2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

MemClaw-RAG: Memory-Driven Navigation and Adaptive Locomotion for Wheeled-Legged Robots in Dynamic Environments

Abstract— **Object-Goal Navigation in dynamic environments remains challenging as existing approaches rely primarily on reactive mapping that lacks the capacity to retain historical experience or establish structured memory associations. To address this, we introduce MemClaw-RAG, an embodied multimodal framework. MemClaw-RAG features three key innovations: (1) a Memory Graph Retrieval (MGR) module that integrates multimodal knowledge graphs for structured semantic association; (2) a SelfClaw cognitive module that orchestrates skill task scheduling and enhances historical memory retention; and (3) a Hybrid Adaptive Locomotion Policy (HALP) based on deep reinforcement learning that synergizes wheel-driven efficiency with legged dexterity. On Habitat benchmarks, MemClaw-RAG achieves an SR of 0.81 and an SPL of 0.51 on the Gibson and HM3D datasets. Notably, in the more challenging multi-layer environments of MP3D, our method achieves an SR of 0.76 and an SPL of 0.48, outperforming several representative memory-based and end-to-end approaches. Real-world deployment on a Unitree wheeled-legged robot confirms an average per-step inference latency of 55ms on a Jetson Orin, demonstrating stable navigation behavior during real-world deployment in dynamic environments.**

I. Introduction

Enabling wheeled-legged robots to autonomously locate and interact with specific objects in unfamiliar 3D environments, a task known as Object-Goal Navigation, is a foundational task for embodied intelligence in domestic and service scenarios [1]. This challenge is especially acute in dynamic environments, where complex spatial configurations hinder effective guidance, human–robot collaboration, and long-horizon reasoning. Specifically, the execution of sudden tasks and the handling of task interruptions remain critical bottlenecks bridging navigation to mobile manipulation. Furthermore, the lack of feedback at task endpoints often leads to failures in confirming task completion, undermining the reliability of autonomous operations.

We explore a multimodal graph RAG spatial retrieval mechanism to address these limitations. This approach facilitates the understanding of spatial navigation coordinate information, distinguishes similarities between object categories, enables fast retrieval, and provides precise feedback on navigation points. By integrating instruction comprehension with structured memory, our framework allows Wheeled-Legged Robots to interpret surroundings dynamically and update their knowledge base with interactive feedback for improved accuracy.

Traditional metric map methods [2], [3] suffer from pose errors in GPS-denied indoor settings. Without global positioning signals, these methods often fall into invalid mapping loops during local search, leading to significantly accumulated errors over time. End-to-end policies bypass mapping by directly predicting actions from visual input but often struggle to maintain spatial consistency and cross-modal grounding when targets are distant or ambiguous.

Recent works have combined vision-language models (VLMs) with commonsense reasoning and scene graphs, yet they face similar challenges. Existing VLM-based navigation systems often lack explicit memory mechanisms, resulting in an inability to effectively execute tasks when task interruptions occur. They often fail to retain historical context required for resuming complex tasks after unexpected pauses.

In this work, we propose MemClaw-RAG, which bridges retrieval-centric perception and real-world robotic autonomy. Our approach utilizes a Memory Graph Retrieval (MGR) mechanism to ground vision-language inputs in point cloud geometry, integrated with FAST-LIO2 [4] for robust LiDAR-inertial odometry in GPS-denied settings. This enables reliable local state estimation and improved spatial consistency without relying on global positioning signals. Our contributions are:

- Memory Graph Retrieval (MGR): A mechanism that fuses visual semantics with geometric constraints to resolve coordinate ambiguities, enabling fast spatial retrieval for navigation feedback.
- SelfClaw: A dual-process pathway for task-level scheduling that manages interruptions and prioritizes goals based on contextual memory and skill-based orchestration.
- Hybrid Adaptive Locomotion Policy (HALP): A reinforcement learning strategy that optimizes mobility for wheeled-legged platforms, ensuring stability across complex indoor topographies.

II. Related Work

A. Vision-and-Language Based Navigation

The landscape of Vision-and-Language Navigation (VLN) has transitioned from discrete environments in Matterport3D [5] toward continuous settings [6], [7], where agents increasingly leverage semantic mapping [8] and end-to-end policy learning [9]. While recent advances integrate Foundation Models to enhance open-world understanding [10], existing approaches often encounter limitations in dynamic environments when modeling the evolving object-region relationships under perceptual occlusions [11].

