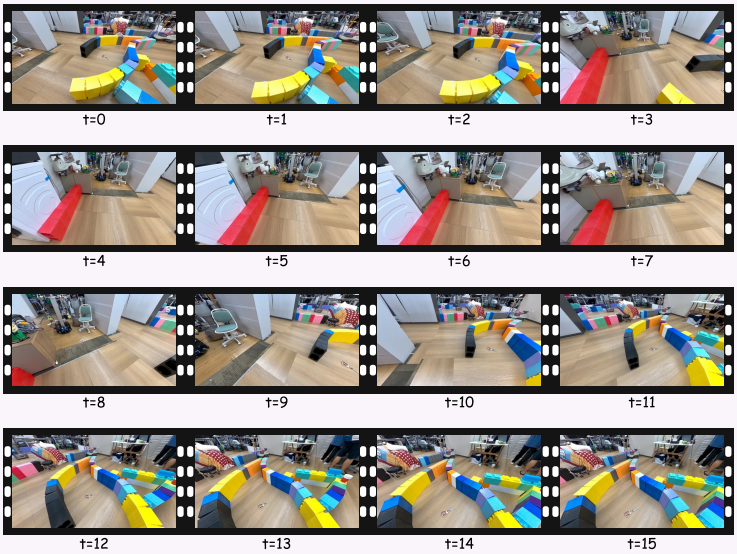


Question: Observe the complete image sequence in temporal order from t=0. After the black box's second temporary disappearance, at which frame does it re-enter view? Answer only t and the number, e.g., t=7.

Answer: t=19

Figure 14: Complete task example for Target Loss Offset.

Case 11: Peak Yaw-Change Characterization



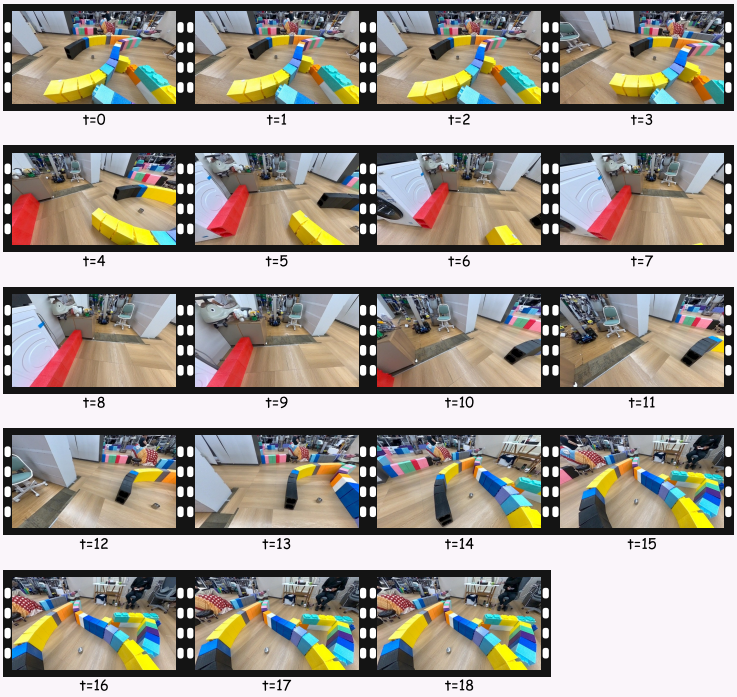
Question: Observe the images in temporal order. What are the direction and magnitude range of the largest turn during the motion? The magnitude bins are: at most 30 degrees, greater than 30 degrees and less than 50 degrees, or at least 50 degrees.

Options:
(A) Turn right, with magnitude at least 50 degrees.
(B) Turn left, with magnitude at least 50 degrees.
(C) Turn right, with magnitude greater than 30 degrees and less than 50 degrees.
(D) Turn right, with magnitude at most 30 degrees.
(E) Turn left, with magnitude greater than 30 degrees and less than 50 degrees.
(F) Turn left, with magnitude at most 30 degrees.

Answer: (E) Turn left, with magnitude greater than 30 degrees and less than 50 degrees.

Figure 15: Complete task example for Peak Yaw-Change Characterization.

