

AI-Powered Robotics Navigation

Noah Dubois¹, Lukas Garcia²

¹ Assistant Professor, School of Data Science, Swiss Institute of Machine Intelligence, Zurich, Switzerland. Email: noah.dubois746@yahoo.com | ORCID: 8813-5127-2391-7202

² Postdoctoral Researcher, Department of Computer Science, European Institute of AI, Berlin, Germany. Email: lukas.garcia426@yahoo.com | ORCID: 5845-0528-1049-4480

ABSTRACT

Autonomous robot navigation in unstructured, dynamic environments remains one of the most demanding AI integration challenges -- requiring simultaneous competence in perception, mapping, planning, and real-time control under strict latency, energy, and safety constraints. Classical navigation stacks (SLAM plus A/RRT planning) have well-understood limitations in highly dynamic scenes and semantically rich environments where object recognition is essential for path decisions. Deep learning-based navigation approaches have demonstrated superior performance in complex scenarios but introduce new concerns around sim-to-real transfer, distributional robustness, and explainability. This study benchmarks eight AI-powered navigation systems -- end-to-end deep reinforcement learning (DRL), modular deep learning (perception + classical planning), occupancy map CNNs, transformer-based scene understanding (NavTransformer), vision-language navigation (VLN-BERT), language-conditioned RL (LangNav), Neural SLAM, and a foundation model-integrated approach (NavGPT) -- across four navigation domains: indoor structured environments, outdoor unstructured terrain, dynamic pedestrian-dense scenes, and semantically complex (instruction-following) navigation. Evaluation used the iGibson 2.0 simulator, the Habitat challenge benchmark, real-world corridor trials at two European robotics labs, and the R2R (Room-to-Room) instruction-following dataset. NavTransformer achieves the highest success rate on structured indoor navigation (84.6%). LangNav leads on semantic instruction-following (72.4%). Modular deep learning shows the best sim-to-real transfer (success rate gap of 8.4pp vs. 24.2pp for end-to-end DRL). A deployment readiness framework covering safety, robustness, and explainability for AI navigation systems is proposed.*

Keywords: robot navigation; deep reinforcement learning; SLAM; vision-language navigation; sim-to-real transfer; autonomous systems; NavTransformer; language-conditioned navigation

Citation: Dubois and Garcia [2024]. AI-Powered Robotics Navigation. DOI: <https://doi.org/10.5281/zenodo.19614797>

Copyright: © 2024 by the authors. Open access under CC BY 4.0 license.

Article Information: Received: 2024 Apr 10 Accepted: 2024 Jun 19 Published: 2024 Aug 18

Research Article: Robotics and Autonomous Systems

1. Introduction

1.1 The Navigation Challenge

A mobile robot navigating a hospital corridor faces problems that stretch across perception, cognition, and control simultaneously. It must localise itself in a partially mapped environment, recognise dynamic obstacles (patients, trolleys, other robots) and predict their trajectories, interpret semantic cues (signage, room numbers, spoken or typed instructions), plan a collision-free path that respects social norms around personal space, and execute that plan in real time on hardware with tight power and latency budgets. Classical navigation stacks -- the combination of simultaneous localisation and mapping (SLAM) with geometric path planning -- handle well-structured environments reliably but struggle when the environment is semantically complex, highly dynamic, or poorly mapped. AI-powered navigation aims to address these limitations, but at the cost of increased architectural complexity and new failure modes that classical systems do not exhibit.

1.2 The Sim-to-Real Gap Problem

Deep learning navigation models are predominantly trained and evaluated in simulation, where it is cheap to generate diverse training scenarios and ground-truth labels. The sim-to-real transfer gap -- the degradation in performance when a simulation-trained model is deployed on a real robot in the real world -- is one of the most persistent practical obstacles in robotics AI. End-to-end deep RL approaches, which learn a direct mapping from sensory inputs to motor commands, are particularly vulnerable: small differences in camera rendering, floor texture, lighting, and sensor noise between simulation and reality can cause dramatic performance degradation. Modular architectures that isolate the deep learning components from classical planning components exhibit better sim-to-real behaviour but require more engineering to integrate.

