What Matters for Real-World Long-Horizon Robot Navigation?: An Experimental Study of Implicit Goals and Sparse Memory

Bokeon Suh * **Hyoseok Ju** * **Giseop Kim** *

* *Daegu Gyeongbuk Institute of Science and Technology, Daegu, Republic of Korea (e-mail: {bksuh, hyoseokju, gsk}@dgist.ac.kr)*

Abstract: For autonomous robots to operate effectively in the real world, they must simultaneously possess two distinct capabilities: interpreting implicit human instructions and managing long-horizon memory efficiently. While recent Retrieval-Augmented Generation based approaches have shown promise, prior studies have not fully addressed the performance trade-offs between query ambiguity and memory sparsity. We identify two underexplored factors: (i) implicit goals, in which queries omit explicit object names, and (ii) sparse memory, in which only selected keyframes are stored. We present a paired dataset of explicit and implicit queries and a keyframe-based memory policy. Implicit phrasing drops success rate by up to 40 percentage points versus matched explicit queries (Short), and the keyframe policy reduces stored entities by 31% versus a fixed-interval baseline (Long).

Keywords: AI-powered robotics, Autonomous navigation, Human-robot interaction, Robot learning and adaptation, Robot perception and sensing, LLMs for modeling and control

1. INTRODUCTION

Advances in Large Language Models (LLMs) and Vision Language Models (VLMs) have enabled robots to interpret visual observations and reason with everyday knowledge. Despite these advances, bridging the semantic gap between abstract, implicit user queries and concrete objects in the environment remains a significant challenge. In real-world human-robot interaction, users frequently express intentions implicitly rather than through direct commands.

Consider a scenario where a person with a severe leg injury asks, "How can I go upstairs?" An agent relying solely on literal interpretation might simply suggest the nearest vertical access, such as a staircase, elevator, or escalator. In contrast, an agent capable of implicit reasoning would naturally recognize the physical constraints imposed by the injury and exclude the staircase, recommending only an elevator or escalator as a plausible solution.

However, evaluating such capabilities is difficult because an existing benchmark, NaVQA (Anwar et al., 2025), is mostly composed of "explicit" instructions, which masks the difficulty of resolving indirect references or ambiguous commands. We expand the questions introduced in the previous work to construct a paired dataset that contrasts explicit instructions with implicit ones while preserving the same underlying intent. Beyond interpreting queries, navigating complex environments requires recalling and reasoning about past experiences, not just geometric mapping. While humans naturally revisit episodic memory to retrieve relevant information for goal achievement, giving robots similar capabilities requires maintaining a comprehensive and structured representation of the environment. Just as humans summarize details over time, maintaining a dense history over long horizons presents a significant computational burden for robots. To resolve this, approaches like ReMEmbR (Anwar et al., 2025) manage long-horizon memories by storing captioned visual histories in a database accessible to LLMs via Retrieval Augmented Generation (RAG).

However, as environment scale and operation duration grow, fixed-time-interval memory updates accumulate redundant data without bound. This unbounded growth burdens both storage and retrieval, as near-duplicate entries dilute RAG lookup precision and inflate the candidate pool the LLMs must reason over. To address this, we explore a keyframe-based memory maintenance strategy that keeps memory compact under long-horizon operation.

The main contributions of this paper are summarized as follows:

- We curate a paired explicit/implicit query benchmark of 28 instances spanning three environment scales (Short/Medium/Long), controlling for intent variability to isolate the effect of linguistic implicitness on retrieval and grounding.
- We demonstrate that implicit queries incur a 40 percentage-point success-rate drop relative to explicit queries even when query intent is correctly classified,

⋆ This work was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No.RS-2025-25420118), the InnoCORE program of the Ministry of Science and ICT (26-InnoCORE-01), and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No.RS-2026-25492530).

exposing a performance gap that persists beyond intent recognition.
- We show that keyframe-based memory admits a 31% entity reduction without accuracy loss at Short and Medium scales.

