HeRo-Nav: Heterogeneous Multi-Robot Collaboration for Semantic Navigation With Vision Language Models

Siddhant Tandon[1,*], Wenjun Cheng[1,*], Junzhe Wu[1], Yue Hu[1,†] and Yulun Tian[1,†]
[1]University of Michigan, Ann Arbor *Equal contribution †Equal advising
{siddtan, wjchengx, junzhewu, huyu, yulunt}@umich.edu

***Abstract*—Heterogeneous multi-robot teams can combine complementary locomotion, sensing, and payload capabilities, but coordinating them requires understanding both what the task entails and what role or functionality each robot affords. We present HeRo-Nav, a zero-shot coordination architecture that feeds egocentric frontier images, a shared occupancy map, and natural-language affordance profiles to a Vision-Language Model (VLM), which assigns robots to frontiers based on both scene semantics and each robot's terrain compatibility. We evaluate HeRo-Nav on a hospital disaster scenario in NVIDIA Isaac Sim with three distinct embodiments and platform-dependent terrain hazards. HeRo-Nav achieves the highest success rate (93.3%) and the lowest team hazard exposure (25.7%), providing direct evidence that the VLM assigns robots according to their affordances rather than geometry alone. We analyze conditions under which this affordance-aware reasoning succeeds, adds unnecessary deliberation cost, or fails to outperform geometric baselines, contributing failure-mode insights for deploying foundation models in multi-robot navigation.**

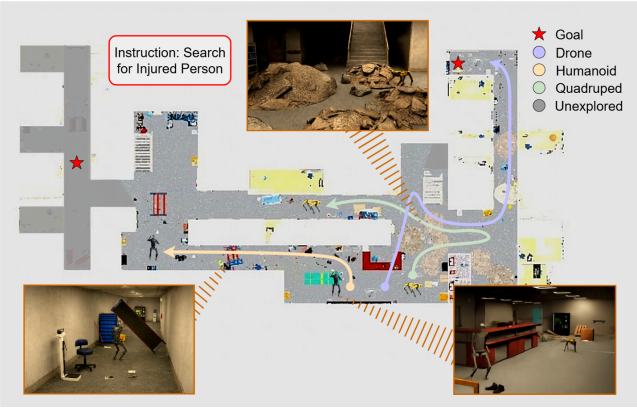

Fig. 1. HeRo-Nav coordinates a heterogeneous team (quadruped, humanoid, drone) to locate casualties in a disaster-struck hospital. The VLM coordinator assigns each robot to frontiers suited to its affordances.

I. Introduction

Multi-robot systems are increasingly deployed in complex, large-scale missions such as subterranean exploration and disaster response [6, 8]. These environments expose the inherent trade-offs of any single platform: a drone's aerial mobility affords it rapid survey of collapsed, debris-filled passages, yet lacks the payload to deliver rescue equipment; a quadruped's sturdy build affords it heavy transport, but it moves slowly in rubble and on stairs. Heterogeneous teams combining diverse locomotion, sensing, and payload capabilities are a natural answer, but fielding such diverse hardware increases coordination complexity. Vision-Language Models (VLMs) [16, 23, 18], which reason jointly over visual observations and natural-language descriptions, offer a promising solution, mapping open-vocabulary scene and task semantics onto the affordances of each robot across diverse environments.

Existing frameworks address pieces of this problem in isolation. Heterogeneous spatial coordinators [4, 24, 26, 20, 14, 7] encode platform kinematics and terrain constraints, but role assignments are fixed and lack semantic scene understanding. VLM-driven multi-robot navigators [21, 17, 10] bring semantic scene reasoning to coordination but assume a homogeneous fleet; the sole exception, COMRES-VLM [19], deploys a heterogeneous air-ground team but feeds VLM without per-agent capability profiles. LLM-based task allocators [22, 25, 1, 13, 12, 5] match robot capabilities to tasks through language reasoning but lack spatial context for navigation.

