IndustryNAV: Evaluating Spatial Reasoning and Safety of Embodied Agents in Dynamic Industrial Navigation

Anonymous authors
Paper under double-blind review

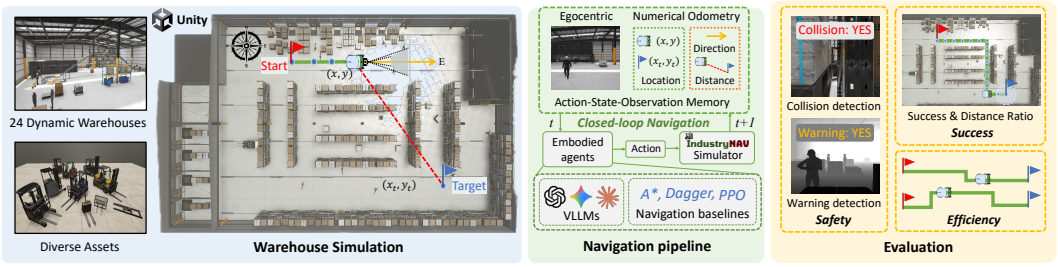

Figure 1: **Overview of the IndustryNav simulation environment.** IndustryNav, built on Unity, features 24 dynamic warehouses. The closed-loop VLLM interface combines egocentric RGB, numerical odometry, and action–state–observation memory. The simulator also supports A*, DAgger, and PPO navigation baselines. Evaluation covers task success, efficiency, and safety.

Abstract

Visual large language models (VLLMs) are increasingly used as embodied agents, yet industrial warehouses demand more than reaching a goal: agents must navigate safely around moving workers and vehicles. To evaluate embodied agents' goal reaching and safety, we introduce IndustryNav, a simulation environment with 24 warehouse scenes for closed-loop navigation to specified destinations. Alongside task success and efficiency, we introduce safety-aware metrics that capture collisions and near-collision warnings along each trajectory. Experiments with eleven leading open- and closed-source VLLMs and three navigation baselines show that agents can reach their goals yet collide with moving obstacles, exposing a gap between success and safe navigation. This gap matters in safety-critical settings, where a single collision can end the task. IndustryNav also supports policy learning through automated oracle trajectories and simulator interaction, with held-out scenes for generalization tests. These experiments show that policies can succeed on new routes in familiar warehouses yet fail to transfer to held-out layouts, making cross-scene generalization an open challenge.

1 Introduction

In a warehouse, people judge gaps between goods and navigate around moving workers and vehicles. Unlike in household scenes, these movements can quickly block a clear route. A collision may injure a worker or damage equipment even if the agent reaches its goal, making spatial reasoning essential for safe movement. Visual large language models (VLLMs), especially GPT-6-Astra [12, 38], increasingly serve as embodied agents [54] across diverse tasks [29, 30, 51], including perception [35, 71], manipulation [26], and navigation [59, 60]. Yet prior work finds that VLLMs struggle to estimate distances between objects, judge relative depth, and remember layouts across views [9, 17, 58]. This raises a question: *can VLLMs reach a goal while maintaining safe clearance in a changing environment like a warehouse?*

Evaluating safe navigation in dynamic industrial warehouses requires observing how agents respond as people and equipment move, not just whether they reach a goal. Existing benchmarks address parts of this problem (see Tab. 1): image-based [9, 14, 33, 48, 50, 65] and video-based evaluations [28, 58] help characterize spatial understanding, while active benchmarks connect reason-

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.

ing to action through embodied question answering [35], spatial-belief construction through exploration [64], household task execution [11, 27], and collaborative planning [6]. This motivates an interactive setting that measures both progress and safety throughout navigation. Such a setting can also provide interaction data for policy training, raising a complementary question: *can policies that reach goals on new routes in familiar warehouses still do so safely in entirely unseen layouts?* Answering this requires holding out entire scenes, rather than only evaluation start–goal pairs.

To address both questions, we introduce IndustryNav (see Fig. 1), a simulation environment with 24 industrial warehouse scenes built in Unity by five experts. Varied layouts, moving equipment, and people provide the setting for evaluating safety and generalization. For VLLM evaluation, agents navigate to a specified goal point (PointGoal navigation), using an updated egocentric RGB view, numerical odometry, and a short history of recent steps to choose each action. We measure task success and efficiency together with collision and warning ratios, finding that *current VLLMs struggle to reach goals safely, and success rate alone can hide collisions along the way*. To support policy learning, IndustryNav automatically collects expert trajectories with a full-map planner for behavior cloning and provides simulator interaction for reinforcement learning. For both learned policies, we compare training with and without the evaluation scenes on the same tasks. We find that *learned policies still struggle to generalize to unseen scenes*. Our contributions are fourfold:

Table 1: IndustryNav is the first simulation environment designed specifically for warehouse navigation. Built in Unity, it supports fast, cross-platform simulation and emphasizes safety-aware evaluation in an active, dynamic environment. Safety: safety-aware evaluation; Dyn.: moving obstacles; Active: environment interaction.

Benchmark	Scenario	Platform	Safety	Dyn.	Active
SpatialRGBT-Bench [9]	Household	Omni3D	✗	✗	✗
EmbSpatial-Bench [16]	Household	AI2-THOR, Habitat	✗	✗	✗
RoboSpatial [48]	Household	Mixed Indoor sim	✗	✗	✗
VSI-Bench [58]	Household	ScanNet/++, ARKitScenes	✗	✗	✗
Q-Spatial [32]	Household	ScanNet	✗	✗	✗
3DSRBench [33]	In-the-wild	COCO	✗	✗	✗
Open3D-VQA [65]	Outdoor	UrbanScene3D, EmbodiedCity	✗	✗	✗
CA-VQA [14]	Household	CA-1M, ARKitScenes	✗	✗	✗
OpenEQA [35]	Household	ScanNet, HM3D	✗	✗	✓
Theory of Space [64]	Multi-room	ThreeDWorld	✗	✗	✓
VSP [53]	Maze Game	OpenAI-Gym	✗	✗	✓
BEHAVIOR-1K [27]	Household	OmniGibson	✗	✗	✓
LoTa-Bench [11]	Household	AI2-THOR, VirtualHome	✗	✗	✓
PARTNR [6]	Household	Habitat 3.0	✗	✓	✓
IndustryNav	Warehouse	Unity	✓	✓	✓

- We introduce **IndustryNav**, an interactive simulation environment comprising 24 industrial scenes for studying navigation in dynamic settings. It supports evaluation of embodied agents' spatial reasoning, policy training, and cross-scene generalization studies.
- A closed-loop navigation pipeline enables zero-shot evaluation of VLLM agents. It combines egocentric RGB, numerical odometry information, and action-state-observation memory to prompt agents to reason and select navigation actions at each step.
- For policy learning, IndustryNav supports training on a subset of scenes and evaluation on held-out scenes, with a full-map planner providing automatically annotated expert trajectories. Navigation baselines cover behavior cloning and reinforcement learning.
- To assess navigation safety, we design Collision Ratio (CR) and Warning Ratio (WR) to quantify collisions and near-collision risks throughout navigation. These safety-aware metrics complement task success and efficiency, capturing unsafe behavior that success rate alone can overlook.