2. RELATED WORK

2.1 Embodied Question Answering

While standard Visual Question Answering (VQA) answers queries from passive video or static images (Antol et al., 2015), Embodied Question Answering (EQA) requires an agent to actively navigate its environment to resolve natural language queries (Das et al., 2018). ObjectNav, ImageNav, and PointNav prioritize reaching a specifically defined target, whether an object class, a reference image, or a coordinate point (Ieong and Tang, 2025). In contrast, EQA emphasizes retrieving semantic information to formulate an answer such as a location, duration, or state description, rather than merely reaching a goal.

Anwar et al. (2025) introduce NaVQA, a benchmark whose queries span diverse question types but are predominantly *explicit*. We call a query *explicit* when it directly names the target referent, as in "Where is an elevator?" or "Where is the restroom?" By contrast, an *implicit* query states only the user's intent or constraint and leaves the referent to be inferred, as in "Where can I move to another floor? I have a leg injury." (which implies an elevator) or "Where can I resolve my stomachache?" (which implies a restroom). Such *implicit* queries are comparatively underrepresented in NaVQA. As a result, the extent to which an EQA agent can bridge from underspecified intent to a concrete spatial referent has not been systematically evaluated. This work addresses that gap by directly comparing agent performance on matched *explicit* and *implicit* query pairs.

2.2 Large Language Model Driven Navigation

The multimodal capabilities of LLMs and VLMs have eased the interpretation of visual inputs into rich natural language descriptions. Furthermore, the extensive commonsense knowledge embedded within LLMs empowers agents to reason about their own actions and the environment. For instance, Zhou et al. (2024) demonstrate an LLM-based instruction-following agent capable of zero-shot sequential action prediction, encompassing high-level planning, sub-goal decomposition, and commonsense integration. Similarly, recent approaches have integrated VLMs directly into end-to-end navigation policies for next action selection within the current view (Nie et al., 2025; Goetting et al., 2024).

In the context of long-horizon tasks, however, directly ingesting the entire visual history into an LLM's context window proves computationally prohibitive and scales poorly with temporal duration. To mitigate this bottleneck, Wang et al. (2025) and Xie et al. (2024) have adopted RAG frameworks or topological memory structures. ReMEmbR (Anwar et al., 2025) utilizes vector databases to archive captioned visual histories, enabling the agent to retrieve contextually relevant memories via semantic similarity. However, the efficacy of these methods depends on

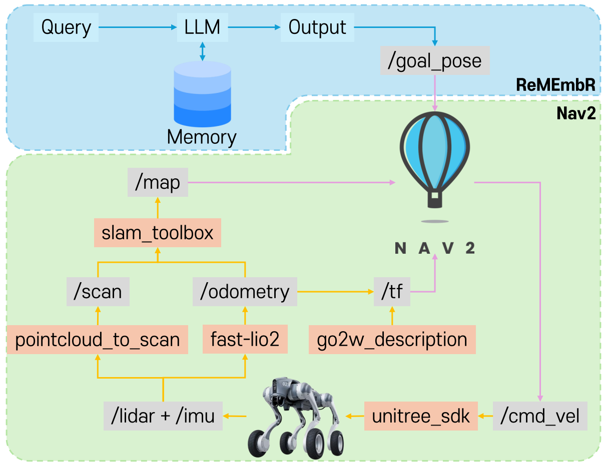


Fig. 1. Architecture of the proposed system.

the semantic alignment between the user's query phrasing and the stored captions. This study therefore examines how explicit versus implicit phrasing affects retrieval and navigation accuracy as the environment and memory scale grow.

3. METHODS

We employ the ReMEmbR framework (Anwar et al., 2025) integrated with the Navigation 2 (Nav2) Stack (Macenski et al., 2020), structured into two phases: building a semantic memory from exploration (Sec. 3.1) and executing navigation from natural language queries (Sec. 3.2). Fig. 1 illustrates the overall architecture.

