From Imitation to Interaction: A Two-Stage Training Paradigm for Social Navigation

Wenxiang Shi* Jingmeng Zhou Weijun Zeng Zhipeng Zhang†
AutoLab, School of Artificial Intelligence, Shanghai Jiao Tong University

Abstract

This paper addresses the task of social navigation, where an autonomous agent must navigate to a target coordinate in a 3D indoor environment while avoiding both static and dynamic human obstacles. We propose a robust training paradigm that synergizes Imitation Learning (IL) and Reinforcement Learning (RL) in a virtuous cycle. The core of our methodology lies in this iterative, mutually boosting process. More specifically, a high-quality IL policy, pre-trained via Behavioral Cloning (BC) on expert data, provides a superior initialization for the RL agent. This agent is subsequently fine-tuned using Proximal Policy Optimization (PPO), enhanced with auxiliary tasks for human motion prediction to improve its efficiency and social compliance. Critically, the resulting enhanced RL policy can then be used to collect a larger and higher-quality dataset of demonstrations, which in turn feeds back into training an even more proficient IL model for the next iteration. This creates a self-reinforcing loop where each stage bootstraps the other, progressively amplifying the agent's capabilities. Our full IL+RL approach achieves a final Total Score of ***0.6977*** *in the* ***Robosense2025 Social navigation*** *competition, demonstrating a significant improvement over baseline methods trained with IL-only or RL-from-scratch. Code and models will be released.*

1. Introduction

Navigating in human-populated environments is a fundamental challenge for embodied AI [1–6]. An autonomous agent must not only reach a specified goal but also do so in a socially compliant manner, respecting personal space and avoiding collisions with human agents [7–10]. This task, known as social navigation, requires the agent to interpret complex sensory inputs, understand implicit social norms, and make decisions in real-time [11]. While classic modular approaches often struggle in dynamic scenes, end-to-end learning methods have shown significant promise [12, 13].

Reinforcement Learning (RL) offers a powerful framework for learning navigation policies through environmental interaction. However, training an RL agent from scratch for complex tasks like social navigation is notoriously difficult, often suffering from sample inefficiency and requiring extensive reward engineering [14, 15]. On the other hand, Imitation Learning (IL) can effectively bootstrap the learning process by leveraging expert demonstrations [16, 17]. Yet, IL-trained policies often suffer from poor generalization to states not seen during training and can be limited by the quality and coverage of the demonstration data [18].

In this work, we propose a hybrid approach that combines the strengths of both paradigms. Our method follows a two-stage training pipeline: IL pre-training followed by RL fine-tuning [18, 19]. First, we pre-train a policy on a curated dataset of expert trajectories using BC. This provides the agent with a strong, behaviorally sound foundation. Second, we fine-tune this pre-trained policy using Proximal Policy Optimization (PPO). To further enhance the agent's social awareness, we integrate a Spatial-Temporal Forecasting Module during fine-tuning, which introduces auxiliary tasks for predicting human trajectories and positions, inspired by recent work in the field [8, 20, 21].

We validate our approach through rigorous experiments in the Habitat simulator. Our IL+RL model not only outperforms RL-from-scratch and IL-only baselines but also achieves a top-ranking score in the official competition. The results demonstrate that our paradigm effectively initializes a competent navigation policy and successfully refines it for improved success, efficiency, and social compliance.

2. Methodology

Our approach to Social navigation is based on a two-stage training paradigm that first imitates an expert policy and then refines it through reinforcement learning [18]. This section details our data collection process, the agent's neural network architecture, and our training strategy. Figure 1 provides an overview of our architecture.

*Email: swx-luna@sjtu.edu.cn

†Corresponding author.