To bridge this gap, we present **HeRo-Nav** (**He**terogeneous Multi-**Ro**bot Collaboration for Semantic **Nav**igation With Vision Language Models), which fuses a shared bird's-eye-view (BEV) occupancy map, egocentric RGB frontier observations, per-robot affordance profiles, and robot-frontier distances into a VLM-based frontier-assignment loop, enabling zero-shot heterogeneous coordination. We evaluate HeRo-Nav on a hospital disaster scenario built in NVIDIA Isaac Sim [15] (Fig. 1), where three hazard categories impose platform-dependent traversal costs and semantic cues scattered through the scene reward visual reasoning over geometric planning.

In summary, our contributions are:

- We present HeRo-Nav, a coordination framework that integrates semantic scene and affordance reasoning into a VLM frontier-assignment loop for zero-shot heterogeneous multi-robot navigation.
- We develop a multi-embodiment evaluation scenario in Isaac Sim with platform-dependent terrain to test affordance-aware coordination.
- We empirically analyze when affordance-aware VLM coordination succeeds (93.3% SR, 25.7% hazard exposure) and when it fails, contributing failure-mode insights for deploying VLMs in multi-robot navigation.

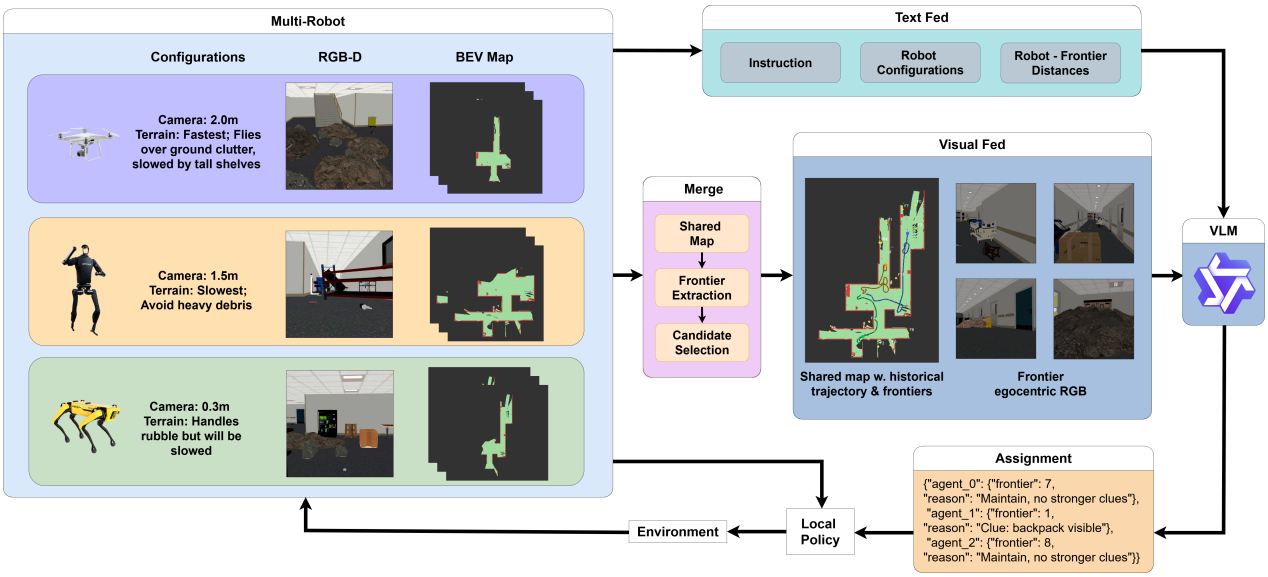

Fig. 2. HeRo-Nav system architecture. Each robot processes RGB-D into per-class BEV occupancy maps (one per robot class). Per-class maps from all robots are fused, then merged under a most-permissive policy for frontier extraction. A candidate set of frontiers is selected and passed, along with egocentric RGB observations, the annotated BEV map, mission goal, per-robot affordance profiles, and robot-candidate distances, to a VLM coordinator that outputs robot-to-frontier assignments. Each robot executes its assignment via A* on its own per-class map.

II. Methods

A. Problem Setup

A team of N heterogeneous robots spanning C classes (e.g., quadruped, drone, humanoid) must locate a natural-language goal g in an unknown environment, maximizing mission success while minimizing mission time. Each class has a textual affordance profile describing its terrain capabilities. The robots build a shared occupancy map and extract frontiers. Then, the system applies geometric filtering to shortlist frontier candidates, and a VLM makes the final assignment by reasoning over scene semantics and affordance compatibility.