3.1 Semantic Memory Building

As the robot explores the environment, it captures RGB image streams and odometry data at regular intervals. Rather than storing raw image data, a dense textual description is generated by the video captioner (Lin et al., 2023). These captions are then embedded via the `mxbai-embed-large-v1` model and stored in a vector database alongside their corresponding timestamps and global coordinates. This process results in a queryable history $H_{1:K}$, effectively transforming the physical environment into a semantic map accessible to LLMs.

3.2 Goal Inference and Navigation

Upon receiving a natural language query Q, the system employs an LLM to act as a reasoning agent. The LLM first classifies the underlying intent of Q. Once the query is identified as position-related, the agent retrieves relevant context from the stored history $H_{1:K}$ through iterative function calls and infers the target location (x, y) that best resolves the query.

Once the target coordinates (x, y) are determined, they are dispatched as a goal pose to the Nav2 stack. Nav2 utilizes a global planner to compute the shortest path on the static map, while a local controller dynamically adjusts velocity commands to avoid unforeseen obstacles.

Table 1. Examples of explicit and implicit query pairs used in the experiments.

Environment	Explicit Query	Implicit Query
Short	(Q1) Where is a fire extinguisher?	Where can I find something useful in case of fire?
	(Q2) Where is the trash can?	Where can I throw away my trash?
	(Q3) Where is an elevator?	Where can I move to another floor? I have a leg injury.
	(Q4) Where is the staircase?	Where can I move to another floor? I prefer walking.
	(Q5) Where is an umbrella?	Where can I find something useful in case of rain?
Medium	Where is the ATM?	Where can I get some money?
	Where is the auditorium?	Where can I hold a concert?
	Where is a photo zone?	Where can I take a photo?
	Where is a water fountain?	Where can I get some water?
	Where is an AED Kit?	Where can I get something useful in case of a heart attack?
	Where is the staircase?	Where can I move to another floor? I prefer walking.
Long	Where is a ping-pong table?	Where can I play ping-pong?
	Where is a sofa?	Where can I sit down?
	Where is the restroom?	Where can I resolve my stomachache?
	Where is a bike rack?	Where can I leave my bike?
	Where is a bookshelf?	Where can I store my books?
	Where is the information desk?	Where can I get some help?
	Where is the vending machine?	Where can I get some drinks?
	Where is a fire extinguisher?	Where can I find something useful in case of fire?
	Where is the trash can?	Where can I throw away my trash?
	Where is an elevator?	Where can I move to another floor? I have a leg injury.

Fig. 2. Overview of the three evaluation environments (Short, Medium, Long) with their corresponding navigation paths.

3.3 Keyframe-based Memory Construction

The baseline, ReMEmbR, updates the memory at fixed temporal intervals, which accumulates duplicate data and uses resources inefficiently when the robot remains stationary. Inspired by keyframe selection strategies in SLAM (Campos et al., 2021; Koide et al., 2019; Murai et al., 2025), we propose a spatial displacement-based update mechanism to mitigate this inefficiency.

4. EXPERIMENTAL SETUP

4.1 Problem Formulation

During the memory building phase, the robot accumulates a long-horizon history $H_{1:K} = \{h_1, h_2, \ldots, h_K\}$, where each entry h_k consists of caption, position, and timestamp. Given a natural language query Q, the system retrieves a relevant subset of memories to generate an answer:

$$A = f(Q, H_{1:K}) \tag{1}$$

To investigate the system's linguistic interpretability, we design two distinct categories of spatial queries:

- **Explicit Query** (Q_{exp}): A direct formulation that specifically names the target object (e.g., "Where is the fire extinguisher?").
- **Implicit Query** (Q_{imp}): An indirect formulation requiring pragmatic inference from context and prior knowledge (e.g., "Where can I find something helpful in case of fire?").

Since our study focuses on spatial navigation, the generated answer A for both query types is mapped to a metric coordinate in the environment:

$$A_{\text{exp}}, A_{\text{imp}} \mapsto \mathbf{p} = (x, y) \in \mathbb{R}^2 \tag{2}$$

where $\mathbf{p}$ denotes the predicted 2D location of the target. Table 1 lists a subset of query pairs used in the experiments.

Table 2. Details of the experimental environments.