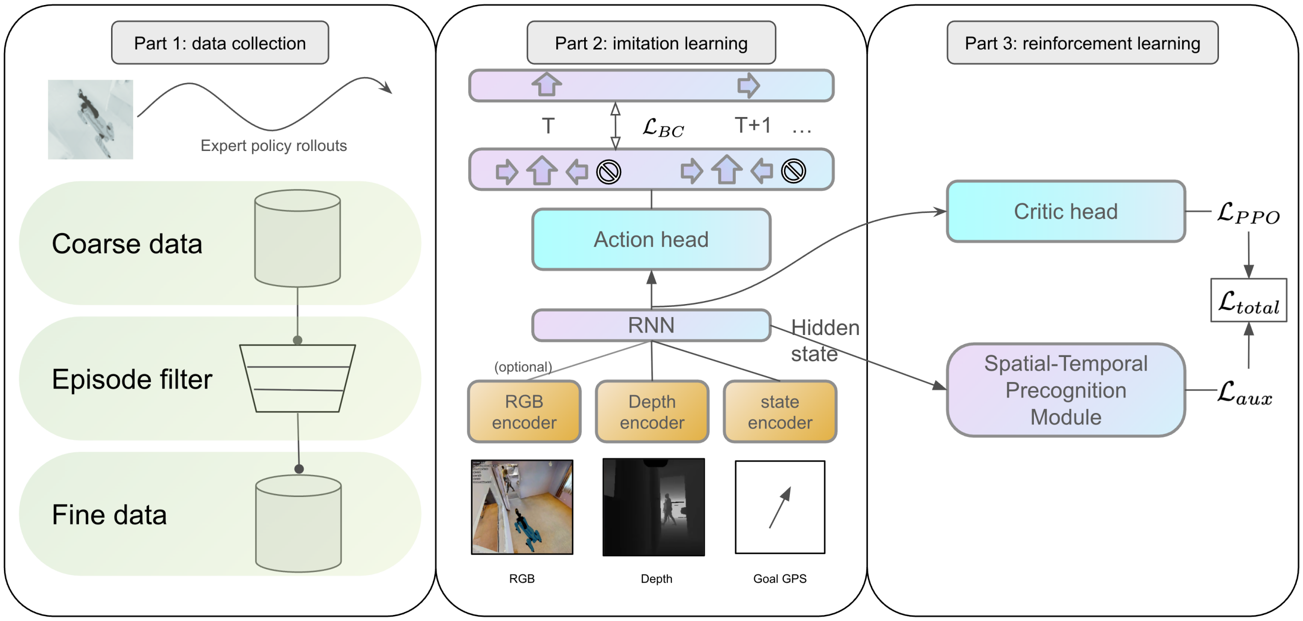

Figure 1. An overview of our proposed two-stage training framework. The process begins with (1) Data Collection, where expert demonstrations are generated and filtered for quality. Next, in (2) Imitation Learning, a policy network is pre-trained via Behavioral Cloning (BC). Finally, in (3) Reinforcement Learning, the pre-trained model is fine-tuned using Proximal Policy Optimization (PPO), incorporating a Spatial-Temporal Precognition Module as an auxiliary task to enhance social awareness.

2.1. Expert Data Collection

To effectively train our agent, we first generated a high-quality dataset of expert demonstrations. We conducted this process within the **Habitat simulator**, utilizing the photorealistic **HM3D dataset** for environments [22] and the **Bullet physics engine** for dynamics.

The expert trajectories were generated by deploying a proficient, pre-trained Reinforcement Learning (RL) policy. This expert agent was tasked with solving Social navigation tasks in the designated training scenes. To ensure the quality of our imitation dataset, we applied a strict filtering protocol to the collected trajectories. Only episodes that successfully reached their goal with a high reward above the threshold were retained, effectively culling suboptimal or failed attempts. This resulted in a dataset of clean, near-optimal expert demonstrations for our agent to learn from.

2.2. Model Architecture

Our agent's policy is represented by a unified neural network, with an architecture designed to process multimodal inputs and maintain a memory of past states. Our implementation is based on the Falcon baseline [8].