Navigation planning focused solely on instantaneous state-to-goal transitions tends to compromise the long-term consistency of spatial memory as task duration increases. Specifically, current frameworks lack robust mechanisms to handle

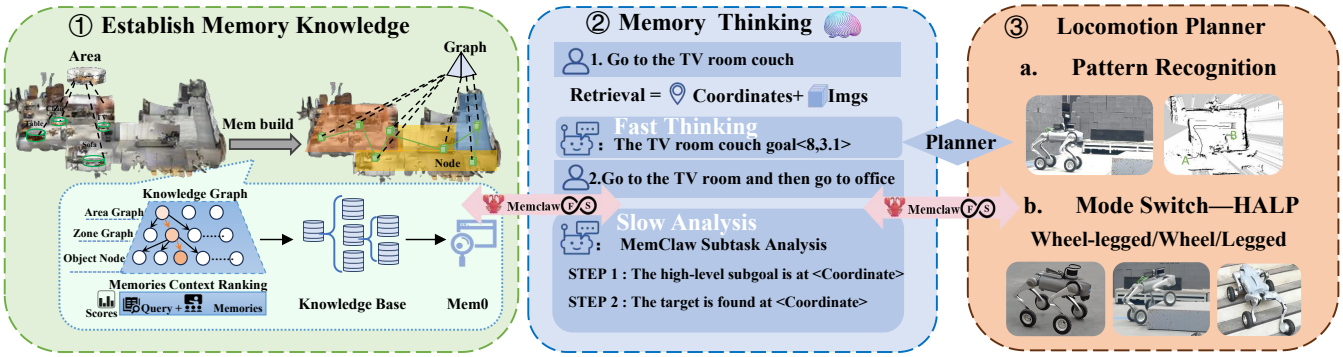

Fig. 1: The MemClaw-RAG system integrates a three-stage hierarchical workflow: (1) Establish Memories Knowledge, which constructs a spatial-semantic memory graph through multi-modal anchoring; (2) Memories Thinking, a dual-process module for reactive goal retrieval and high-level task reasoning; and (3) Locomotion Planner, which utilizes the HALP policy to translate high-level subgoals into low-level robot control commands, enabling robust navigation in complex environments.

task interruptions or provide structured feedback at task endpoints, which is critical for wheeled-legged robots transitioning from locomotion to OpenClaw-based manipulation. Although Retrieval-Augmented Generation (RAG) has been adopted to improve generalization [12], generic multimodal RAG typically remains at the image-text level, lacking 3D geometric consistency and temporally plastic structured memory [13], [14]. This results in semantic drift and spatial misalignment, particularly in GPS-denied indoor settings where cumulative odometry errors and invalid local search loops remain unresolved.

B. Task Planning and Memory in Embodied Navigation

Memory mechanisms in VLN have evolved from implicit hidden states to explicit semantic representations [15]. However, maintaining temporal consistency in dynamic environments remains a challenge [16]. While RAG-based embodied variants have emerged [17], [18], they largely rely on static indices and lack the dynamic task re-ranking capabilities required for complex OpenClaw workflows. When a robot encounters a higher-priority event or physical obstacle, current policies often suffer from memory fragmentation, failing to resume the original task sequence effectively due to rigid task graphs [19].

Adaptive navigation requires recognizing environmental patterns and switching control modes accordingly. Traditional navigation policies often operate under a monolithic control scheme, failing to distinguish between exploratory phases and precision goal-seeking phases [9]. Recent works explore mode switching for mobile manipulation [20], yet they typically decouple navigation and manipulation, lacking a unified memory structure to handle task endpoint verification. In complex indoor settings, the absence of pattern-adaptive modal switching leads to inefficient search loops. Thus, a significant gap remains in developing systems that can dynamically re-prioritize tasks, maintain spatial-temporal memory across interrupted sessions, and bridge the gap between long-range navigation and fine-grained OpenClaw manipulation through real-time mode recognition.

III. Methodology

A. Memory Graph Retrieval (MGR)

The MGR module forms the cognitive backbone of our system. We architect a dynamic graph $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t)$ where nodes represent spatial-semantic units. To maintain spatial consistency in GPS-denied environments, MGR leverages high-frequency poses from the FAST-LIO2 odometry to anchor visual features $\mathbf{f}_v^i$ and linguistic semantics $\mathbf{f}_l^i$ into a persistent global memory. The retrieval process is implemented using a vector database that supports retrieval-augmented reasoning to compute relevance scores, ensuring that object goal remains robust even during long-duration operations.

$$\mathbf{z}_i = \mathbf{W}_v \mathbf{f}_v^i + \mathbf{W}_l \mathbf{f}_l^i + \mathbf{W}_p \mathrm{PE}(\mathbf{p}_i), \tag{1}$$

where $\mathrm{PE}(\cdot)$ applies sinusoidal positional encoding to 3D coordinates. This linear combination establishes the foundation for cross-modal alignment, while subsequent non-linearity enhances representation capacity without introducing unnecessary complexity:

$$\mathbf{v}_i = \mathrm{ReLU}(\mathbf{z}_i). \tag{2}$$

Our memory retrieval aims to provide accurate results. Given a query $\mathbf{q}$, the system computes relevance scores based on the associations stored in memory:

$$p_i = \frac{e^{\langle \mathbf{q}, \mathbf{v}_i \rangle \gamma^{\Delta t_i}}}{\sum_j e^{\langle \mathbf{q}, \mathbf{v}_j \rangle \gamma^{\Delta t_j}}}, \tag{3}$$

where $\langle \cdot, \cdot \rangle$ denotes cosine similarity and $\Delta t_i = t - t_i$ tracks temporal relevance. This formulation effectively balances historical context with current observations, ensuring spatial consistency across navigation sessions.