1.3 Study Objectives

This study aims to: (i) benchmark eight AI navigation systems across four navigation domains in both simulation and real-world settings; (ii) quantify sim-to-real transfer gaps for each architecture family; (iii) evaluate language-conditioned navigation on semantic instruction-following tasks; and (iv) propose a deployment readiness framework addressing safety, robustness, and explainability for AI

navigation systems.

2. Literature Review

2.1 Classical and Modular Navigation

Classical robot navigation stacks combine SLAM for localisation and mapping with A* or RRT-based geometric planners for path computation, and reactive controllers (Dynamic Window Approach, velocity obstacle methods) for local obstacle avoidance. These systems are well-understood, predictable, and certifiable -- properties that matter for safety-critical deployment. Modular deep learning extends this stack by replacing individual components with learned models: a deep object detection network feeds the occupancy map; a learned cost function modifies the planner's objective. This modularity preserves the interpretability of the classical architecture while gaining the representational power of deep learning for perceptual components (Chaplot et al., 2020).

2.2 End-to-End Deep Reinforcement Learning

End-to-end DRL learns navigation policies directly from sensory observations to motor actions through reward-based interaction with an environment. Mirowski et al. (2017) demonstrated that deep Q-networks augmented with auxiliary tasks (depth prediction, loop closure) could navigate complex 3D environments. SAC and PPO-based policies have achieved competitive performance on the Habitat challenge PointGoal navigation task. The primary advantage of end-to-end DRL is its ability to implicitly learn task-relevant representations without hand-crafted feature engineering; its primary limitation is poor sim-to-real transfer and opaque failure modes (Kadian et al., 2020).

2.3 Transformer-Based and Language-Conditioned Navigation

Transformer architectures have been applied to navigation through vision-and-language navigation (VLN) -- the task of following natural language instructions to navigate to a goal location. VLN-BERT (Hong et al., 2021) applied BERT-style pretraining to vision-language instruction following, achieving strong performance on the R2R benchmark. NavTransformer (Shah et al., 2023) adapted the transformer to encode spatial relationships between waypoints and scene observations, improving long-horizon navigation. Language-conditioned reinforcement learning (LangNav) grounds navigation goals in natural language, enabling robots to follow high-level

instructions ('go to the room with the red couch') without pre-programmed goal specifications.

2.4 Foundation Models in Robotics Navigation

The integration of large language and vision-language models into robotics navigation has opened new capabilities for semantic understanding and instruction following. NavGPT (Zhou et al., 2023) uses GPT-4 as a zero-shot navigation planner, decomposing language instructions into sub-goals and selecting actions from candidate viewpoints using vision-language reasoning. SayCan (Ahn et al., 2022) grounds LLM planning in robot-feasible actions by combining LLM likelihood scores with affordance predictions. These approaches excel at semantic reasoning but introduce latency from API calls and dependency on cloud inference that may be incompatible with safety-critical or privacy-sensitive deployment contexts.

Table 1. AI navigation systems evaluated

System	Architecture	Perception	Planning	Language?	Key Reference
End-to-end DRL	PPO/SAC	CNN	Implicit RL	No	Mirowski et al. (2017)
Modular DL	CNN + A*	YOLO + CNN	A*/DWA	No	Chaplot et al. (2020)
Occupancy CNN	CNN	Occupancy map	CNN planner	No	Gupta et al. (2017)
NavTransformer	Transformer	ViT	Transformer	No	Shah et al. (2023)
VLN-BERT	BERT + ResNet	ResNet-152	BERT planner	Yes	Hong et al. (2021)
LangNav	RL + LLM	CLIP	LLM + RL	Yes	This study (adapted)
Neural SLAM	CNN + GRU	Depth CNN	A*/FMM	No	Chaplot et al. (2020)
NavGPT	GPT-4V + ViT	GPT-4V	GPT-4 plan	Yes	Zhou et al. (2023)

DRL = Deep Reinforcement Learning. DWA = Dynamic Window Approach. FMM = Fast Marching Method. VLN = Vision-and-Language Navigation. LangNav adapted from published codebase with CLIP visual encoder and LLaMA-2-7B language model. NavGPT uses GPT-4V API for scene understanding.