Visual Encoder. Egocentric depth observations are processed by a ResNet backbone, pre-trained on ImageNet, to extract rich visual features.

State Encoder. The agent's current pose and the target PointGoal coordinates are jointly encoded by a 2-layer MLP.

Temporal Memory. The visual and state features are concatenated and fed into a 2-layer Gated Recurrent Unit (**GRU**) with a hidden state size of 512. This recurrent component allows the agent to integrate information over time, which is crucial for navigation.

Output Heads. The final output from the GRU is passed to two separate linear layers: a **policy head** that produces a distribution over the discrete action space, and a **value head** that estimates the state-value function, which is used during RL fine-tuning.

This single, end-to-end policy directly maps observations to actions and values without relying on explicit social attention mechanisms. This simpler structure facilitates convergence during training.

2.3. Two-Stage Training Paradigm

Our training pipeline is composed of two sequential stages: (1) Imitation Learning (IL) for pre-training and (2) Reinforcement Learning (RL) for fine-tuning. This strategy leverages expert data to provide a strong initialization, which is then improved upon through environmental interaction.

2.3.1 Stage 1: Imitation Learning Pre-training

In the first stage, we use **Behavioral Cloning (BC)** to train the policy to mimic the expert's behavior. The primary objective is to minimize the cross-entropy loss between the policy's predicted action distribution and the expert's chosen action.

Each training sample consists of a sequence of observations and actions. The model is conditioned not only on the **current observation** (o_t) but also on H **historical observations** ($o_{t-H}, \ldots, o_{t-1}$). This historical context serves to warm up the recurrent state of the GRU, which is more computationally efficient than relying on long-term backpropagation through time.

We employ an auto-regressive approach to predict a sequence of F future actions (e.g., $\hat{a}_t, \hat{a}_{t+1}, \ldots, \hat{a}_{t+F-1}$). For each step k within this prediction horizon, the model receives the ground-truth expert observation o^*_{t+k} as input, but it is conditioned on the action $\hat{a}_{t+k-1}$ sampled from its own predicted distribution at the previous step. The model then predicts the distribution for the next action. The total loss is the sum of the cross-entropy losses between the predicted action distributions and the ground-truth expert actions over the F-step horizon:

$$\mathcal{L}_{BC} = \sum_{k=0}^{F-1} \mathcal{L}_{CE}(\pi_\theta(\cdot|o^*_{t+k}, \hat{a}_{t+k-1}), a^*_{t+k}) \ , \qquad (1)$$

where π_θ is the policy, o^*_{t+k} and a^*_{t+k} are the expert's observation and action, and $\hat{a}_{t+k-1}$ is the action sampled from the policy's previous output.

Through experimentation, we determined the optimal parameter combination to be $H = 2$ and $F = 1$. We observed that increasing the future prediction horizon ($F > 1$) caused the model to overfit more quickly to the training data.

2.3.2 Stage 2: Reinforcement Learning Fine-tuning

Starting with the weights from the best IL checkpoint, we fine-tune the policy using **Proximal Policy Optimization (PPO)**. This on-policy RL algorithm allows the agent to explore the environment and refine its initial, cloned behavior based on direct reward signals.

To enhance the agent's understanding of spatial-temporal dynamics in human-populated environments, we integrate a **Spatial-Temporal Precognition Module** during this stage, following the design of [8]. This module introduces three socially-aware auxiliary tasks: **Human Count Estimation**, **Current Position Tracking**, and **Future Trajectory Forecasting**. These tasks are trained concurrently with the main navigation policy, using auxiliary information from the simulator. The total loss function is a weighted sum of the PPO policy loss and the auxiliary task losses:

$$\mathcal{L}_{total} = \mathcal{L}_{PPO} + \beta_{aux}\mathcal{L}_{aux} \ , \qquad (2)$$

where β_{aux} is the weight for the auxiliary loss term. The PPO objective function $\mathcal{L}_{PPO}$ is maximized to update the policy parameters [18]:

$$J^{PPO}(\theta) = \mathbb{E}_t[min(p_t(\theta)\hat{A}_t, clip(p_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t)] \ . \qquad (3)$$

Here, $p_t(\theta)$ is the probability ratio of the action under the current and old policies, and $\hat{A}_t$ is the advantage estimate [18]. This combined training approach allows the agent to simultaneously optimize its navigation behavior while learning to perceive and predict the actions of other agents in the environment.