B. System Architecture

HeRo-Nav operates as a cyclic coordination framework (Fig. 2) with five stages that carry raw observations to robot-to-frontier assignments and back. The first three stages build a capability-aware shared map and frontier set. The VLM coordinator then makes the assignment conditioned on per-frontier egocentric views and per-robot affordance profiles, the inputs prior multi-robot VLM coordinators omit.

Per-robot mapping. Each robot converts its onboard RGB-D stream into C separate 2D bird's-eye-view (BEV) occupancy maps, one per robot class, since what counts as an obstacle depends on platform geometry. It shares the resulting per-class occupancy updates and RGB keyframes with a centralized map service; raw depth is processed locally and not transmitted.

Map merging and frontier extraction. From the per-class updates of all robots, the map service builds the frontier set in two steps. It first accumulates each class's evidence via log-odds into one per-class map (e.g., all quadruped-class observations into one quadruped map), then merges the C maps under a most-permissive policy. A cell is free if any class can traverse it, so frontier extraction is not bounded by the most constrained platform. Frontiers are connected components of free cells adjacent to unexplored space, filtered by minimum area and obstacle clearance. Each frontier receives a persistent ID, a world location, and one cached egocentric RGB observation o_k, captured once from the robot whose heading best aligns with the frontier (most-centred bearing, within horizontal FOV and a near/far depth range) and not refreshed on subsequent steps.

Frontier candidate selection. From the full frontier set, a global candidate set of up to M frontiers is shortlisted to bound prompt length. A Hungarian assignment [11] over agent–frontier Euclidean distance and frontier area reserves one geometrically favourable frontier per agent (one-to-one, so no two agents share); each agent's current in-flight target is also retained to prevent assignment oscillation, and remaining slots are filled by largest frontier area. This step only selects which frontiers the coordinator sees; the assignment itself is made by the VLM.

VLM coordinator. The candidate set drives the assignment, where affordance reasoning takes effect. A pretrained VLM (Qwen3.6-35B-A3B [16]) receives a multimodal prompt containing: (i) the mission goal g; (ii) for each candidate frontier, its RGB observation o_k and distance from each robot; (iii) for each robot, its current position and a short natural-language affordance profile of its locomotion and terrain traversability (e.g., the quadruped's reads "Quadruped. Small clutter barely slows it; rubble / concrete piles / heavy debris slow it a LOT."); (iv) the permissive BEV map with historical agent trajectories and candidate frontiers. The prompt instructs the model to interpret scene semantics at each frontier, match them against the affordance profiles, and coordinate globally

to avoid redundant coverage. It emits a structured JSON assignment, parsed into $\mathcal{A}_t$.

Local execution. Given $\mathcal{A}_t$, each assigned robot plans an A* path on its own per-class occupancy map and tracks it with a local controller; if a robot stalls, a virtual obstacle placed ahead forces A* to replan around the blockage on the next cycle. Execution yields new observations that feed the next mapping cycle, and the coordinator re-issues assignments periodically, closing the loop.

III. Experiments

A. Simulation Platform and Benchmark

Simulator and team. Built on NVIDIA Isaac Sim for high visual and physical fidelity, our environment supports a configurable heterogeneous team of quadrupeds, humanoids, and drones. This selection provides diverse camera heights, base speeds, and locomotion modes (Table I). To minimize the sim-to-real gap, these parameters are matched to the published specifications of representative physical platforms: the Unitree Go2, Unitree H1, and a mid-range quadrotor. Per-platform hazard multipliers proxy locomotion-dependent cost, keeping coordination separate from low-level control (physical locomotion is future work, Sec. IV). Communication uses the shared map service of Sec. II-B, and each robot runs its own local policy, so sensing, communication, and execution are per-agent.