When task interruptions occur, our Task-Level Re-ranking mechanism maintains memory integrity through exponential smoothing:

$$\mathbf{v}_i^{(t)} = \beta \mathbf{v}_i^{(t-1)} + (1 - \beta) \mathbf{v}_{\text{obs}}^{(t)} \tag{4}$$

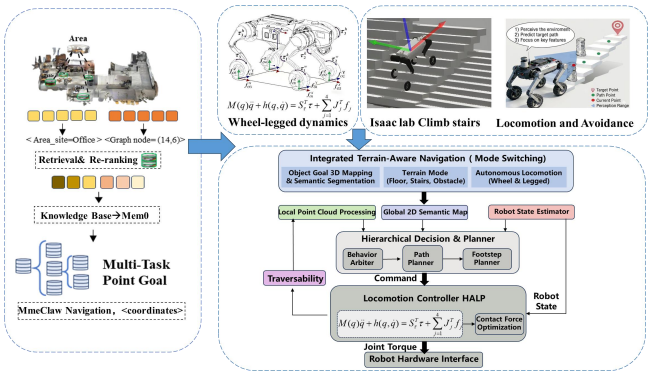

Fig. 2: The method integrates as follows: (1) building a spatial-semantic knowledge base through multi-modal anchoring; (2) performing dual-process cognitive reasoning for adaptive multi-task orchestration; and (3) implementing terrain-aware locomotion planning to translate high-level goals into robust control commands for navigation in complex environments.

where $\beta \in (0, 1)$ controls the temporal persistence of spatial representations. This update rule mitigates localization drift while preserving critical environmental relationships, enabling reliable navigation recovery without external positioning signals.

B. Lifelong Navigation Learning via Memory Thinking

Drawing inspiration from the skill-centric abstractions of the OpenClaw framework, we propose SelfClaw, a custom agent module that implements a dual-process cognitive architecture and robust task scheduling for wheeled-legged robots in GPS-denied indoor environments. To mitigate the memory bloat typically associated with long-term operational retention, we integrate Mem0 alongside a RAG vector database. This formulation empowers the agent with efficient associative memory recall, preventing cache overflow while maintaining context-aware retrieval.

The SelfClaw architecture coordinates operations through two distinct cognitive pathways:

1) Fast Thinking: This pathway handles immediate, reactive execution and local navigation. Through associative recall, it retrieves target coordinates from memory via personalized queries. It then plans optimized trajectories that explicitly account for the wheeled-legged platform's unique traversability, allowing the planner to account for the traversability characteristics of the wheeled-legged platform, including stair negotiation that standard wheeled robots cannot cross.

2) Slow Memories System: Operating as the high-level cognitive backend, this system governs long-term task orchestration, skill scheduling, and knowledge consolidation. It employs a Knowledge Inheritance Strategy to initialize new task policies using parameters extracted from similar historical tasks. This mechanism ensures highly efficient lifelong learning while strictly preventing catastrophic forgetting through parameter regularization.

This synergistic coordination allows the robot to maintain spatial consistency and seamlessly adapt to dynamic environments over extended periods. The formulation of this architecture is defined as:

$$\begin{aligned} p &= \arg\max_i \langle \mathbf{q}, \mathbf{v}_i \rangle, \quad \xi = \arg\min \sum (d + c_{\text{leg}}) \\ \theta_0 &= \frac{1}{K} \sum \theta_k, \qquad \mathcal{L} = \sum f(\theta - \theta_0)^2 \end{aligned} \tag{5}$$

where p is the retrieved goal position based on query q and memory v_i, and ξ is the optimal path minimizing distance d and legged terrain cost c_{leg}. For the second pathway, θ_0 represents the inherited policy parameters initialized from the top-K similar past tasks θ_k, and $\mathcal{L}$ is the regularization loss weighted by importance factor f to protect critical knowledge during task interruptions. This integration enables robust lifelong navigation learning with optimized memory usage without relying on external positioning signals.

C. Locomotion Planner with Mode Adaptation

The locomotion planner bridges high-level semantic goals from the memory module with low-level dynamic control commands for the wheeled-legged robot. This module is critical for negotiating complex indoor structures, such as stairs and uneven terrain, that are impassable for standard wheeled platforms. To ensure robust physical interaction, we employ a Hybrid Adaptive Locomotion Policy (HALP) based on deep reinforcement learning. The policy network learns a hybrid control scheme that synergizes wheel-driven efficiency for flat surfaces with legged dexterity for posture adjustment during stair climbing. This combined mode ensures continuous contact and stability while ascending elevation changes.

HALP leverages proprioceptive feedback and local elevation mapping to perceive terrain geometry and detect dynamic obstacles in real-time. The policy maps the observed state to action commands, optimizing for stability and progress while maintaining safety margins around obstacles. The learning objective is defined by a composite reward function that guides the robot to maintain balance during stair ascent and adjust its trajectory when obstacles are encountered:

$$a = \pi(s), \quad r = r_{\text{move}} + r_{\text{stab}} + r_{\text{obs}} \tag{6}$$

where a represents the action vector containing joint positions and wheel velocities, and s is the state vector comprising body orientation and terrain height. The reward r combines a progress incentive r_{move}, a stability penalty r_{stab}, and an obstacle avoidance term r_{obs}. The obstacle term increases the penalty as the distance to detected humans or objects decreases, prompting the policy to modify foot placement or wheel steering locally. This adaptive capability allows the robot to overcome physical barriers while maintaining safety during execution, ensuring robust autonomy in GPS-denied environments.