Environment	**Duration (s)**	N_{question}	N_{entity}
Short Baseline	218.6	5	63
Short Keyframe			54
Medium Baseline	540.3	7	168
Medium Keyframe			115
Long Baseline	1093.2	16	330
Long Keyframe			227

4.2 Evaluation Metrics

Upon receiving a query, the system initially classifies it into one of the predefined answer types (e.g., position, text, duration, binary, or time). Since this study targets spatial reasoning under implicit instructions, we focus exclusively on the *position* category and assess robustness using the following two metrics.

Query Intent Accuracy (QIA) Implicitly phrased queries may lead to misclassification of the underlying intent (e.g., routing a position question to a text answer). QIA measures the accuracy of identifying the query as a spatial intent:

$$\text{QIA} = \frac{N_{\text{correct\_intent}}}{N_{\text{total\_spatial\_queries}}} \tag{3}$$

Success Rate (SR) A spatial prediction is considered successful if the Euclidean distance between the predicted location $\mathbf{p}_{\text{final}}$ and the ground truth $\mathbf{p}_{\text{goal}}$ falls within a specific threshold δ_{th}. In our experiments, we set $\delta_{\text{th}} = 7.0\,\text{m}$. The success rate is defined as:

$$\text{SR} = \frac{N_{\text{success}}}{N_{\text{total}}} \tag{4}$$

4.3 Implementation Details

As illustrated in Fig. 1, our experimental platform is the Unitree Go2W quadruped robot, equipped with an Orbbec Gemini 335L camera for RGB observations and a Livox Mid360 LiDAR. For state estimation, we utilize FAST-LIO2 (Xu et al., 2022) to obtain odometry. The 3D point cloud data is projected into 2D laser scans to construct an occupancy grid map of the environment. Upon receiving a goal position from the reasoning agent, the system passes control commands to the Unitree SDK to navigate the robot.

All computational tasks for the memory-building process were conducted on a desktop workstation equipped with an NVIDIA RTX 4090 GPU, an AMD Ryzen 9 9900X CPU, and 64 GB of system memory.

5. RESULTS

5.1 Datasets

To evaluate the robustness of the proposed system under varying memory loads, we conducted experiments across three environments with increasing temporal and spatial scales (Fig. 2). As shown in Table 2, longer trajectories produced a larger number of extracted semantic entities that were stored in the memory database.

Table 3. Quantitative evaluation results across environment scales.

Environment	**Query Type**	**QIA(%)**	**SR(%)**
Short	Explicit	100	100
	Implicit	100	60.00
Medium	Explicit	100	57.14
	Implicit	100	57.14
Long	Explicit	100	81.25
	Implicit	93.75	62.50

The standard NaVQA dataset contains 210 questions across five categories. An analysis of the 71 position-related questions reveals a strong class imbalance. Among these, 58 are explicit, 10 are implicit, and only 3 fall into a neutral category. The scarcity of implicit queries limits a meaningful assessment of the agent's ability to resolve indirect and pragmatically challenging commands.

To address this limitation, we created a paired query dataset where each explicit instruction is matched with an implicit instruction that expresses the same underlying intent. Table 1 lists a subset of query pairs used in each environment.

To evaluate the keyframe-based memory construction strategy, we built both the baseline memory and the keyframe memory along the same trajectory without any modification to the robot's exploration pattern. This design isolates the effect of the memory update policy and allows a direct comparison of retrieval behavior under identical environmental conditions.