2 INDUSTRYNAV

2.1 INDUSTRIAL SCENARIO CREATION

IndustryNav comprises 24 industrial warehouse scenes created in Unity by five experts. We built IndustryNav in Unity so that it can run across platforms and benefit from Unity's mature developer ecosystem and fast simulation. The scenes vary in aisle layout, storage density, open floor space, and obstacle placement. Each scene provides four PointGoal tasks defined by four pairs of start and goal points, yielding varied route lengths and navigation challenges.

Dynamic Warehouse Traffic. Beyond this layout diversity, IndustryNav models warehouse traffic with moving forklifts, ground robots, and workers that can cross or temporarily block navigation paths. Physics-based colliders stop the agent when it contacts an obstacle. Each task can be run in either static or dynamic mode for controlled comparison. Further details on scene assets, rendering, motion authoring, and simulator implementation are provided in Section A.2.

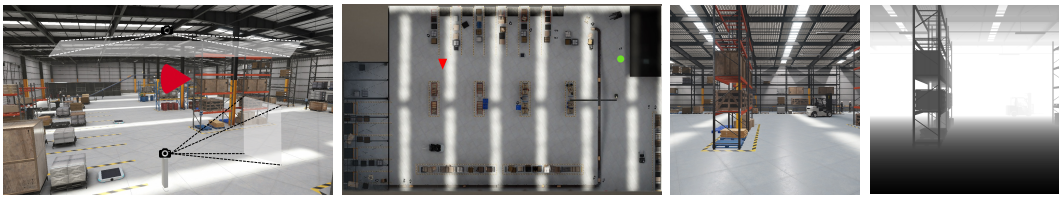

Illustration of agent cameras | Top-down view | Egocentric view (RGB) | Egocentric view (Depth)

Figure 2: Camera setup in IndustryNav. The latest egocentric RGB frame is the default visual input to the VLLM, whereas the learned policies receive depth together with numerical odometry. Depth also supports safety evaluation, while the top-down view supports trajectory visualization.

Observation Interface. IndustryNav synchronizes egocentric RGB-D, numerical odometry (agent and target positions and heading), and a top-down view (Fig. 2). Since a current global layout may be unavailable in practice, the default VLLM receives only RGB and numerical odometry; its memory comes from earlier steps. DAgger and PPO instead receive depth and odometry. Depth also supports safety evaluation; the top-down view is reserved for visualization and a privileged minimap ablation. ML-Agents supports headless training and evaluation.

2.2 Navigation Pipeline

To evaluate the spatial reasoning capabilities of VLLM-based embodied agents without task-specific training, we formulate navigation in IndustryNav as a zero-shot PointGoal task [7, 39, 44, 45, 49, 55]. In contrast to navigation-specific models trained through distillation (*e.g.*, SpatialVLM [8] and SpatialLLM [34]) or systems equipped with a separate planner (*e.g.*, NavGPT [70]), our default pipeline places a single general-purpose VLLM directly in a closed perception–reasoning–action loop. No learned controller, map builder, or external path planner assists the VLLM.

Per-step observation. At step t, the model receives the latest egocentric RGB frame and numerical odometry: current and target positions (x_t, z_t) and (x_g, z_g), goal distance d_t, and yaw θ_t. Positions use Unity world coordinates, and both positions and d_t are measured in meters. RGB provides local obstacle and motion cues, while odometry provides goal direction and progress. The prompt uses $+X$ as North and $+Z$ as West, with yaw $0°/90°/180°/270°$ corresponding to West/North/East/-South. The default VLLM receives neither depth nor a top-down map.

Action–state–observation memory. Each request includes the five most recent action–state–observation records. These help the model avoid repeated actions and change direction when blocked by a collision or local trap. Memory use and length are examined in Tab. 4 and Fig. 20.

Action execution and termination. The model returns one action with brief observation and reasoning. ML-Agents executes it for two simulation steps (approximately 1.50 m forward or a $45°$ turn) before refreshing RGB and odometry. Success requires a final world distance of at most 2 m. For VLLM runs, the maximum number of model decisions $T_{\max}$ is initialized from the start–target world distance D_0:

$$T_{\max} = \max(50, \min(160, \lceil 20 + 1.25 D_0 \rceil)) \,. \tag{1}$$

Here, $D_0 = \sqrt{(x_0 - x_g)^2 + (z_0 - z_g)^2}$ is measured in meters, and 1.25 is the configured number of decision steps per meter. The 20-step overhead allows for turning and recovery, while $T_{\max}$ is bounded between 50 and 160 model decisions. The episode ends upon success, a valid `stop`, or budget exhaustion.

The VLLM interface is summarized below; full prompts are provided in Appendix A.6.

VLLM Navigation Interface

Input Latest RGB; (x_t, z_t); (x_g, z_g); d_t; θ_t; five-step action–state–observation memory.
Output JSON with `observation`, `reasoning`, and `action` fields.
Actions `forward` (≈ 1.50 m), `turn left/right` ($\approx 45°$), or `stop` ($d_t \leq 2$ m).

2.3 EVALUATION

We report means over N valid runs using task-success (SR and DR), efficiency (ES), and safety metrics (CR and WR).

Success Ratio (SR). SR measures goal attainment from the final world distance d_i:

$$\mathrm{SR} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}[d_i \le \delta]\,, \tag{2}$$

where $\delta = 2$ m by default; we also report SR@5 m and SR@10 m.

Distance Ratio (DR). DR captures progress when a run does not succeed. For the first and last valid logged distances D_i and $\hat{d}_i$,

$$r_i = \begin{cases} 1, & \hat{d}_i \le \delta, \\ \dfrac{|D_i - \hat{d}_i|}{D_i}, & \hat{d}_i > \delta, \end{cases} \qquad \mathrm{DR} = \frac{1}{N}\sum_{i=1}^{N} r_i. \tag{3}$$

All tasks use $D_i > 0$, and successful runs are clamped to one.