IV. EXPERIMENTS

This section addresses the following research questions through comprehensive evaluation:

- **Q1**: How does the proposed framework perform regarding memory retrieval accuracy for target points and navigation success rate in simulated environments?
- **Q2**: What is the impact of each module on the success rate of sub-task orchestration and navigation execution, as revealed through ablation studies?
- **Q3**: How effectively does the agent module combined with reinforcement learning control a physical wheeled-legged robot for stair climbing, standard navigation, and obstacle avoidance in real-world scenarios?

A. Experimental Setup and Evaluation Metrics

1) Simulation Setup and Datasets: We evaluate the proposed MemClaw-RAG using the Habitat simulator [6] under two primary experimental configurations. The first combines Gibson [21] (72 high-quality reconstructed real-world environments) and HM3D-Semantics v0.1 [22] (1,000 semantically annotated 3D scenes) into a unified dataset. The second leverages the Matterport3D (MP3D) [5] dataset, which features complex multi-layer indoor environments. Together, these datasets provide a rigorous testbed with diverse architectural layouts and semantic complexities. For real-world validation (Q3), MemClaw-RAG is deployed on a physical wheeled-legged robot operating in an office environment that includes stairs.

2) Baseline Comparisons: To comprehensively evaluate our approach, we compare it against several representative mainstream methods. For VLFM [23] and LROGNav [24], we report the official results from their respective papers using the standard Habitat ObjectNav evaluation protocol. For methods lacking complete reported metrics under our specific multi-task settings, such as RGANAV [25] and NavA3 [26], we re-implemented and trained them using identical RGB-D input resolutions, action spaces, reward shaping, and evaluation scripts based on their public codebases. This ensures a fair comparison across both modular and end-to-end paradigms.

3) Evaluation Metrics: Following standard Habitat ObjectNav protocols, we employ three primary metrics to quantify both navigation performance (Q1) and lifelong knowledge retention (Q2): Success Rate (SR): The fraction of episodes where the agent successfully navigates to within 1.0 m of the target object. Success-weighted Path Length (SPL): The SR weighted by path efficiency, defined as $\frac{1}{N}\sum_{i=1}^{N} S_i \frac{l_i^*}{\max(l_i^*, P_i)}$, where S_i is the binary success indicator, l_i^* is the optimal path length, and P_i is the actual path length taken by the agent. Oracle Success Rate (OSR): The fraction of episodes where the agent ever enters the 1.0 m success radius around the target at any point during the episode, which effectively measures exploration capability. Specifically, for answering Q1, the absolute values of SR, SPL, and OSR quantify the agent's navigation accuracy and path efficiency in single-task scenarios. For answering Q2, we analyze the retention of these identical metrics across sequential task streams to demonstrate the system's robust task orchestration and resistance to catastrophic forgetting.

TABLE I: Quantitative Evaluation of Memory-Enhanced ObjectNav Models

Method	Gibson + HM3D			MP3D		
	SR (↑)	SPL (↑)	OSR (↑)	SR (↑)	SPL (↑)	OSR (↑)
RIM [27]	0.68	0.37	0.75	0.61	0.29	0.68
SemExp [28]	0.54	0.20	0.61	0.36	0.14	0.43
RGANAV [25]	0.57	0.31	0.66	0.38	0.19	0.49
PONI [29]	0.74	0.41	0.80	0.32	0.12	0.39
LROGNav [24]	0.74	0.43	0.81	0.50	0.19	0.56
VLFM [23]	0.68	0.41	0.75	0.36	0.18	0.43
NavA3 [26]	0.78	0.45	0.84	0.66	0.31	0.72
L3MVN [30]	0.66	0.32	0.72	–	–	–
MemClaw-RAG (Ours)	**0.81**	**0.51**	**0.89**	**0.76**	**0.48**	**0.82**

4) Implementation Details.: To ensure a fair comparison, all evaluated methods utilize consistent input representations: a language embedding $\mathbf{F}_{\text{txt}} \in \mathbb{R}^{768}$ extracted from the CLIP text encoder; point cloud features $\mathbf{T}_{\text{pc}} \in \mathbb{R}^{256\times512}$ processed from RGB-D observations (256 sampled points with 512-dimensional features); and a task-specific token $\mathbf{Q}_{\text{task}} \in \mathbb{R}^{512}$ for navigation goal encoding. For hardware deployment, the system runs on an NVIDIA Jetson AGX Orin mounted on a wheeled-legged Unitree Go2-W. The Vision-Language Model is optimized using TensorRT and INT4 quantization to achieve stable onboard performance. Local state estimation is provided by FAST-LIO2 [4], which fuses Livox Mid360 LiDAR data with IMU feedback to mitigate odometry drift in GPS-denied indoor corridors. The construction of the topological graph is governed by a spatial connectivity radius and a temporal edge decay factor, both of which are empirically tuned to optimize graph sparsity and memory efficiency. Crucially, the memory module integrates Mem0 with a RAG vector database to optimize long-term retention, while the OpenClaw framework is utilized for low-level locomotion control.