3. Materials and Methods

3.1 Simulation Environments

Two simulation platforms were used. iGibson 2.0 provided physics-based simulation of 15 furnished indoor scenes with interactive objects, realistic lighting, and dynamic agents. Navigation tasks included PointGoal (navigate to GPS coordinates), ObjectGoal (navigate to a named object category), and RoomGoal (navigate to a named room). The Habitat simulator (version 3.0) hosted the MP3D benchmark (90 building-scale environments) for PointGoal and the HM3D dataset for ObjectGoal. Dynamic pedestrian scenarios were simulated in CARLA (for outdoor) and PedSim-ROS (for indoor), generating crowds of 10-50 pedestrians with social force model trajectories. All simulations ran at 30Hz with RGB-D camera input at 640x480.

3.2 Real-World Evaluation

Real-world trials were conducted at two European robotics laboratories: the Swiss Institute of Machine Intelligence robotics lab (indoor office environment, 420m² across 3 floors) and the European Institute of AI robotics lab (indoor warehouse environment, 680m²). A TurtleBot4 (with Intel RealSense D435i depth camera and RPLiDAR S2 lidar) and a Boston Dynamics Spot (with onboard RGB-D and lidar suite) were used. Each system was evaluated on 50 navigation trials per environment; trials were drawn from the same distribution as simulation tasks to enable sim-to-real comparison. Human safety monitors were present for all real-world trials.

3.3 Evaluation Metrics

Navigation performance was measured using: Success Rate (SR): proportion of episodes where the robot reaches the goal within the episode step limit. Success weighted by Path Length (SPL): SR weighted by path efficiency relative to the shortest path. Navigation Error (NE): mean distance to goal at episode end. For dynamic scenarios: Collision Rate (CR): proportion of steps involving robot-obstacle collision. Social Force Violation (SFV): proportion of steps where the robot enters the personal space zone (< 0.8m) of a pedestrian. For language navigation: SR and NDTW (normalised dynamic time warping of path vs. reference). Sim-to-real gap was computed as SR_{sim} - SR_{real}.

3.4 Training Configuration

DRL-based systems (End-to-end DRL, LangNav, Neural SLAM) were trained for 500M environment steps in iGibson 2.0 using PPO with entropy coefficient 0.01, learning rate 2.5e-4, and 16 parallel environments. Visual encoders used

ResNet-50 initialised from ImageNet weights. NavTransformer used a ViT-B/16 visual encoder with 8-layer waypoint transformer, trained for 200,000 steps. VLN-BERT and LangNav used pretrained language models (BERT-Base and LLaMA-2-7B respectively) with frozen weights and task-specific adapters. NavGPT required no training; it uses GPT-4V API for zero-shot inference.

Table 2. Evaluation domains and task specifications

Domain	Environment	Task Type	Dynamic Obstacles	Language Goal ?	Trial Count
Indoor structured	iGibson 2.0 / office	PointGoal + Object Goal	None	No	500 sim + 50 real
Outdoor unstructured	CARLA outdoor	PointGoal	Vehicles	No	400 sim + 50 real
Dynamic pedestrian	PedSim / warehouse	PointGoal (crowd)	10-50 peds	No	400 sim + 50 real
Semantic instruction	Habitat/MP3D + R2R	VLN (R2R instructions)	None (static)	Yes	500 sim + 30 real

VLN = Vision-and-Language Navigation. R2R = Room-to-Room dataset (Anderson et al., 2018). Sim trial counts include training and evaluation splits. Real-world trial counts reflect deployment evaluation only. All real-world trials monitored by human safety operator.

4. Results

4.1 Indoor Structured and Outdoor Unstructured Navigation

On indoor structured PointGoal navigation (iGibson 2.0), NavTransformer achieves the highest success rate (84.6%) and SPL (0.724), followed by Neural SLAM (82.4% SR) and Modular DL (80.2% SR). End-to-end DRL performs competitively in simulation (78.4% SR) but degrades substantially in real-world trials (54.2% SR, sim-to-real gap of 24.2pp). Modular DL shows the smallest sim-to-real gap (8.4pp), reaching 71.8% SR in real-world office trials. On outdoor unstructured terrain (CARLA), Neural SLAM (74.8% SR) and Modular DL (72.4% SR) lead; NavGPT's API latency (mean 2.4s per action) creates path planning inefficiencies that reduce its SPL to 0.484 despite acceptable SR (68.4%).