Efficiency Score (ES). ES measures extra actions beyond the obstacle-free optimum T_i^*, computed from the start–goal distance and heading under 1.5 m forward moves and 45° turns:

$$\eta_i = \min\left(\frac{T_i^* + K - T_i}{K}, 1\right), \qquad \mathrm{ES} = \frac{1}{N}\sum_{i=1}^{N} \eta_i. \tag{4}$$

Here, T_i is the actual step count and $K = 50$ is the default excess-step budget; higher ES is better.

Collision Ratio (CR). We use reduced forward progress as a collision proxy: a forward step is flagged if the agent's position changes by less than 95% of the commanded distance. We measure this change between consecutive world positions and obtain the commanded distance from the logged `move` value (0.1 m per unit). If C_i of the F_i eligible forward steps in run i are flagged,

$$\mathrm{CR} = \frac{1}{N}\sum_{i=1}^{N} \frac{C_i}{F_i}, \tag{5}$$



Figure 3: **Warning detection mechanism.** A warning requires at least 0.5% of valid forward-RoI pixels to lie within the action-dependent safety distance.

with $C_i/F_i = 0$ when $F_i = 0$. CR can expose blocked moves even when the agent eventually reaches the goal.

Warning Ratio (WR). WR measures near-collision risk from action-aligned depth (Fig. 3). We flag a warning when at least 0.5% of valid forward-RoI pixels lie within the commanded forward distance plus a 0.4 m margin (or 0.4 m for actions without forward motion). If W_i of the L_i readable frames in run i are flagged,

$$\mathrm{WR} = \frac{1}{N}\sum_{i=1}^{N} \frac{W_i}{L_i}, \tag{6}$$

with the per-run ratio set to zero when $L_i = 0$. Lower CR and WR indicate safer navigation.

3 EXPERIMENT

3.1 EMBODIED AGENT COMPARISON

Settings. Across 24 dynamic warehouse scenes, we evaluate eleven VLLMs and three navigation baselines. The closed-source models are Claude-Sonnet-5, GPT-5-Mini, GPT-5.6-Luna, GPT-5.6-Terra, GPT-5.6-Sol, GPT-6-Astra, and Gemini-3.8-Flash [2, 18, 37, 38]; the open-source models are Qwen3.8-Flash, MiniMax-M3, DeepSeek-V4-Flash-Vision-Exp, and GLM-5.3-Flash [15, 36,

Table 2: Navigation performance in the 24-scene IndustryNav simulation environment. All experiments are conducted in **dynamic industrial environments** with moving obstacles. Training types include supervised fine-tuning (SFT), reinforcement learning (RL), and zero-shot evaluation (ZS). All scores are reported as percentages. SR denotes Success Ratio (at 2 m, 5 m, and 10 m thresholds); DR, Distance Ratio; ES, Efficiency Score; CR, Collision Ratio; and WR, Warning Ratio. Higher is better for SR, DR, and ES; lower is better for CR and WR. Within each category, **bold** and underlined values indicate the best and second-best performance, respectively.

Embodied Agent	Training Type	Task Success (%)				Efficiency (%)	Safety (%)	
		SR@2m ↑	SR@5m ↑	SR@10m ↑	DR ↑	ES ↑	CR ↓	WR ↓
Navigation Baselines								
DAgger [43]	SFT	52.08	53.12	65.62	71.83	29.91	12.27	18.01
PPO [46]	RL	73.56	79.17	87.50	42.12	84.46	11.87	13.80
A* [20]	ZS	**100.00**	**100.00**	**100.00**	**100.00**	**92.19**	**5.55**	**10.04**
Closed-Source VLLMs								
Claude-Sonnet-5 [2]	ZS	19.79	29.17	40.62	63.85	17.96	46.55	52.41
GPT-5-Mini [37]	ZS	11.83	15.05	18.28	45.76	11.68	21.90	36.96
GPT-5.6-Luna [38]	ZS	17.71	26.04	41.67	61.88	14.77	9.28	13.35
GPT-5.6-Terra [38]	ZS	18.48	25.00	39.13	59.05	14.79	15.65	24.21
GPT-5.6-Sol [38]	ZS	20.83	30.21	43.75	63.15	16.81	**3.79**	**12.15**
GPT-6-Astra [38]	ZS	51.04	57.29	66.67	78.89	44.71	4.76	14.92
Gemini-3.8-Flash [18]	ZS	**59.38**	**62.50**	**73.96**	**83.34**	**54.37**	4.85	13.62
Open-Source VLLMs								
Qwen3.8-Flash [41]	ZS	3.19	3.19	8.51	31.72	2.53	13.34	**13.30**
MiniMax-M3 [36]	ZS	17.71	28.12	43.75	66.19	14.48	27.85	37.77
DeepSeek-V4-Flash-Vision-Exp [15]	ZS	21.88	31.25	48.96	67.89	20.15	20.65	33.13
GLM-5.3-Flash [62]	ZS	**41.67**	**48.96**	**61.46**	**77.44**	**37.12**	**10.54**	18.96

41, 62]. Navigation references include A* [20], DAgger [43], and PPO [46]. Their training types are zero-shot (ZS), supervised fine-tuning (SFT), and reinforcement learning (RL), respectively; all VLLMs are evaluated zero-shot. Further baseline details are provided in Appendix B.

Each scene has four start–target pairs, yielding up to 96 dynamic runs per agent in Tab. 2. At each decision, the VLLM receives the latest egocentric RGB, numerical odometry, and five recent action–state–observation records, then returns an action and brief reasoning. The goal-distance-dependent decision budget is clipped to 50–160 steps.

We use the metrics in Section 2.3, with a default success radius of $\delta = 2$ m and excess-step budget of $K = 50$; WR uses the evaluator's action-dependent safety distance. In Tab. 2, the best and second-best results within each category are bold and underlined, respectively.

Results Analysis. Tab. 2 shows that success, progress, and safety rank models differently. An agent may reach the goal despite collisions, so success alone does not establish safe navigation. We examine this distinction for general-purpose agents and task-specific policies.