We also experimented with the IL+RL learning rate schedule proposed in PIRLNav [18]. This approach addresses the fact that the critic is not trained during the IL phase by first freezing the actor network and exclusively training the critic. Subsequently, the learning rates for the critic and the actor are gradually converged to the same level. While this methodology is theoretically sound, it may require a greater number of training steps to be effective. The original work utilized 20 million RL training steps; however, due to time constraints, our experiment was limited to 10 million steps [18]. At this training duration, the scheduled approach did not demonstrate a significant advantage over a direct, full-parameter fine-tuning.

3. Experiments

Our work addresses the task of social navigation in a multi-agent setting. The agent, equipped with RGB-D egocentric sensors and a target coordinate (PointGoal), must navigate to a specified location within a 3D indoor environment. The core challenge is to reach the goal while avoiding collisions with both static obstacles and dynamic humanoid agents.

3.1. Experimental Setups

Dataset. We use the official data provided by the *RoboSense Challenge 2025* [23] held at IROS 2025. This competition builds upon the legacy of the *RoboDepth Challenge 2023* [24, 25] at ICRA 2023 and the *RoboDrive Challenge 2024* [26, 27] at ICRA 2024, continuing the collective effort to advance robust and scalable robot perception. Each track in this competition is grounded on an established benchmark designed for evaluating real-world robustness and generalization [8, 28–32].

Environment. We conduct all experiments in the Habitat simulator, utilizing its realistic 3D environments from the HM3D dataset and Bullet physics for dynamics. For Imitation Learning (IL), we generate a custom dataset of expert demonstrations. This is achieved by executing a proficient pre-trained Reinforcement Learning (RL) policy within the training scenes. We then filter the resulting trajectories to retain only high-quality episodes that meet stringent success and reward thresholds, ensuring the quality of our demonstration data.

Evaluation Metrics. We evaluate our agent's performance using several key metrics established by the competition. These include **SR** (Success Rate), the fraction of successful episodes; **SPL** (Success weighted by Path Length), which penalizes inefficient paths; **PSC** (Personal Space Compliance), the rate of avoiding human personal space (1.0m threshold); and **H-Coll** (Human Collision Rate), where any collision with a human results in a task failure. The final ranking is determined by a **Total Score**, a weighted combination defined as: $Total = 0.4 \times \text{SR} + 0.3 \times \text{SPL} + 0.3 \times \text{PSC}$.

3.2. Implementation Details

Model architecture. We use Resnet-50 with ImageNet pretraining as the depth backbone. We experimented with several methods to integrate RGB features into the model, ranging from direct concatenation with depth features to more complex approaches based on gated fusion and attention. However, none of these methods yielded significant performance improvements. We hypothesize two primary reasons for this outcome. First, the inherent instability of RL makes it challenging for larger network architectures to converge efficiently. Second, for the task of obstacle avoidance, depth information is largely sufficient. Consequently, incorporating RGB features via simple methods may introduce noise that interferes with the learning process rather than enhancing it.

Training Pipeline. We adopt a two-stage training paradigm: Imitation Learning (IL) pre-training followed by Reinforcement Learning (RL) fine-tuning.

1) Imitation Learning: We use Behavioral Cloning (BC) to pre-train our policy on the expert dataset described in Sec. 4.1. The model is trained for 20 epochs using the Adam optimizer with a learning rate of 1e-5. The policy operates in an auto-regressive manner, conditioned on the observation-action pairs from the two preceding timesteps to predict the current expert action. While the RNN's hidden state propagates temporal information, gradients are truncated and not back-propagated to the inputs of previous timesteps. The primary objective is to minimize the cross-entropy loss between the predicted and expert actions. The checkpoint with the highest validation success rate is selected for the next stage.