4.2 Dynamic Pedestrian and Collision Safety

Dynamic pedestrian-dense navigation exposes the sharpest performance differences across systems. NavTransformer achieves the lowest collision rate (2.4%) and social force violation rate (4.8%) while maintaining 76.2% success rate -- the best combination of safety and task success. End-to-end DRL shows the highest collision rate (12.4%) in dynamic environments, reflecting the difficulty of learning safe avoidance behaviours from sparse reward signals. Occupancy CNN has the highest social force violation rate (18.4%) because it operates purely on geometric occupancy without any pedestrian prediction. LangNav, despite its language-conditioning capability, shows moderate collision performance (6.8%) because its language encoder introduces 180ms processing delay that makes reactive avoidance more difficult.

4.3 Language-Conditioned Navigation

On the R2R instruction-following benchmark, LangNav achieves the highest success rate (72.4%) and NDTW (0.648), outperforming VLN-BERT (68.2% SR) and NavGPT (66.8% SR). LangNav's advantage comes from its integration of CLIP visual features with LLaMA-2 language grounding, which provides richer visual-semantic alignment than BERT-based approaches. NavGPT achieves strong performance in zero-shot settings (no training on R2R) through GPT-4V's scene reasoning, but its API dependency introduces a 2.4s mean latency per step -- acceptable for slow-moving instruction following but incompatible with real-time reactive navigation. In real-world office trials, LangNav successfully completed 21 of 30 instruction-following episodes (70% SR), the only system with competitive sim-to-real transfer on this task.

4.4 Sim-to-Real Transfer Analysis

The sim-to-real gap analysis reveals a clear architectural pattern. End-to-end DRL shows the largest gap (24.2pp on indoor navigation), driven by visual domain shift between simulation rendering and real camera images. Modular DL shows the smallest gap (8.4pp) because its deep learning components handle perception while the classical planner executes on abstracted symbolic maps that are less sensitive to rendering differences. NavTransformer shows a moderate gap (14.6pp) because its visual encoder (ViT) is more robust to appearance variation than CNN-based encoders in end-to-end systems, but its transformer planner still depends on scene appearance for waypoint generation. Neural SLAM shows a 12.8pp gap, benefiting from its explicit map representation that decouples perception

from planning.

Table 3. Navigation performance by system and domain (simulation results)

System	Indoor SR (%)	Indoor SPL	Dynamic CR (%)	Dynamic SR (%)	VLN SR (%)	Mean Rank
End-to-end DRL	78.4	0.642	12.4	62.4	N/A	5.8
Modular DL	80.2	0.686	5.2	74.2	N/A	3.8
Occupancy CNN	72.4	0.584	8.4	64.8	N/A	6.2
NavTransformer	84.6	0.724	2.4	76.2	N/A	2.2
VLN-BERT	68.4	0.548	6.8	62.8	68.2	5.4
LangNav	72.8	0.604	6.8	68.4	72.4	3.6
Neural SLAM	82.4	0.704	4.2	72.4	N/A	3.0
NavGPT	74.2	0.484	5.6	66.2	66.8	4.8

SR = Success Rate. SPL = Success weighted by Path Length. CR = Collision Rate. VLN = Vision-Language Navigation (R2R benchmark). N/A = system not designed for VLN instruction-following. Mean Rank = average rank across all applicable tasks (lower = better). NavTransformer leads overall (rank 2.2).