Approaching the goal remains easier than completing navigation. Gemini-3.8-Flash leads the VLLMs at 59.38% SR@2m, yet its higher SR@10m and DR show that many runs get close without finishing. Since agents receive the target coordinates, this gap suggests that the final approach, not goal localization alone, is a bottleneck: progress must be converted into precise closed-loop actions. A* reaches every goal using privileged geometry and a dedicated controller, showing headroom but not isolating VLLM reasoning.

The leading closed-source models set the pace, but model variation matters. Gemini-3.8-Flash and GPT-6-Astra lead all VLLMs in SR, DR, and reported ES. GLM-5.3-Flash is the strongest open-source model, reaching 41.67% SR@2m and 77.44% DR, and exceeding five of the seven closed-source models in success. The results therefore support an advantage for the leading closed-source models rather than a uniform separation between the two groups. GLM-5.3-Flash also substantially outperforms the other open-source models, showing that navigation performance varies widely even within the same model category. Models should therefore be evaluated individually rather than judged solely by whether they are open- or closed-source.

Success can conceal collisions along the way. Tab. 2 shows that success and safety do not coincide. A* reaches 100% SR@2m but still records 5.55% CR and 10.04% WR. Among VLLMs, Gemini-3.8-Flash has the highest SR@2m at 59.38%, whereas GPT-5.6-Sol has the lowest CR and WR but only 20.83% SR@2m. Success therefore cannot characterize navigation quality on its own; it must be interpreted alongside safety metrics. In real industrial environments, a single collision

Table 3: Cross-scene policy generalization on the same 24 canonical evaluation tasks from scenes 19–24. For each policy, training on scenes 1–18 (scene-held-out) is compared directly with training on independent point pairs from scenes 1–24 (scene-seen but evaluation-pair-held-out). SR denotes Success Ratio; DR, Distance Ratio; ES, success-weighted A*-normalized Efficiency Score; CR, Collision Ratio; and WR, Warning Ratio. Higher is better for SR, DR, and ES; lower is better for CR and WR. Green Δ rows report the signed change from training on scenes 1–18 to training on scenes 1–24; every highlighted change is an improvement.

Training Scenes	**Task Success (%)** SR@2m ↑	SR@5m ↑	SR@10m ↑	DR ↑	**Efficiency (%)** ES ↑	**Safety (%)** CR ↓	WR ↓
DAgger (SFT)							
1–18 (held-out)	25.00	33.33	37.50	55.99	10.04	29.63	8.74
1–24 (scene-seen)	45.83	45.83	58.33	67.28	27.00	16.12	7.37
Δ (1–24 − 1–18)	**+20.83**	**+12.50**	**+20.83**	**+11.29**	**+16.96**	**−13.51**	**−1.37**
PPO (RL)							
1–18 (held-out)	41.67	45.83	58.33	66.68	23.25	25.45	14.09
1–24 (scene-seen)	66.67	66.67	75.00	80.89	38.92	18.98	12.80
Δ (1–24 − 1–18)	**+25.00**	**+20.84**	**+16.67**	**+14.21**	**+15.67**	**−6.47**	**−1.29**

can damage equipment or terminate operation. By evaluating CR and WR alongside success, IndustryNav reveals collision and near-collision risks that goal-reaching metrics alone would overlook, enabling a more safety-aware assessment of navigation.

Task-trained policies outperform most VLLMs. PPO reaches 73.56% SR@2m versus 59.38% for Gemini-3.8-Flash, the strongest VLLM; DAgger at 52.08% also outperforms ten of the eleven VLLMs. On IndustryNav, the leading VLLMs we evaluate still fall short of the strongest task-trained policy, despite their broad pretraining.

3.2 Policy Learning and Cross-Scene Generalization

IndustryNav supports a learning loop beyond pretrained-agent evaluation: its privileged A* planner supplies expert trajectories for imitation learning, while simulator interaction supports reinforcement learning. This enables us to ask: *do task-trained policies transfer to unfamiliar warehouse layouts, or does their performance depend on training in the evaluation scenes?* We compare DAgger [43] and PPO [46] under two training regimes. D18/P18 train on scenes 1–18, leaving scenes 19–24 entirely unseen; D24/P24 train on all 24 scenes. In both regimes, training start–goal pairs are sampled separately from the 24 fixed evaluation pairs, and none of the evaluation pairs is used for training. We test all four policies on these same pairs in scenes 19–24, using the same dynamic setting and step limits. The within-method comparison therefore distinguishes *unseen-layout transfer* from *new-route generalization in familiar scenes*, rather than testing reproduction of training routes.

New-route success does not imply unseen-layout transfer. In Tab. 3, excluding the evaluation scenes from training lowers SR@2m from 45.83% to 25% for DAgger and from 66.67% to 41.67% for PPO. The same ordering holds at 5 m and 10 m, with lower DR and higher CR and WR in both held-out settings. The gap is thus not merely a consequence of a strict stopping radius: it also appears in goal progress and obstacle interaction. ES likewise decreases, although this table uses a success-weighted score, so the change does not independently establish less efficient trajectories among successful runs. Together, these results show that performance on new routes within familiar warehouses can overstate a policy's ability to navigate unfamiliar layouts.

Goal reaching and safety remain distinct objectives. PPO reaches more goals than DAgger in both training regimes. With all-scene training, DAgger has lower CR and WR, revealing a trade-off between task completion and obstacle exposure.

A testbed for learning transferable policies. Even P24, the strongest setting here, achieves only 66.67% SR@2m despite training on all 24 scenes. Training in a scene does not guarantee success on new routes there, and performance drops further in unseen scenes. Safety also worsens in the held-out setting: DAgger's CR rises from 16.12% to 29.63%, PPO's from 18.98% to 25.45%, and WR increases for both. Cross-scene transfer must therefore preserve safe obstacle avoidance as well as goal reaching. IndustryNav's A* data collection enables studies of demonstration coverage



Figure 4: GPT-6 Astra turns right to avoid nearby pallets at a goal distance of 4.25 m. The episode later finishes 0.60 m from the goal.