2) RL Fine-tuning: We fine-tune the IL-pretrained model using Proximal Policy Optimization (PPO). We collect experience from 8 parallel environments, updating the policy with rollouts of 64 steps each. We use Generalized Advantage Estimation (GAE) with $\gamma = 0.99$ and $\lambda = 0.95$. The policy is trained for a total of 6M environment steps with a linearly decaying learning rate.

3.3. Comparative Study

Baselines. To evaluate the effectiveness of our proposed IL+RL paradigm, we compare our full method against two fundamental baselines. 1) **RL-from-scratch**: This agent is trained using our PPO setup, but from a randomly initialized policy without expert pre-training. It demonstrates the performance achievable by pure reinforcement learning. 2) **IL-only**: This agent uses the policy obtained solely from Behavioral Cloning on the expert dataset, without any RL fine-tuning. It quantifies the performance ceiling of pure imitation.

Main Results. Table 1 presents the official evaluation results from the competition server. Our full approach, which synergizes IL and RL, demonstrates a clear performance improvement across the most critical metrics.

Method	**SR** ↑	**SPL** ↑	**PSC** ↑	**H-Coll** ↓	**Total** ↑
Baseline (RL-only)	0.5400	0.4997	0.8630	0.3920	0.6248
Ours (IL-only)	0.6280	0.5761	0.8609	0.3540	0.6823
Ours (IL+RL)	**0.6480**	**0.6010**	0.8607	**0.3420**	**0.6977**

Table 1. Quantitative results on the official competition test server.

As shown in Table 1, the **RL-from-scratch** baseline establishes a solid performance with a 0.6248 Total Score, yet it suffers from a high human collision rate (39.20%). The **IL-only** agent significantly improves upon this, boosting the success rate to 62.80% and the Total Score to 0.6823. This demonstrates the profound impact of leveraging expert demonstrations for acquiring a competent initial policy.

Our **full method (IL+RL)** achieves the best overall performance, reaching a final Total Score of **0.6977**. Notably, the RL fine-tuning stage further improves the Success Rate to 64.80% and SPL to 0.6010, while also achieving the lowest human collision rate (34.20%). This confirms our core hypothesis: IL provides a strong foundation, and RL fine-tuning is crucial for refining the policy to enhance its efficiency, safety, and robustness.

4. Conclusion

In this paper, we presented a highly effective two-stage training paradigm for the challenging task of social PointGoal navigation. Our approach synergizes the strengths of Imitation Learning (IL) and Reinforcement Learning (RL), beginning with a **Behavioral Cloning (BC)** pre-training phase on a curated expert dataset, followed by a **Proximal Policy Optimization (PPO)** fine-tuning stage. To further enhance the agent's social awareness, the RL stage was augmented with auxiliary tasks for predicting human movement.

Our extensive experiments demonstrate the superiority of this hybrid IL+RL method. It achieved a final Total Score of 0.6977, significantly outperforming both IL-only and RL-from-scratch baselines across critical metrics, including Success Rate and Human Collision Rate. These findings confirm our central hypothesis: IL provides a robust behavioral foundation by leveraging expert knowledge, while RL fine-tuning is crucial for refining the policy to enhance its efficiency, safety, and adaptability in dynamic, multi-agent environments. Ultimately, our work validates the combination of imitation and reinforcement as a practical and high-performing strategy for developing socially compliant navigation agents.

References

[1] Roya Firoozi, Johnathan Tucker, Stephen Tian, Anirudha Majumdar, Jiankai Sun, Weiyu Liu, Yuke Zhu, Shuran Song, Ashish Kapoor, Karol Hausman, Brian Ichter, Danny Driess, Jiajun Wu, Cewu Lu, and Mac Schwager. Foundation models in robotics: Applications, challenges, and the future. *The International Journal of Robotics Research*, 44(5):701–739, 2025.