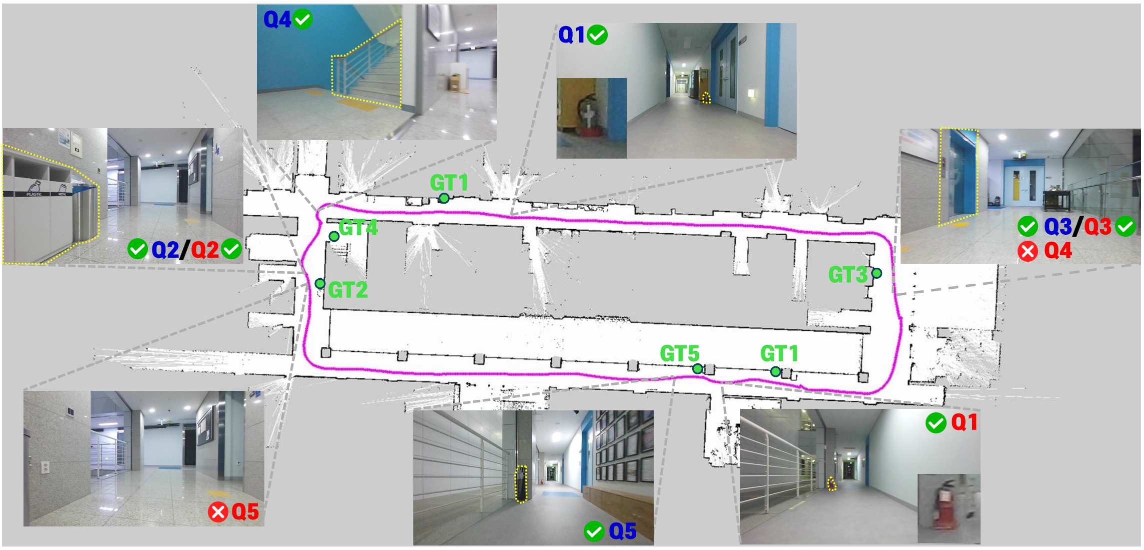
5.2 Qualitative Comparison

Fig. 3 illustrates the predicted goal positions for both baseline and keyframe-based memory in the Short environment, with the corresponding RGB frame visualized for interpretability. Ground-truth goals are green, intended target regions yellow, explicit queries blue, and implicit queries red; the query subset is listed in Table 1. The keyframe-based method still produces goal positions broadly consistent with the baseline even when the final success rate declines: although sparsity occasionally causes the agent to miss fine-grained distinctions required for long-horizon implicit queries, overall navigation behavior and predicted goal trends remain similar.

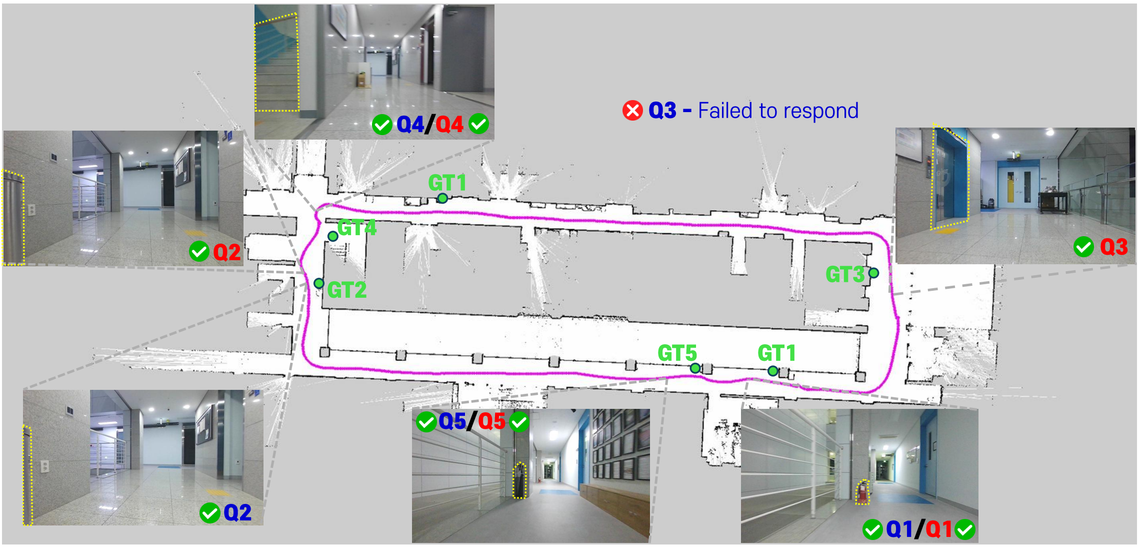
5.3 Quantitative Comparison

As shown in Table 3, **QIA** remains perfect for all explicit queries, indicating that clearly phrased instructions reliably map to the correct spatial intent regardless of memory size. In contrast, implicit queries exhibit a degradation in the Long environment. This specific failure originated from the query, "Where can I resolve my stomachache?", where the user intended to find a "restroom". However, the agent misclassified the intent as a general information question, outputting the textual response: "You can resolve a stomachache in a hospital", instead of providing spatial coordinates.