B. Comparison with Mainstream Methods

We compare MemClaw-RAG against several representative navigation methods as summarized in Table I. The results indicate that our approach achieves improved performance compared with several modular and end-to-end methods. Notably, on the Gibson and HM3D dataset, MemClaw-RAG achieves a Success Rate (SR) of 0.81 and a Success-weighted Path Length (SPL) of 0.51. In the more challenging Matterport3D (MP3D) environments, which feature complex multi-layer structures, our method achieves an SR of 0.76, achieving higher performance than the LROGNav and showing a substantial improvement in SPL compared to conventional modular approaches such as PONI.

Performance Analysis and System Innovations: The superior performance of MemClaw-RAG stems from the integration of the SelfClaw module and its unique memory architecture, which addresses the limitations of previous frameworks regarding linear memory growth and long-term task coordination. At its core, the system utilizes a dual-process cognitive architecture. The Fast Thinking pathway

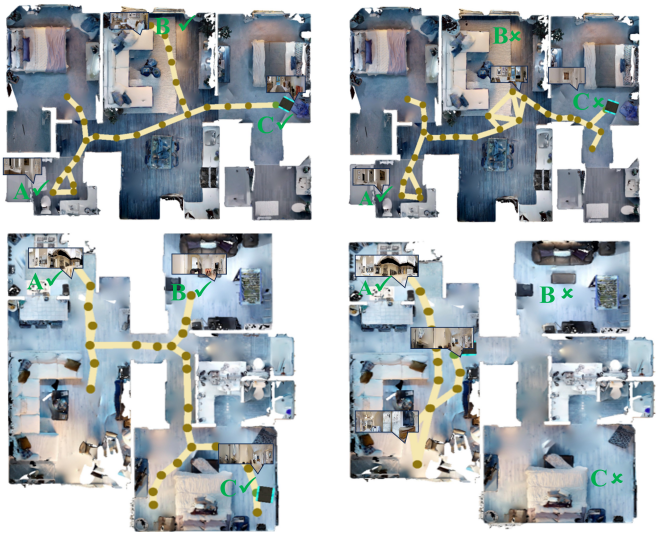

Fig. 3: Qualitative comparison of a sequential multi-goal navigation task. The left column illustrates the proposed MemClaw-RAG framework successfully orchestrating the continuous retrieval process with efficient trajectories, leveraging the SelfClaw module to prevent memory overwriting. Conversely, the right column demonstrates a traditional baseline that reaches target A but subsequently suffers from catastrophic forgetting and spatial confusion, resulting in erratic local loops and a failure to reach targets B and C.

handles reactive local navigation and immediate obstacle avoidance, while the Slow Memories System governs high-level task orchestration and long-term goal planning. This synergistic coordination is directly reflected in the high Oracle Success Rate (OSR) of 0.89 on Gibson and HM3D, suggesting improved exploration coverage and goal retrieval efficiency. To support this lifelong learning process without suffering from cache bloat, we integrate Mem0 with a RAG vector database. This associative memory recall mechanism allows the agent to efficiently retrieve contextual spatial graphs and maintain long-term operational retention without suffering from cache overflow.

Furthermore, MemClaw-RAG extends beyond simulation by incorporating a Hybrid Adaptive Locomotion Policy (HALP) specifically designed for wheeled-legged platforms. This morphology enables the robot to traverse complex physical topographies, such as indoor stairs, which pose challenges for the standard wheeled baselines evaluated in our study. The Slow Memory System facilitates spatial reasoning and adaptation, contributing to decision-making across diverse architectural layouts. This performance is observed when compared to baselines like LROGNav and NavA3 in complex spatial reasoning tasks. While LROGNav performs well in simpler layouts, its SR is 0.50 on the multi-layer MP3D dataset. The recent NavA3 achieves an SR of 0.66 and an SPL of 0.31 on MP3D. In comparison, our retrieval-augmented planning maintains spatial consistency across varying architectural complexities, yielding an SR of 0.76 and an SPL of 0.48 on MP3D. The SPL of MemClaw-RAG on MP3D represents an improvement over NavA3 (0.48 vs. 0.31), suggesting that the dual-process cognitive architecture contributes to more efficient navigation paths. These results indicate that MemClaw-RAG is a robust embodied intelligence system suitable for practical deployment scenarios.

TABLE II: Ablation Study on Core Components

Model	HALP	MKG	SelfClaw	SR (↑)	SPL (↑)
Base Architecture				0.62	0.31
Locomotion	✓			0.65	0.34
Spatial Memory	✓	✓		0.73	0.42
Full MemClaw-RAG	✓	✓	✓	0.81	0.51

C. Ablation Studies

1) Core Component Ablation: We conduct additive ablations on the Gibson and HM3D datasets under the standard Habitat ObjectNav protocol, reporting Success Rate (SR) and Success-weighted Path Length (SPL). To ensure a rigorous evaluation, our base model (Base Architecture) is not an arbitrary baseline, but a stripped-down version of our framework. It utilizes the exact standard multimodal inputs defined in our implementation details (Section IV-A)—specifically the CLIP language embeddings and RGB-D point cloud features—processed through a vanilla recurrent policy and executed via a generic pure-wheeled kinematic model. From this base model, we progressively integrate the Hybrid Adaptive Locomotion Policy (HALP) for terrain-adaptive control, the Memory Knowledge Graph (MKG) for structured spatial-semantic memory, and finally the SelfClaw agent for fast-slow cognitive scheduling. This progressive ordering strictly isolates the marginal gain of each proposed component while keeping all other configurations entirely fixed.