[2] Yifan Zhong, Fengshuo Bai, Shaofei Cai, Xuchuan Huang, Zhang Chen, Xiaowei Zhang, Yuanfei Wang, Shaoyang Guo, Tianrui Guan, Ka Nam Lui, Zhiquan Qi, Yitao Liang, Yuanpei Chen, and Yaodong Yang. A Survey on Vision-Language-Action Models: An Action Tokenization Perspective, 2025.

[3] Zeying Gong et al. Stairway to success: An online floor-aware zero-shot object-goal navigation framework via LLM-driven coarse-to-fine exploration. *arXiv preprint arXiv:2505.23019*, 2025.

[4] Runnan Chen et al. CLIP2Scene: Towards label-efficient 3D scene understanding by CLIP. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7020–7030, 2023.

[5] Runnan Chen et al. Towards label-free scene understanding by vision foundation models. In *Advances in Neural Information Processing Systems*, volume 36, pages 75896–75910, 2023.

[6] Rong Li et al. SeeGround: See and ground for zero-shot open-vocabulary 3D visual grounding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3707–3717, 2025.

[7] Amir Hossain Raj, Zichao Hu, Haresh Karnan, Rohan Chandra, Amirreza Payandeh, Luisa Mao, Peter Stone, Joydeep Biswas, and Xuesu Xiao. Rethinking Social Robot Navigation: Leveraging the Best of Two Worlds, 2024.

[8] Zeying Gong, Tianshuai Hu, Ronghe Qiu, and Junwei Liang. From Cognition to Precognition: A Future-Aware Framework for Social Navigation, 2025.

[9] Xiaoshuai Hao et al. SafeMap: Robust HD map construction from incomplete observations. In *International Conference on Machine Learning*, pages 22091–22102. PMLR, 2025.

[10] Xiaoshuai Hao, Mengchuan Wei, Yifan Yang, et al. Is your HD map constructor reliable under sensor corruptions? In *Advances in Neural Information Processing Systems*, volume 37, pages 22441–22482, 2024.

[11] Xiaojun Lu, Angela Faragasso, Yongdong Wang, Atsushi Yamashita, and Hajime Asama. Group-Aware Robot Navigation in Crowds Using Spatio-Temporal Graph Attention Network With Deep Reinforcement Learning. *IEEE Robotics and Automation Letters*, 10(4):4140–4147, 2025.

[12] Changan Chen, Yuejiang Liu, Sven Kreiss, and Alexandre Alahi. Crowd-Robot Interaction: Crowd-aware Robot Navigation with Attention-based Deep Reinforcement Learning, 2019.

[13] Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J. Lim, Abhinav Gupta, Li Fei-Fei, and Ali Farhadi. Target-driven visual navigation in indoor scenes using deep reinforcement learning. *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3357–3364, 2017.

[14] Jinyeob Kim, Sumin Kang, Sungwoo Yang, Beomjoon Kim, Jargalbaatar Yura, and Donghan Kim. Transformable Gaussian Reward Function for Socially Aware Navigation Using Deep Reinforcement Learning. *Sensors*, 24(14):4540, 2024.

[15] Sicheng Feng, Kaiwen Tuo, Song Wang, et al. RewardMap: Tackling sparse rewards in fine-grained visual reasoning via multi-stage reinforcement learning. *arXiv preprint arXiv:2510.02240*, 2025.

[16] Haresh Karnan, Anirudh Nair, Xuesu Xiao, Garrett Warnell, Soren Pirk, Alexander Toshev, Justin Hart, Joydeep Biswas, and Peter Stone. Socially CompliAnt Navigation Dataset (SCAND): A Large-Scale Dataset of Demonstrations for Social Navigation. *IEEE Robotics and Automation Letters*, 7(4):11807–11814, 2022.