SR declines considerably for implicit queries in the Short and Long environments. In the Short environment, the implicit SR drops to 60% despite perfect intent recognition,



(a) Baseline memory



(b) Keyframe-based memory

Fig. 3. Qualitative results illustrating robot trajectories and predicted goal locations in the Short environment. Explicit queries (Blue), Implicit queries (Red), Ground Truth (Green), Expected target (Yellow). Qk and GTk denote the k-th Short-environment query in Table 1 and its corresponding ground-truth target.

suggesting that indirect formulations introduce reasoning complexity that hinders target isolation even when the task is identified as spatial. This gap persists in the Long environment, reflecting the higher cognitive load of resolving implicit references over larger trajectories and denser memories.

Keyframe-based memory substantially reduces the number of stored entities (e.g., a 31% reduction in the Long environment) because it prevents the database from accumulating redundant data during pauses in robot motion (Table 2). Fig. 4 presents the performance evaluation using the same query set as the baseline. In the Short and Medium environments, the keyframe-based approach yields comparable or even superior results while maintaining a smaller database. However, in the Long environment, the SR declines sharply for both explicit and implicit queries.

This suggests that while keyframe selection is efficient at smaller scales, sparse sampling in larger environments may discard transitional frames carrying semantic context critical for long-horizon navigation.

We decompose the mean query latency into three stages: the agent's iterative function-calling loop (Tool), memory retrieval (Mem), and final answer generation (Gen). As shown in Table 4, memory retrieval contributes negligibly (0.2–0.4% of total latency) and remains invariant to entity count. The dominant cost is the Tool stage, where multi-round reasoning over retrieved context determines overall latency. Notably, the reduced memory under the keyframe policy is associated with a shorter Tool-stage latency at Short and Medium scales. We find that the baseline issues more retrieval keywords per iteration than the keyframe variant. The resulting accumulation of retrieved entities in the chat history enlarges the prompt for subsequent iterations.

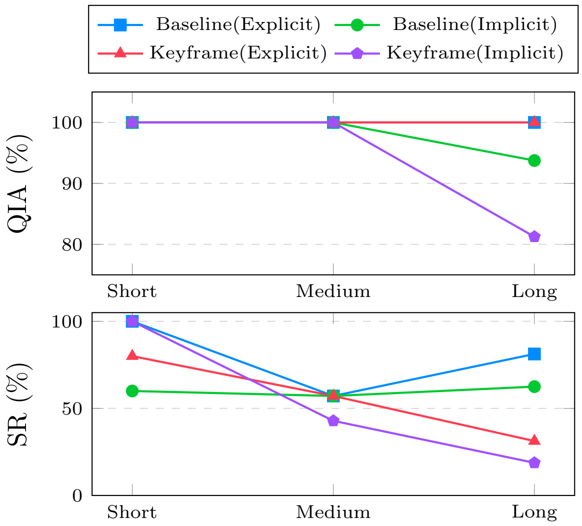


Fig. 4. Comparison of query intent accuracy (QIA, top) and success rate (SR, bottom) for four conditions on the Short, Medium, and Long environments.

Table 4. Mean query latency breakdown (s), averaged over all queries in each split. BL = baseline, KF = keyframe; -E = explicit, -I = implicit.