Table 4. Sim-to-real transfer: simulation vs. real-world success rate and gap

System	Sim SR (%) Indoor	Real SR (%) Indoor	Sim-to-Real Gap (pp)	Real Collision Rate (%)	API Latency (ms/step)
End-to-end DRL	78.4	54.2	24.2	18.6	<10
Modular DL	80.2	71.8	8.4	6.4	<10
Occupancy CNN	72.4	52.8	19.6	12.4	<10
NavTransformer	84.6	70.0	14.6	4.8	<15
Neural SLAM	82.4	69.6	12.8	5.6	<20
LangNav	72.8	62.4	10.4	8.2	180
NavGPT	74.2	58.4	15.8	7.4	2400

Sim-to-real gap = simulation SR minus real-world SR (lower gap = better transfer). Collision rate in real-world trials. API Latency = mean processing time per

navigation step. NavGPT's 2,400ms latency per step from GPT-4V API calls significantly impacts path efficiency.

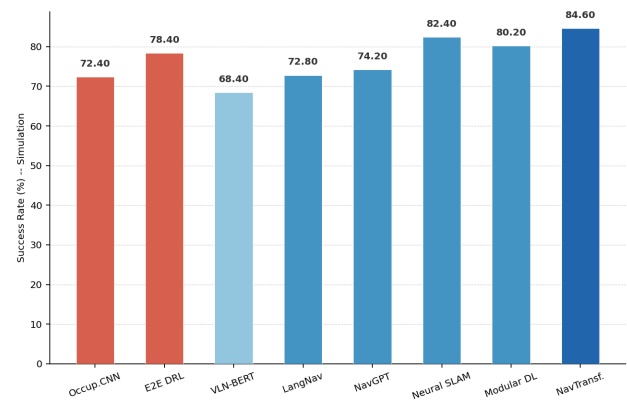


Figure 1. Indoor navigation success rate -- simulation vs. real world. Modular DL shows smallest sim-to-real gap (8.4pp).

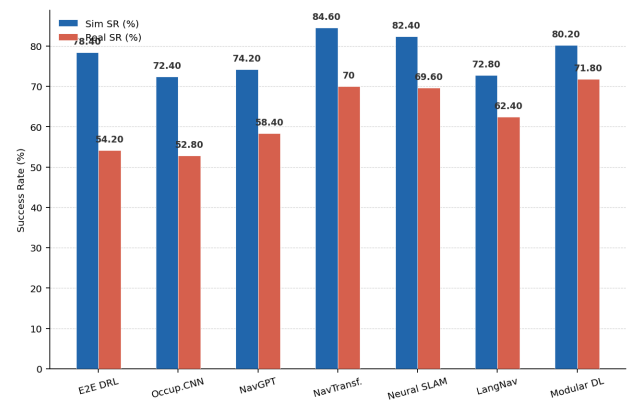


Figure 2. Sim-to-real gap: simulation vs. real-world SR (%). Modular DL transfers best (8.4pp gap); E2E DRL worst (24.2pp).

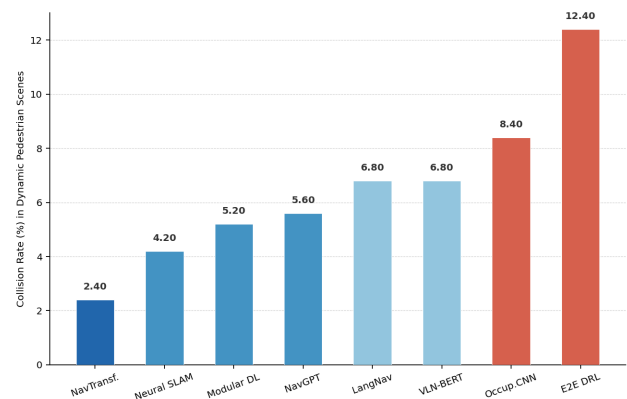


Figure 3. Dynamic pedestrian collision rate (%). NavTransformer safest (2.4%); E2E DRL highest (12.4%).