Hazard scenario. To evaluate multi-robot teaming, we design a hospital disaster relief environment.[1]. The hospital environment contains three hazard categories that impose platform-dependent speed multipliers (Table I): hard-ground obstacles (rubble, concrete), easy-ground clutter (books, bags), and high-occlude structures (toppled shelving). The speed multipliers are hand-specified proxies reflecting known platform traits: quadrupeds handle uneven terrain better than humanoids, drones are unaffected by ground clutter but must detour around tall shelving. Semantic cues such as personal belongings are placed throughout the environment to suggest likely victim locations, providing visual breadcrumbs that a VLM coordinator can leverage but geometric methods cannot.

Benchmark generation. We evaluate across 15 episodes from three spawn locations; each varies the target location, semantic cue placement, and hazard distribution. An episode is scored as successful when any agent comes within 5 m of the ground-truth goal location (the casualty) while observing the target.

B. Baselines

All methods share the same perception, mapping, frontier-extraction and execution stack; they differ only in the frontier-assignment policy, isolating the effect of coordination. The assignment frequency is fixed at 0.125 Hz for all methods, matching the slowest (VLM-based) approaches.

1) **Greedy Cost-Utility (GCU) [2]:** each robot is assigned to the frontier that best trades off travel distance against information gain.

[1]Built upon the NVIDIA Isaac Sim Hospital environment.

TABLE I
PLATFORM CHARACTERISTICS AND SPEED MULTIPLIERS BY HAZARD CATEGORY (1.0 = UNIMPEDED).

	Quadruped	Drone	Humanoid
Camera height (m)	0.3	2.0	1.5
Base speed (m/s)	2.0	3.0	1.0
Hard-ground (rubble)	0.30	1.00	0.10
Easy-ground (clutter)	0.90	1.00	0.80
High-occlude (shelves)	0.80	0.20	0.60

2) **Hungarian [11]:** the globally optimal one-to-one assignment minimizing total travel cost.
3) **Co-NavGPT*:** our adaptation of Co-NavGPT [21] on our heterogeneous stack, preserving its top-view map with robot positions and frontiers without per-frontier RGB observations or per-robot affordance profiles.
4) **HeRo-Nav (Ours):** the full affordance-aware VLM coordinator (Sec. II).

C. Evaluation Metrics

1) **Success Rate (SR) ↑.** Fraction of episodes reaching the goal within 1000s.
2) **Travel Distance.** Total team travel distance, averaged over all episodes including failures and timeouts.
3) **Mission Time ↓.** Mean episode time (failures charged 1000s).
4) **Hazard Exposure ↓.** Fraction of traversal on speed-degraded terrain, per robot and team-pooled.

D. Results

Table II compares HeRo-Nav with three representative baselines over all 15 episodes, split by casualty count, with per-robot hazard exposure.

HeRo-Nav routes robots by affordance, achieving the best success rate and lowest hazard exposure. It attains 93.3% SR (vs. 86.7% for Greedy, 66.7% for Co-NavGPT) and the lowest team hazard exposure (25.7% vs. 40.1% for Greedy). The margin is largest on single-casualty episodes, where sustained search is required: HeRo-Nav solves 88.9% against 55.6–77.8% for geometric planners; on easier two-casualty episodes every method succeeds.

The gain comes from coordination design, not the VLM backbone. Co-NavGPT uses the same VLM but reasons over a single BEV map without per-frontier RGB or affordance profiles. It performs worst overall (66.7% SR) and spends more time on degraded terrain than HeRo-Nav on every robot type (31.3% vs. 25.7% team-pooled).

HeRo-Nav trades efficiency for better terrain decisions. Greedy completes episodes faster (398s vs. 473s) by taking short routes through hazardous terrain, recording the highest hazard exposure. HeRo-Nav routes around hazards at the cost of extra distance, a tradeoff the current benchmark does not fully reward.

TABLE II
RESULTS OVER ALL 15 EPISODES AND SPLIT BY CASUALTY COUNT (LEFT), WITH PER-ROBOT HAZARD EXPOSURE (RIGHT).