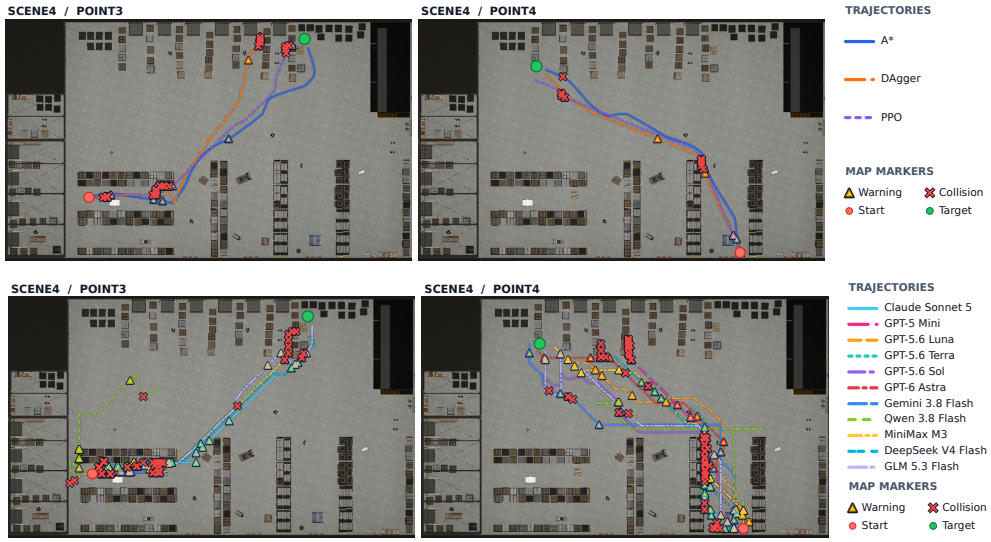

Figure 5: Matched trajectories on Scene 4 Point 3 (left) and Point 4 (right). The top row compares A*, DAgger, and PPO; the bottom row compares eleven VLLM agents. A* uses privileged top-down geometry, whereas all displayed maps are post-hoc diagnostics for the learned policies and VLLMs. Circles mark starts and targets, triangles indicate warnings, and crosses indicate collisions.

and policies trained for both success and safety. These 24-task results remain descriptive; matched training budgets and repeated seeds are needed to isolate the effect of scene familiarity.

3.3 Case Analysis

To examine how visual observations influence navigation decisions, we analyze a representative GPT-6 Astra case in Fig. 4. The model detects pallets obstructing the previously clear route and turns right to seek clearance, despite the steady progress recorded in its action-state-observation memory. This behavior connects local obstacle recognition to action selection: previous progress does not guarantee that the next forward step remains traversable. The episode subsequently finishes only 0.60 m from the goal, showing how reassessing local geometry as new observations arrive can support goal-directed navigation. Additional cases are presented in the Appendix.



Figure 6: Failure modes pooled across 845 VLLM and policy failures.

Failures are often more than near misses. In Fig. 6, obstacle stalls and limited net goal progress account for roughly 57% of failed episodes, while near-goal or drift cases account for 10.4%. Improving the final

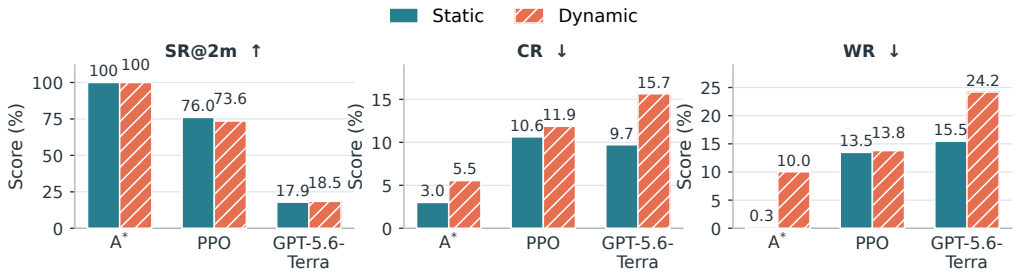

Figure 7: Static and dynamic navigation performance for A*, PPO, and GPT-5.6-Terra. We compare task success using SR@2m and safety using collision ratio (CR) and warning ratio (WR). Higher SR@2m is better, while lower CR and WR are better. Hatched bars denote the dynamic setting. Across all three approaches, dynamic obstacles raise CR and WR despite little change in SR@2m.

approach alone would leave many route-choice and obstacle-recovery failures unresolved. Model-wise distributions and representative cases appear in Appendix C.2.3.

Navigation-specific policies and general-purpose agents expose complementary gaps. The matched Scene 4 tasks in Fig. 5 expose both gaps. On P3, A* succeeds without a collision, whereas PPO reaches the goal after 15 collision steps; DAgger and ten of eleven VLLMs fail. On P4, all three navigation baselines succeed, but only three VLLMs do. In these cases, task-trained policies find viable routes more often than general-purpose agents, yet incur more unsafe interactions than privileged A*.

These cases lead to the following insight:

> *Task-specific navigation policies recover feasible routes more consistently than general-purpose VLLM agents, yet privileged geometric planning remains substantially cleaner in efficiency and safety.*

3.4 Ablation Study

We compare A*, PPO, and GPT-5.6-Terra in static and dynamic settings. We then use Gemini-3.8-Flash to test RGB, memory, and optional minimap input; the map-free interface remains the default. Further results appear in the Appendix.

Dynamics expose risks hidden by success rates. Fig. 7 shows that dynamics affect success and safety differently. A* preserves 100% SR@2m, but its CR increases from 3.0% to 5.5% and its WR rises from 0.3% to 10.0%. PPO exhibits only a modest success decrease, from 76.0% to 73.6%, while both safety ratios increase. GPT-5.6-Terra has similar success in the two settings, but its CR increases from 9.7% to 15.7% and its WR from 15.5% to 24.2%. Judged by success alone, these agents would appear largely unaffected by moving obstacles. Their trajectories instead show greater exposure to collisions and nearby hazards. Because collisions do not end the current evaluation, stable SR demonstrates continued goal reaching despite these interactions; it does not establish robustness in settings where the first collision could prevent further navigation.

Role of action-state-observation memory. Tab. 4 shows that temporal context consistently improves navigation under matched visual inputs. With both RGB and the minimap, adding five-step memory raises SR@2m from 50.01% to 81.26%, ES from 50.62% to 75.37%, and DR from 76.88% to 93.32%, while reducing CR from 6.37% to 4.44%. The RGB-only comparison shows an even larger success gap, from 9.38% to 59.38%. Memory therefore helps the agent connect current observations with recent outcomes, reducing repeated ineffective actions and supporting longer-horizon progress. We also explore the impact of the memory length in the Appendix.