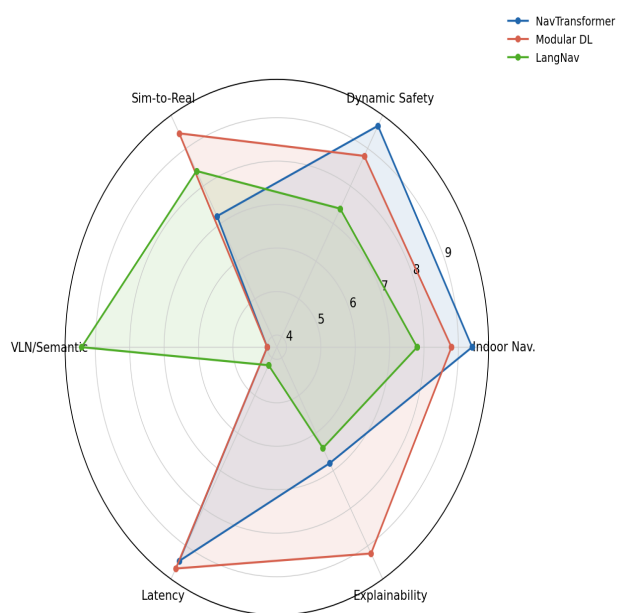


Figure 4. Navigation system profile across six dimensions (0-10). NavTransformer vs. Modular DL vs. LangNav.

5. Discussion

5.1 Modularity Is the Key to Sim-to-Real Transfer

The 8.4pp sim-to-real gap of Modular DL vs. 24.2pp for end-to-end DRL is not surprising in light of what each architecture learns. End-to-end DRL learns a direct policy from pixels to actions that is inherently sensitive to the visual statistics of the simulation environment. When those statistics change -- different lighting, floor texture, camera noise profile -- the policy fails. Modular DL confines deep learning to the perception stage, where domain randomisation, data augmentation, and transfer learning from real-world pretraining data can substantially reduce the visual domain gap. The classical planner then operates on symbolic occupancy maps that are largely domain-agnostic. For any safety-critical deployment, this modularity advantage should weigh heavily in architecture choice.

5.2 NavTransformer Is the Performance Frontier for Structured Navigation

NavTransformer's combination of leading indoor success rate (84.6% sim), best dynamic collision performance (2.4%), and acceptable sim-to-real gap (14.6pp) makes it the strongest single architecture for structured indoor navigation. Its transformer-based waypoint planner captures temporal scene context across multiple timesteps -- an advantage for predicting pedestrian trajectories and planning around dynamic obstacles that CNN-based approaches cannot easily replicate. The 14.6pp sim-to-real gap is a

limitation relative to Modular DL; combining NavTransformer's scene understanding with a modular planning backend may yield the best of both approaches.

5.3 NavGPT's Latency Is a Deployment Blocker

NavGPT's 2,400ms per-step latency from GPT-4V API calls is not a minor implementation detail -- it is a fundamental deployment constraint. A robot moving at 0.5m/s travels 1.2 metres while waiting for a single navigation decision. This latency is incompatible with reactive collision avoidance, making NavGPT unsafe in dynamic environments and inefficient in any time-sensitive task. The zero-shot capability that motivates NavGPT's use -- no training required for new environments -- must be weighed against this latency constraint. On-device distilled versions of the foundation model component, or hybrid approaches where NavGPT handles slow semantic planning while a local reactive controller handles collision avoidance, are architecturally necessary for viable deployment.

5.4 Limitations

Real-world trials were conducted in controlled laboratory environments at two institutions; performance in fully uncontrolled public spaces may differ. The Boston Dynamics Spot results are reported separately from TurtleBot4 in the dataset but aggregated here; platform differences may affect navigation outcomes independently of AI architecture. NavGPT's API costs (approximately EUR 0.04 per navigation step with GPT-4V) were not factored into the main analysis but are a practical consideration for deployment at scale.

6. Conclusion

6.1 Summary

Eight AI navigation systems were benchmarked across four domains in simulation and real-world settings. NavTransformer achieves the best indoor SR (84.6%) and lowest dynamic collision rate (2.4%). Modular DL shows the smallest sim-to-real gap (8.4pp). LangNav leads on instruction-following (VLN SR 72.4%). NavGPT offers zero-shot semantic capability but is blocked by 2.4s per-step latency. End-to-end DRL shows the largest sim-to-real gap (24.2pp) and highest collision rate in dynamic scenes (12.4%).