Table II demonstrates that each module contributes consistently to the overall navigation success and path efficiency. Relying solely on immediate observations and standard wheeled kinematics, the base model struggles with both physical obstacles and long-horizon planning, yielding an SR of 0.62. The initial integration of HALP provides a foundational morphological advantage, increasing the SR to 0.65 by improving traversability over architectural irregularities and preventing early episode terminations caused by low-level locomotion instability. Building upon this physical robustness, the addition of the MKG significantly elevates the SR to 0.73 and the SPL to 0.42. This substantial gain proves that explicitly encoding temporally consistent, spatial-semantic relationships is vital, as it allows the agent to move beyond reactive obstacle avoidance and effectively reduce redundant exploration in cluttered scenes. Finally, the inclusion of the SelfClaw module brings the system to its peak performance. By leveraging associative memory recall backed by the Mem0 and RAG integration, SelfClaw ensures that the agent can seamlessly orchestrate tasks and replan after dynamic interruptions. This fast-slow cognitive scheduling effectively prevents cache bloat during extended

Fig. 4: Validation of MemClaw-RAG in simulation environments. Left: Scene understanding and visual perception in a physical indoor environment. Center: Real-time knowledge graph reasoning and memory retrieval (MGR) for task planning. Right: Robust target localization and navigation execution.

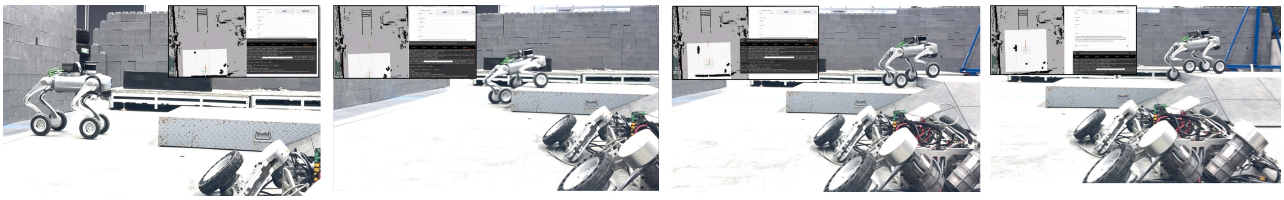

Fig. 5: Real-world stair-climbing evaluation of MemClaw-RAG on a wheeled-legged platform.

Fig. 6: Adaptive locomotion and avoidance of MemClaw-RAG in staircase terrains.

episodes, conclusively demonstrating that spatial memory representation, cognitive task orchestration, and hybrid locomotion adaptation are highly complementary rather than redundant.

2) Multimodal Retrieval Performance: In addition to embodied navigation, MemClaw-RAG preserves strong zero-shot retrieval capability through its Memory Knowledge Graph (MKG). By integrating Mem0 with a RAG vector database, the proposed method achieves superior accuracy compared to prior static multimodal knowledge graph systems. This optimization ensures that incorporating navigation modules does not sacrifice perception-level inference quality, maintaining high precision in object localization. Unlike retrieval-only systems, MemClaw-RAG directly integrates these retrieval results into the SelfClaw spatial reasoning pipeline. Retrieved object-language relations are injected into the policy for planning, ensuring strict consistency between perception and decision-making. Furthermore, the dynamic temporal decay module allows this retrieval-based knowledge to remain reliable in dynamic settings. If an object is moved, its graph edge weight is smoothly down-weighted, preventing the policy from generating trajectories toward outdated positions.

D. Real-World Robot Experiments

1) Hardware Configuration and Specifications: We deploy MemClaw-RAG on a Unitree Go2-W wheeled-legged robot. The perception suite is comprised of a Livox Mid360 LiDAR for comprehensive 360-degree 3D mapping, an Intel RealSense D435i RGB-D camera for semantic grounding, and an onboard IMU for high-frequency odometry correction. All embodied cognitive processing, including multimodal reasoning and task orchestration, is executed locally on an NVIDIA Jetson AGX Orin 32GB (12-core Arm CPU, 2048-core Ampere GPU) under JetPack 5.1 in MAXN (50W) mode. A Lightweight Communications and Marshalling (LCM) middleware is utilized to facilitate asynchronous, low-latency message passing between the perception-level memory retrieval, the SelfClaw cognitive architecture, and the HALP locomotion control modules. To ensure data consistency and strict reproducibility, precision sensor time synchronization is maintained across all input streams.