	Overall (15)			1-Casualty (9)			2-Casualty (6)			Hazard exposure (%)			
Method	SR ↑	Dist	Time ↓	SR ↑	Dist	Time ↓	SR ↑	Dist	Time ↓	Quad ↓	Drone ↓	Human ↓	Team ↓
Greedy Cost Utility	86.7	**138.6**	**398.2**	77.8	**167.6**	**521.8**	100.0	**95.0**	**212.9**	52.4	17.2	50.6	40.1
Hungarian	73.3	177.9	506.3	55.6	218.1	678.7	100.0	117.4	247.8	36.2	7.0	51.5	31.6
Co-NavGPT	66.7	207.1	614.4	44.4	255.3	821.5	100.0	134.7	303.8	31.3	10.9	51.6	31.3
HeRo-Nav (Ours)	**93.3**	156.6	472.5	**88.9**	182.1	560.1	100.0	118.3	341.1	**30.7**	**5.7**	**40.6**	**25.7**

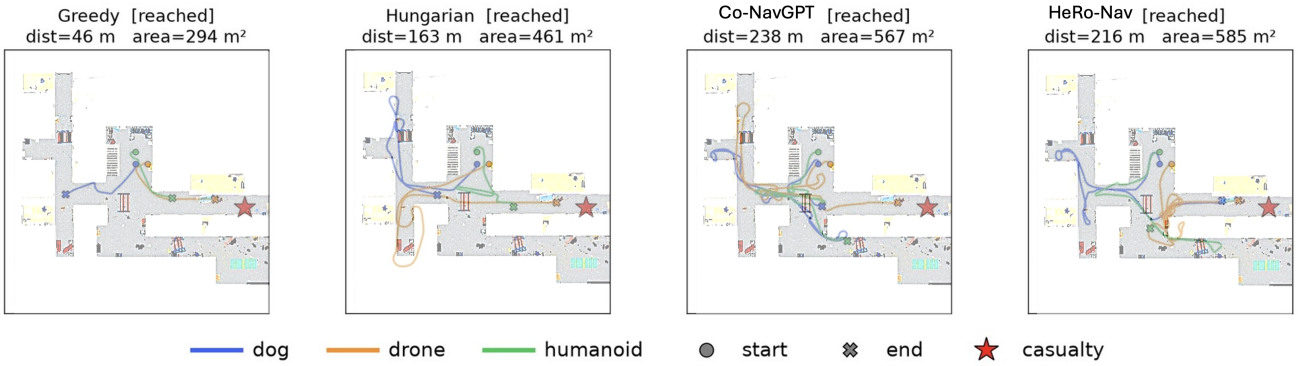

Fig. 3. Agent trajectories on a hard episode; every method reaches the casualty (red star). Greedy takes short, direct routes, while the VLM methods frequently backtrack.

IV. DISCUSSION

We identify two failure modes of the VLM coordinator observed in our experiments, and discuss limitations of the current evaluation that motivate future work.

A. Lack of Explicit Search and Coordination Memory

The coordinator's only record of prior search is an a weak form of memory: an annotated BEV crop of past trajectories and frontier locations. It cannot reliably distinguish covered from uncovered regions, leading to backtracking (Fig. 3). Misleading semantic cues compound this: a clue already investigated keeps attracting agents to the searched area because nothing marks the clue as investigated. The two failures share a cause. The coordinator weighs each frontier and each cue reactively against the current map, with no persistent belief over where the team has been or which cues have already been investigated. A representation that binds the geometric history of the search to the semantic evidence gathered along it, in a form the model can reason over, would let the coordinator discount both explored regions and exhausted cues, producing more directed assignments.

B. Lack of Structured Spatial Representation

The coordinator operates zero-shot over a bird's-eye-view map, without a detailed spatial representation it can reason over. The geometry the allocator needs therefore has to be written into the prompt: we encode the distance between every agent and every frontier as an explicit pairwise comparison in text. This scales poorly, since the prompt and the inference latency grow with the number of agent-frontier pairs. A structured geometric representation would reduce this bottleneck. 3D scene graphs [9] compress an environment into a compact relational form, and multi-robot variants already extend to heterogeneous teams [3]; adapting such a representation to carry both the semantic and the geometric structure the coordinator reasons over, in place of a hand-written list of distances, is a promising next iteration of the system. There is also scope to extend the information these representations classically encode, to enable better planning decisions by the VLM for the varying abilities of members of the team.