Env	Cond	Tool	Mem	Gen	Total
Short	BL-E	5.234	0.018	2.386	7.639
	KF-E	4.660	0.029	2.145	6.836
	BL-I	7.253	0.022	2.673	9.950
	KF-I	5.426	0.017	2.238	7.683
Medium	BL-E	7.848	0.023	3.153	11.025
	KF-E	6.110	0.023	3.035	9.169
	BL-I	7.069	0.023	3.120	10.213
	KF-I	5.817	0.021	3.360	9.199
Long	BL-E	7.060	0.023	2.502	9.587
	KF-E	6.525	0.024	2.433	8.983
	BL-I	5.917	0.017	2.706	8.641
	KF-I	6.523	0.027	2.641	9.193

6. CONCLUSION

We examined how explicit and implicit spatial queries affect long-horizon navigation when relying on a RAG-based memory system. Across environments of increasing scale, implicitly phrased queries led to lower success rates, even when the agent correctly identified the underlying spatial intent. This pattern indicates that indirect language continues to challenge current retrieval and reasoning pipelines. We also evaluated a keyframe-based memory update policy that reduced redundancy in long trajectories, producing a more compact set of stored entities while preserving overall navigation trends in smaller environments. Consequently, future work will focus on two directions: (1) improving the robustness of coordinate inference when queries are phrased abstractly, and (2) developing memory-construction strategies that preserve task-relevant semantic context while controlling database growth during extended deployments.

REFERENCES

Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C.L., and Parikh, D. (2015). Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Anwar, A., Welsh, J., Biswas, J., Pouya, S., and Chang, Y. (2025). Remembr: Building and reasoning over long-horizon spatio-temporal memory for robot navigation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, 2838–2845. IEEE.

Campos, C., Elvira, R., Rodríguez, J.J.G., Montiel, J.M., and Tardós, J.D. (2021). Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE transactions on robotics*, 37(6), 1874–1890.

Das, A., Datta, S., Gkioxari, G., Lee, S., Parikh, D., and Batra, D. (2018). Embodied question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1–10.

Goetting, D., Singh, H.G., and Loquercio, A. (2024). End-to-end navigation with vision language models: Transforming spatial reasoning into question-answering. *arXiv preprint arXiv:2411.05755*.

Ieong, I.T. and Tang, H. (2025). Multimodal perception for goal-oriented navigation: A survey. *arXiv preprint arXiv:2504.15643*.

Koide, K., Miura, J., and Menegatti, E. (2019). A portable three-dimensional lidar-based system for long-term and wide-area people behavior measurement. *International Journal of Advanced Robotic Systems*, 16(2), 1729881419841532.

Lin, J., Yin, H., Ping, W., Lu, Y., Molchanov, P., Tao, A., Mao, H., Kautz, J., Shoeybi, M., and Han, S. (2023). Vila: On pre-training for visual language models.

Macenski, S., Martín, F., White, R., and Ginés Clavero, J. (2020). The marathon 2: A navigation system. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. URL `https://github.com/ros-planning/navigation2`.

Murai, R., Dexheimer, E., and Davison, A.J. (2025). Mast3r-slam: Real-time dense slam with 3d reconstruction priors. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 16695–16705.

Nie, D., Guo, X., Duan, Y., Zhang, R., and Chen, L. (2025). Wmnav: Integrating vision-language models into world models for object goal navigation. *arXiv preprint arXiv:2503.02247*.

Wang, Z., Zhu, Y., Lee, G.H., and Fan, Y. (2025). Navrag: Generating user demand instructions for embodied navigation through retrieval-augmented llm. *arXiv preprint arXiv:2502.11142*.

Xie, Q., Min, S.Y., Ji, P., Yang, Y., Zhang, T., Xu, K., Bajaj, A., Salakhutdinov, R., Johnson-Roberson, M., and Bisk, Y. (2024). Embodied-rag: General non-parametric embodied memory for retrieval and generation. *arXiv preprint arXiv:2409.18313*.

Xu, W., Cai, Y., He, D., Lin, J., and Zhang, F. (2022). Fast-lio2: Fast direct lidar-inertial odometry. *IEEE Transactions on Robotics*, 38(4), 2053–2073.

Zhou, G., Hong, Y., and Wu, Q. (2024). Navgpt: Explicit reasoning in vision-and-language navigation with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 7641–7649.