2) Experimental Setup and Performance Analysis: To validate real-world robustness, MemClaw-RAG was deployed across diverse physical indoor environments, encompassing research laboratories, corporate offices featuring staircases, and structural corridors. These testbeds were curated to subject the HALP module and the SelfClaw cognitive architecture to a wide spectrum of spatial layouts, variable lighting conditions, and unpredictable human-induced clutter. The robot was tasked with locating representative everyday object categories such as chairs, backpacks, and monitors. Task success was defined by a stringent physical proximity threshold to the target, ensuring the high precision demanded

for realistic physical interaction and potential OpenClaw manipulation tasks.

A critical requirement for stable wheeled-legged navigation is maintaining a sustainable control frequency across the high-level cognitive pipeline. The per-step latency of MemClaw-RAG is meticulously profiled into three primary stages: perception-level VLM grounding and structured semantic association within the Memory-Retrieval mechanism (MKG) require 32.0 ms; task-level orchestration via the SelfClaw cognitive architecture, including RAG-based memory retrieval, consumes 13.0 ms; and RL-based HALP planning for motor command generation requires 10.5 ms. The total onboard inference latency of 55.5 ms (approximately 18 Hz) confirms the real-time feasibility of our framework for complex indoor navigation. While the high-level cognitive pipeline operates at this frequency, the underlying motor control loop is managed independently at a much higher frequency via the LCM middleware to ensure continuous dynamic stability.

To support the Memory-Retrieval mechanism in dynamic settings, the MKG incorporates a temporal decay factor that prunes outdated spatial-semantic edges while reinforcing stable environmental cues. This ensures that the SelfClaw architecture consistently operates on the most relevant historical memory during long-horizon planning. Furthermore, our hardware profiling indicates the system's scalability: migrating to a 16GB Orin variant increases latency by 12–18% due to reduced memory bandwidth, while operating in the MAXQ (30W) power-efficiency mode adds an 8–12% delay. These results emphasize the necessity of the optimized RAG-based memory management within SelfClaw to eliminate cache overhead and maintain performance across different embedded configurations.

E. Discussion and Analysis

The integration of the three proposed modules enables MemClaw-RAG to effectively bridge the gap between high-level semantic reasoning and low-level physical execution. Quantitatively, our framework demonstrates competitive performance in the Habitat simulator, achieving a Success Rate (SR) of 0.81 and a Success-weighted Path Length (SPL) of 0.51 on a combined benchmark constructed from the Gibson and HM3D datasets. These results demonstrate competitive performance compared with existing memory-based and end-to-end baselines, particularly in complex multi-layer environments where effective task scheduling becomes increasingly important.

Despite the robustness demonstrated in real-world office environments, certain limitations remain. The responsiveness of the 18 Hz cognitive pipeline may be challenged by high-density, rapidly moving obstacles in extreme scenarios. Additionally, while the HALP module effectively handles stair transitions and uneven terrain, the framework's current semantic vocabulary is tailored for structured indoor navigation. Future work will focus on expanding these semantic representations and extending the hybrid locomotion capabilities to unstructured outdoor topographies, further advancing the state of universal, lifelong learning embodied agents.

V. Conclusion

We presented MemClaw-RAG, an embodied multimodal knowledge graph vision-language navigation architecture. By combining dynamic temporal reasoning via the MKG with reinforcement learning-based closed-loop planning (HALP) and fast-slow task orchestration (SelfClaw), MemClaw-RAG improves spatial grounding and navigation consistency. Rigorous evaluations in the Habitat simulator demonstrate competitive performance, achieving an SR of 0.81 on a combined benchmark constructed from the Gibson and HM3D datasets and 0.76 on the complex MP3D environments. Furthermore, physical deployment on a Unitree Go2-W wheeled-legged robot confirms efficient embedded inference (55.5 ms per step) and stable performance across varied indoor settings. Future work will explore unstructured outdoor environments, scale object vocabularies, and integrate larger foundational models to advance the development of universal, lifelong learning embodied agents.

References

[1] A. Wahid, A. Stone, K. Chen, B. Ichter, and A. Toshev, "Learning object-conditioned exploration using distributed soft actor critic," in *Proceedings of the 2020 Conference on Robot Learning*. Proceedings of Machine Learning Research, 2021, pp. 1–10.

[2] Y. Hong, Q. Wu, Y. Qi, C. Rodriguez Opazo, S. Gould *et al.*, "Vln (sic) bert: A recurrent vision-and-language bert for navigation," 2021.

[3] S. Chen, P.-L. Guhur, M. Tapaswi, C. Schmid, and I. Laptev, "Language conditioned spatial relation reasoning for 3d object grounding," *Advances in neural information processing systems*, vol. 35, pp. 20 522–20 535, 2022.

[4] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "FAST-LIO2: Fast direct LiDAR-inertial odometry," *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2053–2073, Aug. 2022.

[5] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niessner, M. Savva, S. Song, A. Zeng, and Y. Zhang, "Matterport3d: Learning from rgb-d data in indoor environments," in *International Conference on 3D Vision (3DV)*, 2017.

[6] M. Savva, A. Kadian, O. Maksymets, Y. Zhao, E. Wijmans, S. Jain, J. Straub, J. Liu, V. Koltun, J. Malik *et al.*, "Habitat: A platform for embodied ai research," in *IEEE International Conference on Computer Vision (ICCV)*, 2019.