Role of the top-down minimap. The minimap provides privileged global-layout information that can be hard to obtain or keep current in a working warehouse; it is an ablation, not the default input. In Tab. 4, adding it to RGB and five-step memory raises SR@2m from 59.38% to 81.26%, ES from 54.37% to 75.37%, and DR from 83.34% to 93.32%, while slightly lowering CR and WR. Without memory, SR@2m rises from 9.38% to 50.01%. Yet the minimap-only setting reaches 62.51% SR@2m with 25.46% CR and 19.80% WR: global layout helps routing but cannot replace local visual evidence for obstacle avoidance. The map-free default instead tests spatial reasoning

Table 4: Ablation results for Gemini-3.8-Flash in dynamic IndustryNav scenes. Check marks indicate enabled inputs; a check mark under Memory denotes access to the five most recent action-state-observation records. All scores are percentages. SR denotes Success Ratio; DR, Distance Ratio; ES, Efficiency Score; CR, Collision Ratio; and WR, Warning Ratio. Higher is better for SR, DR, and ES, while lower is better for CR and WR.

Ego RGB	Minimap	Memory	Task Success (%)				Efficiency (%)	Safety (%)	
			SR@2m ↑	SR@5m ↑	SR@10m ↑	DR ↑	ES ↑	CR ↓	WR ↓
✓			9.38	12.50	17.71	40.86	16.37	7.40	**9.48**
	✓		62.51	65.63	70.84	81.14	63.43	25.46	19.80
✓	✓		50.01	56.25	67.71	76.88	50.62	6.37	13.46
✓		✓	59.38	62.50	73.96	83.34	54.37	4.85	13.62
✓	✓	✓	**81.26**	**87.50**	**95.84**	**93.32**	**75.37**	**4.44**	13.34

from RGB and numerical goal cues. Only fixed-visual-input memory comparisons isolate one factor, since modality changes use different prompt families. The low WR of RGB-only navigation without memory reflects limited forward exposure under its low success rate, not safer behavior. The corresponding GLM-5.3-Flash ablation follows the same evaluation protocol and is reported in Appendix Tab. 5.

4 Related Work

Spatial Reasoning of VLLMs. Recent advances in VLLMs have explored various technical strategies to enhance spatial reasoning capabilities, which can be broadly categorized three pillars: *i) explicit geometric pretraining* using 3D data [65, 68] or annotated scene datasets [8, 34], *ii) depth-aware perception* for enhanced 2D understanding [9, 16, 21, 56, 57], and *iii) structured relational reasoning* through scene graphs [5, 32, 52, 60].

Spatial Reasoning Benchmarks. Evaluation on spatial reasoning has progressed from passive geometric reasoning [25] to complex embodied interactions [3, 42]. *Passive perception benchmarks* focus on visual grounding [24, 48], spatial relationship inference [32], and multi-view scene layout understanding [9, 16] without any interaction with the environment [33, 66]. Active exploration benchmarks add multi-step navigation, incorporating strategic search and question-driven exploration [35], generalization to unseen scenes [11, 27], and integrate reasoning for embodied QA [13], emphasizing exploration efficiency under simplified dynamics [6]. More recent efforts address domain-specific needs, such as outdoor navigation [22], urban spatial understanding [65], and augmented reality tasks focusing on spatial anchoring and cross-reality alignment [14].

Agent Navigation. Agent navigation aims to equip embodied models with the ability to perceive, reason, and act in complex environments [69]. Earlier research focused on instruction-following reasoning [4], command understanding [1], trajectory generation [40], and environment comprehension [47]. Subsequent research has concentrated on exploration-based reasoning [13] using search strategies [10, 67] and exploration policies [31, 63]. More recent advances introduce structured reasoning paradigms that utilize hierarchical planning [60, 61], topological reasoning [19], and semantic graphs [23].

Our work, IndustryNav, builds on these foundations by introducing a dynamic industrial simulation environment for navigation and safety evaluation, active interaction, and policy learning.

5 Conclusion

IndustryNav provides 24 dynamic Unity warehouses for evaluating closed-loop navigation and training policies. Its safety metrics reveal risks hidden by success rates: moving obstacles increase collisions and warnings even when goal-reaching changes little, while the strongest VLLM still trails a task-trained policy. The environment also supports automatic expert-trajectory collection and policy training. Yet DAgger and PPO perform worse on held-out layouts, highlighting a cross-scene transfer gap. A key limitation is the sim-to-real gap: simulated appearance, motion, and interactions do not fully capture real warehouses. Future work should validate these findings beyond simulation; further limitations appear in Appendix F.

REFERENCES

[1] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *CVPR*, pp. 3674–3683, 2018.

[2] Anthropic. System Card: Claude Sonnet 5, June 2026. URL https://www.anthropic.com/claude-sonnet-5-system-card.

[3] Daichi Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. Scanqa: 3d question answering for spatial scene understanding. In *CVPR*, pp. 19129–19139, 2022.

[4] Christian Bizer and Andreas Schultz. The r2r framework: Publishing and discovering mappings on the web. *COLD*, 665:97–108, 2010.

[5] Wenxiao Cai, Iaroslav Ponomarenko, Jianhao Yuan, Xiaoqi Li, Wankou Yang, Hao Dong, and Bo Zhao. Spatialbot: Precise spatial understanding with vision language models. In *ICRA*, pp. 9490–9498. IEEE, 2025.

[6] Matthew Chang, Gunjan Chhablani, Alexander Clegg, Mikael Dallaire Cote, Ruta Desai, Michal Hlavac, Vladimir Karashchuk, Jacob Krantz, Roozbeh Mottaghi, Priyam Parashar, et al. Partnr: A benchmark for planning and reasoning in embodied multi-agent tasks. In *ICLR*, 2025.

[7] Devendra Singh Chaplot, Ruslan Salakhutdinov, Abhinav Gupta, and Saurabh Gupta. Neural topological slam for visual navigation. In *CVPR*, pp. 12875–12884, 2020.

[8] Boyuan Chen, Zhuo Xu, Sean Kirmani, Brain Ichter, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In *CVPR*, pp. 14455–14465, 2024.

[9] 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 *NeurIPS*, volume 37, pp. 135062–135093, 2024.

[10] Kai Cheng, Zhengyuan Li, Xingpeng Sun, Byung-Cheol Min, Amrit Singh Bedi, and Aniket Bera. Efficienteqa: An efficient approach for open vocabulary embodied question answering. *arXiv preprint arXiv:2410.20263*, 2024.

[11] Jaewoo Choi, Youngwoo Yoon, Hyobin Ong, Jaehong Kim, and Minsu Jang. Lota-bench: Benchmarking language-oriented task planners for embodied agents. In *ICLR*, 2024.