C. Evaluation Limitations

Neither the affordance nor the semantic capability of the method is fully exercised by the current task. Hazard penalties are mild enough that short routes through degraded terrain remain fastest, and with two casualties every method succeeds, so success rate does not separate semantic reasoning from coverage. A benchmark that exercises both capabilities would pose tasks a homogeneous team or a geometric planner cannot solve, where success depends on matching robots to what only their embodiment can do. Additionally, our platform-dependent speed multipliers are hand-specified proxies; incorporating physics-based locomotion to empirically calibrate these values is an important direction for future work.

REFERENCES

[1] Michael Ahn, Debidatta Dwibedi, Chelsea Finn, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Karol Hausman, Brian Ichter, Alex Irpan, Nikhil Joshi,

Ryan Julian, Sean Kirmani, Isabel Leal, Edward Lee, Sergey Levine, Yao Lu, Isabel Leal, Sharath Maddineni, Kanishka Rao, Dorsa Sadigh, Pannag Sanketi, Pierre Sermanet, Quan Vuong, Stefan Welker, Fei Xia, Ted Xiao, Peng Xu, Steve Xu, and Zhuo Xu. AutoRT: Embodied Foundation Models for Large Scale Orchestration of Robotic Agents, July 2024.

[2] W. Burgard, M. Moors, C. Stachniss, and F.E. Schneider. Coordinated multi-robot exploration. *IEEE Transactions on Robotics*, 21(3):376–386, 2005. doi: 10.1109/TRO.2004.839232.

[3] Yun Chang, Nathan Hughes, Aaron Ray, and Luca Carlone. Hydra-Multi: Collaborative online construction of 3D scene graphs with multi-robot teams. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023.

[4] Junfeng Chen, Yuxiao Zhu, Xintong Zhang, Bin Luo, and Meng Guo. SLEI3D: Simultaneous Exploration and Inspection via Heterogeneous Fleets Under Limited Communication. *IEEE Transactions on Automation Science and Engineering*, 23:2339–2360, 2026. ISSN 1545-5955, 1558-3783. doi: 10.1109/TASE.2025.3643166.

[5] Junting Chen, Checheng Yu, Xunzhe Zhou, Tianqi Xu, Yao Mu, Mengkang Hu, Wenqi Shao, Yikai Wang, Guohao Li, and Lin Shao. EMOS: Embodiment-aware Heterogeneous Multi-robot Operating System with LLM Agents, February 2025.

[6] Timothy H. Chung, Viktor L. Orekhov, and Angela Maio. Into the robotic depths: Analysis and insights from the darpa subterranean challenge. *Annu. Rev. Control. Robotics Auton. Syst.*, 6:477–502, 2022. URL https://api.semanticscholar.org/CorpusID:256789326.

[7] Muhammad Fadhil Ginting, Kyohei Otsu, Mykel J. Kochenderfer, and Ali-akbar Agha-mohammadi. Capability-aware Task Allocation and Team Formation Analysis for Cooperative Exploration of Complex Environments. *arXiv e-prints*, page arXiv:2411.00400, November 2024. doi: 10.48550/arXiv.2411.00400. _eprint: 2411.00400.

[8] Nicolas Hudson, Fletcher Talbot, Mark Cox, Jason Williams, Thomas Hines, Alex Pitt, Brett Wood, Dennis Frousheger, Katrina Lo Surdo, Thomas Molnar, Ryan Steindl, Matt Wildie, Inkyu Sa, Navinda Kottege, Kazys Stepanas, Emili Hernández, Gavin Catt, William Docherty, and Brendan Tidd. Heterogeneous ground and air platforms, homogeneous sensing: Team csiro data61's approach to the darpa subterranean challenge. *Field Robotics*, 2:595–636, 03 2022. doi: 10.55417/fr.2022021.

[9] Nathan Hughes, Yun Chang, and Luca Carlone. Hydra: A real-time spatial perception system for 3D scene graph construction and optimization. In *Robotics: Science and Systems (RSS)*, 2022.

[10] MoniJesu James, Amir Atef Habel, Aleksey Fedoseev, and Dzmitry Tsetserokou. GoalVLM: VLM-driven Object Goal Navigation for Multi-Agent System, March 2026.