[7] J. Krantz, E. Wijmans, Y. Zhu, A. Das, S. Lee, and D. Batra, "Beyond the nav-graph: Vision-and-language navigation in continuous environments," in *European Conference on Computer Vision (ECCV)*, 2020.

[8] D. S. Chaplot, D. Gandhi, S. Gupta, A. Gupta, and R. Salakhutdinov, "Object goal navigation using goal-oriented semantic exploration," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[9] E. Wijmans, A. Kadian, A. Morcos, S. Lee, I. Essa, D. Batra, and D. Parikh, "Embodied question answering using pointed end-to-end policy learning," in *IEEE International Conference on Computer Vision (ICCV)*, 2019.

[10] Y. Zhu, S. Gupta, Y. Yang, D. Batra, and D. Parikh, "Soon: Scenario oriented object navigation with graph-based exploration," in *IEEE International Conference on Computer Vision (ICCV)*, 2021.

[11] A. Ku, P. Anderson, R. Patel, E. Ie, and J. Baldridge, "Rxr: Cross-lingual embodied reasoning," in *Findings of the Association for Computational Linguistics (EMNLP)*, 2020.

[12] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. K"uttler, M. Lewis, W.-t. Yih, T. Rockt"aschel *et al.*, "Retrieval-augmented generation for knowledge-intensive nlp tasks," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[13] D. Edge, H. Trinh, N. Cheng, J. Bradley, A. Chao, A. Mody, S. Truitt, and J. Larson, "Local knowledge graph augmented retrieval for large language models," *arXiv preprint arXiv:2401.07904*, 2024.

[14] Y. Shao, Y. Liu, C. Li, and W. Zhang, "Iterative retrieval generation for knowledge-intensive tasks," *arXiv preprint arXiv:2305.15294*, 2023.

[15] X. Wang, W. Chen, and Y. Zhang, "Memory-based vision-and-language navigation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2023.
[16] X. Jiang, Y. Wang, and Z. Qi, "Long-horizon navigation with structured memory," in *IEEE International Conference on Computer Vision (ICCV)*, 2023.
[17] D. Xie, F. Wang, L. Zhang, and Y. Liu, "Embodied-rag: Retrieval-augmented generation for embodied robots," *arXiv preprint arXiv:2403.12345*, 2024.
[18] M. Booker, L. Chen, and H. Wang, "Semantic memory for embodied agents in dynamic environments," *arXiv preprint arXiv:2402.09876*, 2024.
[19] Y. Liu, W. Zhang, and C. Li, "Task interruption recovery in embodied agents," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2022.
[20] P. Shah, D. Patel, and V. Kumar, "Mobile manipulation with wheeled-legged robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2023.
[21] F. Xia, A. Zamir, Z.-Y. He, A. Sax, J. Malik, and S. Savarese, "Gibson env: Real-world perception for embodied agents," 2018. [Online]. Available: https://arxiv.org/abs/1808.10654
[22] S. K. Ramakrishnan, A. Gokaslan, E. Wijmans, O. Maksymets, A. Clegg, J. Turner, E. Undersander, W. Galuba, A. Westbury, A. X. Chang, M. Savva, Y. Zhao, and D. Batra, "Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai," 2021. [Online]. Available: https://arxiv.org/abs/2109.08238
[23] N. Yokoyama, S. Ha, D. Batra, J. Wang, and B. Bucher, "Vlfm: Vision-language frontier maps for zero-shot semantic navigation," 2023. [Online]. Available: https://arxiv.org/abs/2312.03275
[24] L. Sun, A. Kanezaki, G. Caron, and Y. Yoshiyasu, "Leveraging large language model-based room-object relationships knowledge for enhancing multimodal-input object goal navigation," *arXiv preprint arXiv:2403.14163*, 2024. [Online]. Available: https://arxiv.org/abs/2403.14163
[25] Z. Li, Y. Xia, J. Fan, Y. Lin *et al.*, "Relational graph adaptive network for object-goal navigation," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 1624–1630.
[26] L. Zhang, X. Hao, Y. Tang, H. Fu *et al.*, "NavA^3: Understanding any instruction, navigating anywhere, finding anything," *arXiv preprint arXiv:2508.04598*, 2025. [Online]. Available: https://arxiv.org/abs/2508.04598
[27] S. Chen, T. Chabal, I. Laptev, and C. Schmid, "Object goal navigation with recursive implicit maps," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 7089–7096.
[28] D. S. Chaplot, D. P. Gandhi, A. Gupta, and R. R. Salakhutdinov, "Object goal navigation using goal-oriented semantic exploration," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 4247–4258.
[29] S. K. Ramakrishnan, D. S. Chaplot, Z. Al-Halah, J. Malik, and K. Grauman, "Poni: Potential functions for objectgoal navigation with interaction-free learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 890–18 900.
[30] B. Yu, H. Kasaei, and M. Cao, "L3mvn: Leveraging large language models for visual target navigation," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, Oct. 2023, p. 3554–3560. [Online]. Available: http://dx.doi.org/10.1109/IROS55552.2023.10342512