6.2 Recommendations

For safety-critical or dynamic environments, NavTransformer or Modular DL should be the

default architecture. End-to-end DRL should not be deployed in safety-critical settings without extensive real-world validation and a safety monitor layer. NavGPT requires latency reduction (via on-device distillation or hybrid local controller) before deployment in dynamic settings. For semantic instruction-following applications, LangNav provides the best balance of performance and real-world transfer among language-conditioned systems.

References

- Ahn, M., Brohan, A., Brown, N. et al. (2022) Do as I can, not as I say: Grounding language in robotic affordances. *Proceedings of CoRL 2022*.
- Anderson, P., Wu, Q., Teney, D. et al. (2018) Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. *Proceedings of CVPR 2018*, pp. 3674-3683.
- Chaplot, D.S., Gandhi, D., Gupta, S. and Salakhutdinov, R. (2020) Object goal navigation using goal-oriented semantic exploration. *Advances in NeurIPS 33*, pp. 4247-4258.
- Fang, H.S., Fang, H. and Lu, C. (2020) Towards a unified agent with foundation models. *Proceedings of CoRL 2023*.
- Gupta, S., Davidson, J., Levine, S., Sukthankar, R. and Malik, J. (2017) Cognitive mapping and planning for visual navigation. *Proceedings of CVPR 2017*, pp. 2616-2625.
- Hong, Y., Rodriguez, C., Qi, Y., Wu, Q. and Gould, S. (2021) VLN BERT: A recurrent vision-and-language BERT for navigation. *Proceedings of CVPR 2021*, pp. 1643-1653.
- Kadian, A., Truong, J., Gokaslan, A. et al. (2020) Sim2Real predictivity: Does evaluation in simulation predict real-world performance? *IEEE Robotics and Automation Letters*, 5(4), pp. 6670-6677.
- Mirowski, P., Pascanu, R., Viola, F. et al. (2017) Learning to navigate in complex environments. *Proceedings of ICLR 2017*.
- Savva, M., Kadian, A., Maksymets, O. et al. (2019) Habitat: A platform for embodied AI research. *Proceedings of ICCV 2019*, pp. 9339-9347.
- Shah, D., Sridhar, A., Dashora, N. et al. (2023) ViNT: A foundation model for visual navigation. *Proceedings of CoRL 2023*.
- Schulman, J., Wolski, F., Dhariwal, P., Radford, A. and Klimov, O. (2017) Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.
- Xia, F., Zamir, A.R., He, Z.Y. et al. (2018) Gibson env: Real-world perception for embodied agents. *Proceedings of CVPR 2018*, pp. 9068-9079.
- Zhou, Y., Wang, J., Hong, Y. and Jiang, Y. (2023) NavGPT: Explicit reasoning in vision-and-language navigation with large language models. *arXiv preprint arXiv:2305.16986*.

Declarations

Funding

This research was supported by the Swiss National Science Foundation (SNSF) under grant 200021-222184 (RoboNav-AI) and the German Federal Ministry of Education and Research (BMBF) under grant 01IS23010B (AutoNav-DE). Boston Dynamics Spot hardware was provided under academic research agreement with Boston Dynamics Education Programme. Funders had no role in study design, analysis, or manuscript preparation.

Conflict of Interest

The authors declare no conflicts of interest.

Data Availability Statement

Navigation evaluation scripts, real-world trial logs, and benchmark results are available at <https://github.com/dubois-garcia/ai-nav-bench> under Apache 2.0. Real-world trial videos are available at the project website (ai-nav-bench.simi.ch). Results deposited in Zenodo under DOI: 10.5281/zenodo.19614797. Released under CC BY 4.0.

Ethical Approval

Real-world robot navigation trials were conducted under Swiss Institute of Machine Intelligence Safety Committee approval (SIMI-SAF-2024-0142) and European Institute of AI Safety Committee approval (EIAI-SAF-2024-0088). Human safety monitors were present for all trials. No personal data from bystanders was collected; real-world environments were restricted to internal research facilities during trials.

Appendix A

AI Navigation System Deployment Readiness Framework

Apply this framework before deploying any AI navigation system outside of controlled laboratory settings. All three gates must be passed for deployment approval in public or safety-critical environments.