[17] Pete Florence, Corey Lynch, Andy Zeng, Oscar A. Ramirez, Ayzaan Wahid, Laura Downs, Adrian Wong, Johnny Lee, Igor Mordatch, and Jonathan Tompson. Implicit Behavioral Cloning. In *Proceedings of the 5th Conference on Robot Learning*, pages 158–168. PMLR, 2022.

[18] Ram Ramrakhya, Dhruv Batra, Erik Wijmans, and Abhishek Das. PIRLNav: Pretraining with Imitation and RL Finetuning for OBJECTNAV. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Vancouver, BC, Canada, 2023. IEEE.

[19] Liu Huajian, Dong Wei, Mao Shouren, Wang Chao, and Gao Yongzhuo. Sample-Efficient Learning-Based Dynamic Environment Navigation With Transferring Experience From Optimization-Based Planner. *IEEE Robotics and Automation Letters*, 9(8):7055–7062, 2024.

[20] Enrico Cancelli, Tommaso Campari, Luciano Serafini, Angel X. Chang, and Lamberto Ballan. Exploiting Proximity-Aware Tasks for Embodied Social Navigation. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10923–10933, 2023.

[21] Bolei Chen, Haina Zhu, Shengkang Yao, Siyi Lu, Ping Zhong, Yu Sheng, and Jianxin Wang. Socially Aware Object Goal Navigation With Heterogeneous Scene Representation Learning. *IEEE Robotics and Automation Letters*, 9(8):6792–6799, 2024.

[22] Santhosh K. Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alex Clegg, John Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X. Chang, Manolis Savva, Yili Zhao, and Dhruv Batra. Habitat-Matterport 3D Dataset (HM3D): 1000 Large-scale 3D Environments for Embodied AI, 2021.

[23] Lingdong Kong, Shaoyuan Xie, Zeying Gong, Ye Li, Meng Chu, Ao Liang, Yuhao Dong, Tianshuai Hu, Ronghe Qiu, Rong Li, Hanjiang Hu, Dongyue Lu, Wei Yin, Wenhao Ding, Linfeng Li, Hang Song, Wenwei Zhang, Yuexin Ma, Junwei Liang, Zhedong Zheng, Lai Xing Ng, Benoit R. Cottereau, Wei Tsang Ooi, Ziwei Liu, Zhanpeng Zhang, Weichao Qiu, Wei Zhang, Ji Ao, Jiangpeng Zheng, Siyu Wang, Guang Yang, Zihao Zhang, Yu Zhong, Enzhu Gao, Xinhan Zheng, Xueting Wang, Shouming Li, Yunkai Gao, Siming Lan, Mingfei Han, Xing Hu, Dusan Malic, Christian Fruhwirth-Reisinger, Alexander Prutsch, Wei Lin, Samuel Schulter, Horst Possegger, Linfeng Li, Jian Zhao, Zepeng Yang, Yuhang Song, Bojun Lin, Tianle Zhang, Yuchen Yuan, Chi Zhang, Xuelong Li, Youngseok Kim, Sihwan Hwang, Hyeonjun Jeong, Aodi Wu, Xubo Luo, Erjia Xiao, Lingfeng Zhang, Yingbo Tang, Hao Cheng, Renjing Xu, Wenbo Ding, Lei Zhou, Long Chen, Hangjun Ye, Xiaoshuai Hao, Shuangzhi Li, Junlong Shen, Xingyu Li, Hao Ruan, Jinliang Lin, Zhiming Luo, Yu Zang, Cheng Wang, Hanshi Wang, Xijie Gong, Yixiang Yang, Qianli Ma, Zhipeng Zhang, Wenxiang Shi, Jingmeng Zhou, Weijun Zeng, Kexin Xu, Yuchen Zhang, Haoxiang Fu, Ruibin Hu, Yanbiao Ma, Xiyan Feng, Wenbo Zhang, Lu Zhang, Yunzhi Zhuge, Huchuan Lu, You He, Seungjun Yu, Junsung Park, Youngsun Lim, Hyunjung Shim, Faduo Liang, Zihang Wang, Yiming Peng, Guanyu Zong, Xu Li, Binghao Wang, Hao Wei, Yongxin Ma, Yunke Shi, Shuaipeng Liu, Dong Kong, Yongchun Lin, Huitong Yang, Liang Lei, Haoang Li, Xinliang Zhang, Zhiyong Wang, Xiaofeng Wang, Yuxia Fu, Yadan Luo, Djamahl Etchegaray, Yang Li, Congfei Li, Yuxiang Sun, Wenkai Zhu, Wang Xu, Linru Li, Longjie Liao, Jun Yan, Benwu Wang, Xueliang Ren, Xiaoyu Yue, Jixian Zheng, Jinfeng Wu, Shurui Qin, Wei Cong, and Yao He. The RoboSense challenge: Sense anything, navigate anywhere, adapt across platforms. https://robosense2025.github.io, 2025.