Case 12: Peak Yaw-Change Localization



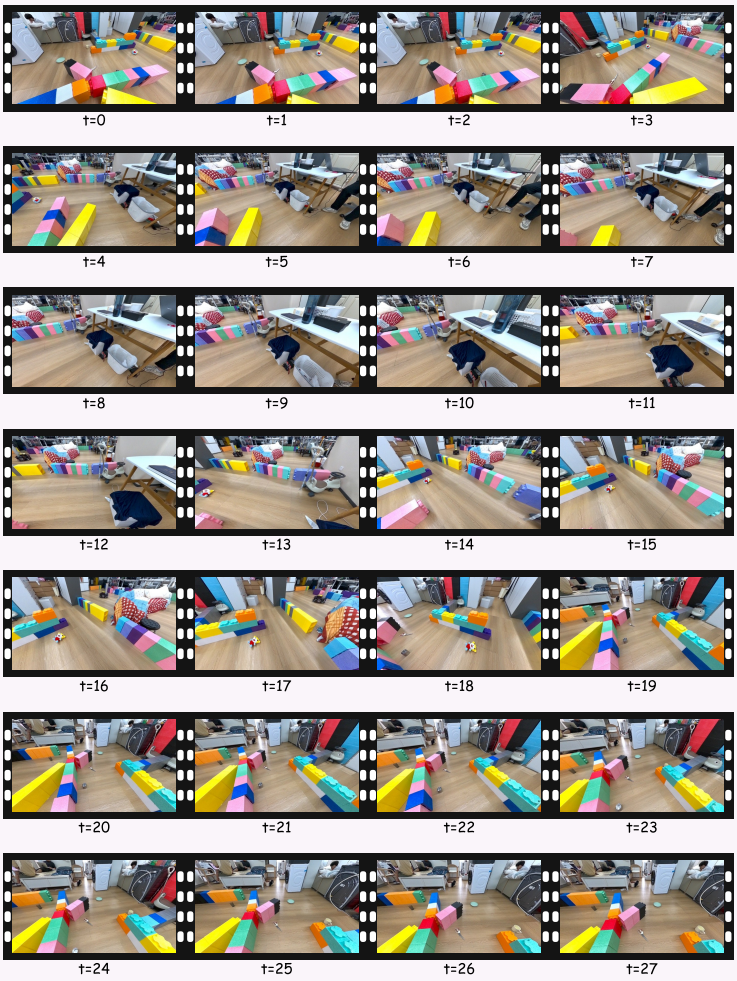
Question: Observe the images in temporal order. Between which two adjacent time points does the largest turn occur?

Options:
(A) t=13 and t=14
(B) t=14 and t=15
(C) t=0 and t=1
(D) t=12 and t=13

Answer: (A) t=13 and t=14

Figure 16: Complete task example for Peak Yaw-Change Localization.

Case 13: Yaw-Change Sequence Reconstruction



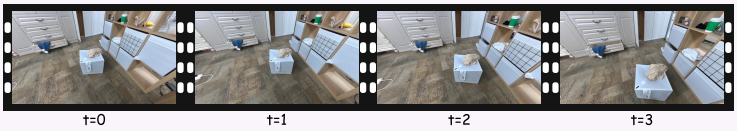
Question: Recall the complete motion sequence in temporal order. Did any large turns occur? In what order did the significant turns occur?

Options:
(A) No obvious turns occur.
(B) Straight -> right -> right -> left -> left -> straight
(C) Straight -> right -> right -> left -> right -> right -> straight
(D) Straight -> left -> left -> left -> right -> straight
(E) Straight -> right -> left -> left -> left -> straight

Answer: (E) Straight -> right -> left -> left -> left -> straight

Figure 17: Complete task example for Yaw-Change Sequence Reconstruction.

Case 14: Peak Translational Speed Interval Localization





Question: Given an approximately 1.1 s interval between adjacent frames, observe the images in temporal order. Between which adjacent time points does the robot's translational speed reach its maximum?

Options:
(A) t=1 and t=2
(B) t=2 and t=3
(C) t=0 and t=1
(D) t=3 and t=4

Answer: (B) t=2 and t=3

Figure 18: Complete task example for Peak Translational Speed Interval Localization.