[12] Guangzhao Dai, Qi Wu, and Bin Zhu. Gpt-6-astra in a navigation workflow: Behavioral analysis in zero-shot vision-and-language navigation in continuous environments. *arXiv preprint arXiv:2609.20116*, 2026.

[13] Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Embodied question answering. In *CVPR*, pp. 1–10, 2018.

[14] Erik Daxberger, Nina Wenzel, David Griffiths, Haiming Gang, Justin Lazarow, Gefen Kohavi, Kai Kang, Marcin Eichner, Yinfei Yang, Afshin Dehghan, et al. Mm-spatial: Exploring 3d spatial understanding in multimodal llms. In *ICCV*, pp. 7395–7408, 2025.

[15] DeepSeek-AI. DeepSeek V4 Technical Documentation, April 2026. URL https://fe-static.deepseek.com/chat/transparency/deepseek-V4-model-card-EN.pdf.

[16] Mengfei Du, Binhao Wu, Zejun Li, Xuanjing Huang, and Zhongyu Wei. EmbSpatial-bench: Benchmarking spatial understanding for embodied tasks with large vision-language models. In *ACL*, pp. 346–355, 2024.

[17] 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 *ECCV*, pp. 148–166, 2024.

[18] Google DeepMind. Gemini 3.8 Flash Model Card, September 2026. URL https://deepmind.google/models/model-cards/gemini-3-8-flash/.

[19] Christian Grévisse, Claude Braun, and José Batista da Costa. Autotag & tagmap: Llm-powered moodle plugins for pedagogical alignment checks. *SN Computer Science*, 6(7):1–11, 2025.

[20] Peter E. Hart, Nils J. Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2): 100–107, 1968. doi: 10.1109/TSSC.1968.300136.

[21] Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. *Advances in Neural Information Processing Systems*, 36:20482–20494, 2023.

[22] Zhenghang Hou, Weiping He, and Shuxia Wang. Enhancing visualization and interaction of complex spatial data through augmented reality. *The International Journal of Advanced Manufacturing Technology*, 134(11):5891–5906, 2024.

[23] Chenguang Huang, Oier Mees, Andy Zeng, and Wolfram Burgard. Visual language maps for robot navigation. In *ICRA*, pp. 10608–10615. IEEE, 2023.

[24] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *CVPR*, pp. 6700–6709, 2019.

[25] Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *CVPR*, pp. 2901–2910, 2017.

[26] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

[27] Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Michael Lingelbach, Jiankai Sun, et al. Behavior-1k: A benchmark for embodied ai with 1,000 everyday activities and realistic simulation. In *CoRL*, pp. 80–93. PMLR, 2023.

[28] Yifan Li, Yuhang Chen, Anh Dao, Lichi Li, Zhongyi Cai, Zhen Tan, Tianlong Chen, and Yu Kong. Industryeqa: Pushing the frontiers of embodied question answering in industrial scenarios. *arXiv preprint arXiv:2505.20640*, 2025.

[29] Yifan Li, Zhixin Lai, Wentao Bao, Zhen Tan, Anh Dao, Kewei Sui, Jiayi Shen, Dong Liu, Huan Liu, and Yu Kong. Visual large language models for generalized and specialized applications. *arXiv preprint arXiv:2501.02765*, 2025.

[30] Zhimin Li, Pan Wang, Jingxian Chen, Yuantao Tang, Anthony Chen, Qian Lou, and Jingtong Hu. Defending wearable vlms against private attribute inference. *arXiv preprint arXiv:2608.28691*, 2026.

[31] Mingfu Liang, Ying Wu, et al. Toa: Task-oriented active vqa. *Advances in Neural Information Processing Systems*, 36:54061–54074, 2023.

[32] 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 *EMNLP*, pp. 17028–17047, 2024.

[33] Wufei Ma, Haoyu Chen, Guofeng Zhang, Yu-Cheng Chou, Jieneng Chen, Celso de Melo, and Alan Yuille. 3dsrbench: A comprehensive 3d spatial reasoning benchmark. In *ICCV*, pp. 6924–6934, 2025.

[34] Wufei Ma, Luoxin Ye, Celso M de Melo, Alan Yuille, and Jieneng Chen. Spatialllm: A compound 3d-informed design towards spatially-intelligent large multimodal models. In *CVPR*, pp. 17249–17260, 2025.

[35] Arjun Majumdar, Anurag Ajay, Xiaohan Zhang, Pranav Putta, Sriram Yenamandra, Mikael Henaff, Sneha Silwal, Paul Mcvay, Oleksandr Maksymets, Sergio Arnaud, et al. Openeqa: Embodied question answering in the era of foundation models. In *CVPR*, pp. 16488–16498, 2024.

[36] MiniMax. MiniMax M3: Frontier Coding, 1M Context, Native Multimodality — All in One Model, June 2026. URL https://www.minimax.io/blog/minimax-m3.

[37] OpenAI. GPT-5 Mini Model. OpenAI API documentation, 2025. URL https://developers.openai.com/api/docs/models/gpt-5-mini.

[38] OpenAI. Models. OpenAI API documentation, 2026. URL https://developers.openai.com/api/docs/models.

[39] Ruslan Partsey, Erik Wijmans, Naoki Yokoyama, Oles Dobosevych, Dhruv Batra, and Oleksandr Maksymets. Is mapping necessary for realistic pointgoal navigation? In *CVPR*, pp. 17232–17241, 2022.

[40] Yuankai Qi, Qi Wu, Peter Anderson, Xin Wang, William Yang Wang, Chunhua Shen, and Anton van den Hengel. Reverie: Remote embodied visual referring expression in real indoor environments. In *CVPR*, pp. 9982–9991, 2020.

[41] Qwen Team. Latest model: Qwen3.8-Flash. QwenCloud model documentation, August 2026. URL https://docs.qwencloud.com/developer-guides/getting-started/latest-model.

[42] Jiawei Ren, Yan Zhuang, Xiaokang Ye, Lingjun Mao, Xuhong He, Jianzhi Shen, Mrinaal Dogra, Yiming Liang, Ruixuan Zhang, Tianai Yue, et al. Simworld: An open-ended realistic simulator for autonomous agents in physical and social worlds. *arXiv preprint arXiv:2512.01078*, 2025.

[43] Stéphane Ross, Geoffrey J. Gordon, and J. Andrew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 627–635. PMLR, 2011. URL https://proceedings.mlr.press/v15/ross11a.html.

[44] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In *CVPR*, pp. 9339–9347, 2019.