[11] H. W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955. doi: https://doi.org/10.1002/nav.3800020109. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/nav.3800020109.

[12] Kehui Liu, Zixin Tang, Dong Wang, Zhigang Wang, Xuelong Li, and Bin Zhao. COHERENT: Collaboration of Heterogeneous Multi-Robot System with Large Language Models, March 2025.

[13] Zhao Mandi, Shreeya Jain, and Shuran Song. RoCo: Dialectic Multi-Robot Collaboration with Large Language Models, July 2023.

[14] David Morilla-Cabello and Eduardo Montijano. CHORAL: Traversal-Aware Planning for Safe and Efficient Heterogeneous Multi-Robot Routing, January 2026. URL http://arxiv.org/abs/2601.10340. arXiv:2601.10340 [cs].

[15] NVIDIA. Isaac Sim. URL https://github.com/isaac-sim/IsaacSim.

[16] Qwen Team. Qwen3.6-35B-A3B: Agentic coding power, now open to all, April 2026. URL https://qwen.ai/blog?id=qwen3.6-35b-a3b.

[17] Zhixuan Shen, Haonan Luo, Kexun Chen, Fengmao Lv, and Tianrui Li. Enhancing Multi-Robot Semantic Navigation Through Multimodal Chain-of-Thought Score Collaboration, August 2025. URL http://arxiv.org/abs/2412.18292. arXiv:2412.18292 [cs].

[18] Gemini Team et al. Gemini: A family of highly capable multimodal models, 2025. URL https://arxiv.org/abs/2312.11805.

[19] Ruiyang Wang, Hao-Lun Hsu, David Hunt, Jiwoo Kim, Shaocheng Luo, and Miroslav Pajic. COMRES-VLM: Coordinated Multi-Robot Exploration and Search using Vision Language Models, March 2026.

[20] Longrui Yang, Yiyu Wang, Jingfan Tang, Yunpeng Lv, Shizhe Zhao, Chao Cao, and Zhongqiang Ren. HEHA: Hierarchical Planning for Heterogeneous Multi-Robot Exploration of Unknown Environments, October 2025. URL http://arxiv.org/abs/2510.04161. arXiv:2510.04161 [cs].

[21] Bangguo Yu, Qihao Yuan, Kailai Li, Hamidreza Kasaei, and Ming Cao. Co-NavGPT: Multi-Robot Cooperative Visual Semantic Navigation Using Vision Language Models, May 2025. URL http://arxiv.org/abs/2310.07937. arXiv:2310.07937 [cs].

[22] Hongxin Zhang, Weihua Du, Jiaming Shan, Qinhong Zhou, Yilun Du, Joshua B. Tenenbaum, Tianmin Shu, and Chuang Gan. Building Cooperative Embodied Agents Modularly with Large Language Models, February 2024.

[23] Jingyi Zhang, Jiaxing Huang, Sheng Jin, and Shijian Lu. Vision-language models for vision tasks: A survey, 2024. URL https://arxiv.org/abs/2304.00685.

[24] Mingjie Zhang, Chen Feng, Zengzhi Li, Guiyong Zheng, Yiming Luo, Zhu Wang, Jinni Zhou, Shaojie Shen, and Boyu Zhou. SOAR: Simultaneous Exploration

and Photographing with Heterogeneous UAVs for Fast Autonomous Reconstruction, September 2024. URL http://arxiv.org/abs/2409.02738. arXiv:2409.02738 [cs].

[25] Zhonghan Zhao, Kewei Chen, Dongxu Guo, Wenhao Chai, Tian Ye, Yanting Zhang, and Gaoang Wang. Hierarchical Auto-Organizing System for Open-Ended Multi-Agent Navigation, March 2024.

[26] Lanxiang Zheng, Mingxin Wei, Ruidong Mei, Kai Xu, Junlong Huang, and Hui Cheng. AAGE: Air-Assisted Ground Robotic Autonomous Exploration in Large-Scale Unknown Environments. *IEEE Transactions on Robotics*, 41:1918–1937, 2025. ISSN 1941-0468. doi: 10.1109/TRO.2025.3543275.