[45] Alexander Sax, Jeffrey O Zhang, Bradley Emi, Amir Zamir, Silvio Savarese, Leonidas Guibas, and Jitendra Malik. Learning to navigate using mid-level visual priors. In *CoRL*, pp. 791–812, 2020.

[46] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

[47] Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In *CVPR*, 2020.

[48] 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 *CVPR*, pp. 15768–15780, 2025.

[49] Andrew Szot, Alexander Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Singh Chaplot, Oleksandr Maksymets, et al. Habitat 2.0: Training home assistants to rearrange their habitat. In *NeurIPS*, volume 34, pp. 251–266, 2021.

[50] Peter Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Adithya Jairam Vedagiri IYER, Sai Charitha Akula, Shusheng Yang, Jihan Yang, Manoj Middepogu, Ziteng Wang, et al. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. In *NeurIPS*, volume 37, pp. 87310–87356, 2024.

[51] Pan Wang, Siwei Song, Hui Ji, Siqi Cao, Heng Yu, Zhijian Liu, Huanrui Yang, Yingyan Celine Lin, Beidi Chen, Mohit Bansal, et al. From models to systems: A comprehensive survey of efficient multimodal learning. *Transactions on Machine Learning Research*, 2026.

[52] Abdelrhman Werby, Chenguang Huang, Martin Büchner, Abhinav Valada, and Wolfram Burgard. Hierarchical open-vocabulary 3d scene graphs for language-grounded robot navigation. In *ICRA*, 2024.

[53] Qiucheng Wu, Handong Zhao, Michael Saxon, Trung Bui, William Yang Wang, Yang Zhang, and Shiyu Chang. Vsp: Diagnosing the dual challenges of perception and reasoning in spatial planning tasks for mllms. In *ICCV*, pp. 2270–2280, 2025.

[54] Zhiheng Xi, Wenxiang Chen, Xin Guo, Wei He, Yiwen Ding, Boyang Hong, Ming Zhang, Junzhe Wang, Senjie Jin, Enyu Zhou, et al. The rise and potential of large language model based agents: A survey. *SCIS*, 68(2):121101, 2025.

[55] Fei Xia, Amir R Zamir, Zhiyang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson env: Real-world perception for embodied agents. In *CVPR*, pp. 9068–9079, 2018.

[56] Peijin Xie, Lin Sun, Bingquan Liu, Dexin Wang, Xiangzheng Zhang, Chengjie Sun, and Jiajia Zhang. Expand vsr benchmark for vllm to expertize in spatial rules. In *AAAI*, volume 39, pp. 8745–8752, 2025.

[57] Yifan Xu, Ziming Luo, Qianwei Wang, Vineet Kamat, and Carol Menassa. Point2graph: An end-to-end point cloud-based 3d open-vocabulary scene graph for robot navigation. In *ICRA*, pp. 2853–2860. IEEE, 2025.

[58] Jihan Yang, Shusheng Yang, Anjali W Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. In *CVPR*, pp. 10632–10643, 2025.

[59] Zeyuan Yang, Jiageng Liu, Peihao Chen, Anoop Cherian, Tim K Marks, Jonathan Le Roux, and Chuang Gan. Rila: Reflective and imaginative language agent for zero-shot semantic audio-visual navigation. In *CVPR*, pp. 16251–16261, 2024.

[60] Hang Yin, Xiuwei Xu, Zhenyu Wu, Jie Zhou, and Jiwen Lu. Sg-nav: Online 3d scene graph prompting for llm-based zero-shot object navigation. In *NeurIPS*, volume 37, pp. 5285–5307, 2024.

[61] Bangguo Yu, Hamidreza Kasaei, and Ming Cao. L3mvn: Leveraging large language models for visual target navigation. In *IROS*, pp. 3554–3560. IEEE, 2023.

[62] Z.ai. GLM-5.3-Flash: More intelligence with less compute, September 2026. URL https://autoclaw.z.ai/blog/model/glm-5.3-flash/.

[63] Xin Zeng, Haonan Luo, Zihang Wang, Sijia Li, Zhixuan Shen, and Tianrui Li. A continual learning approach for embodied question answering with generative adversarial imitation learning. In *ICASSP*, pp. 1–5. IEEE, 2025.

[64] Pingyue Zhang, Zihan Huang, Yue Wang, Jieyu Zhang, Letian Xue, Zihan Wang, Qineng Wang, Keshigeyan Chandrasegaran, Ruohan Zhang, Yejin Choi, Ranjay Krishna, Jiajun Wu, Fei-Fei Li, and Manling Li. Theory of space: Can foundation models construct spatial beliefs through active exploration? In *ICLR*, 2026.

[65] Weichen Zhang, Zile Zhou, Xin Zeng, Liu Xuchen, Jianjie Fang, Chen Gao, Jinqiang Cui, Yong Li, Xinlei Chen, and Xiao-Ping Zhang. Open3d-vqa: A benchmark for embodied spatial concept reasoning with multimodal large language model in open space. In *ACMMM*, pp. 12784–12791, 2025.

[66] Ziang Zhang, Zehan Wang, Guanghao Zhang, Weilong Dai, Yan Xia, Ziang Yan, Minjie Hong, and Zhou Zhao. Dsi-bench: A benchmark for dynamic spatial intelligence. *arXiv preprint arXiv:2510.18873*, 2025.

[67] Yong Zhao, Kai Xu, Zhengqiu Zhu, Yue Hu, Zhiheng Zheng, Yingfeng Chen, Yatai Ji, Chen Gao, Yong Li, and Jincai Huang. Cityeqa: A hierarchical llm agent on embodied question answering benchmark in city space. *arXiv preprint arXiv:2502.12532*, 2025.

[68] Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 3d-vla: a 3d vision-language-action generative world model. In *ICML*, pp. 61229–61245, 2024.

[69] Duo Zheng, Shijia Huang, Lin Zhao, Yiwu Zhong, and Liwei Wang. Towards learning a generalist model for embodied navigation. In *CVPR*, pp. 13624–13634, 2024.

[70] Gengze Zhou, Yicong Hong, and Qi Wu. Navgpt: Explicit reasoning in vision-and-language navigation with large language models. In *AAAI*, volume 38, pp. 7641–7649, 2024.

[71] Filippo Ziliotto, Tommaso Campari, Luciano Serafini, and Lamberto Ballan. Tango: training-free embodied ai agents for open-world tasks. In *CVPR*, pp. 24603–24613, 2025.