[24] Lingdong Kong, Yaru Niu, Shaoyuan Xie, Hanjiang Hu, Lai Xing Ng, Benoit Cottereau, Liangjun Zhang, Hesheng Wang, Wei Tsang Ooi, Ruijie Zhu, Ziyang Song, Li Liu, Tianzhu Zhang, Jun Yu, Mohan Jing, Pengwei Li, Xiaohua Qi, Cheng Jin, Yingfeng Chen, Jie Hou, Jie Zhang, Zhen Kan, Qiang Lin, Liang Peng, Minglei Li, Di Xu, Changpeng Yang, Yuanqi Yao, Gang Wu, Jian Kuai, Xianming Liu, Junjun Jiang, Jiamian Huang, Baojun Li, Jiale Chen, Shuang Zhang, Sun Ao, Zhenyu Li, Runze Chen, Haiyong Luo, Fang Zhao, and Jingze Yu. The RoboDepth challenge: Methods and advancements towards robust depth estimation. *arXiv preprint arXiv:2307.15061*, 2023.

[25] Lingdong Kong, Shaoyuan Xie, Hanjiang Hu, Lai Xing Ng, Benoit R. Cottereau, and Wei Tsang Ooi. RoboDepth: Robust out-of-distribution depth estimation under corruptions. In *Advances in Neural Information Processing Systems*, volume 36, pages 21298–21342, 2023.

[26] Lingdong Kong, Shaoyuan Xie, Hanjiang Hu, Yaru Niu, Wei Tsang Ooi, Benoit R. Cottereau, Lai Xing Ng, Yuexin Ma, Wenwei Zhang, Liang Pan, Kai Chen, Ziwei Liu, Weichao Qiu, Wei Zhang, Xu Cao, Hao Lu, Ying-Cong Chen, et al. The RoboDrive challenge: Drive anytime anywhere in any condition. *arXiv preprint arXiv:2405.08816*, 2024.

[27] Shaoyuan Xie, Lingdong Kong, Wenwei Zhang, Jiawei Ren, Liang Pan, Kai Chen, and Ziwei Liu. Benchmarking and improving bird's eye view perception robustness in autonomous driving. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(5):3878–3894, 2025.

[28] Shaoyuan Xie, Lingdong Kong, Yuhao Dong, Chonghao Sima, Wenwei Zhang, Qi Alfred Chen, Ziwei Liu, and Liang Pan. Are VLMs ready for autonomous driving? an empirical study from the reliability, data, and metric perspectives. In *IEEE/CVF International Conference on Computer Vision*, pages 6585–6597, 2025.

[29] Ye Li, Lingdong Kong, Hanjiang Hu, Xiaohao Xu, and Xiaonan Huang. Is your LiDAR placement optimized for 3D scene understanding? In *Advances in Neural Information Processing Systems*, volume 37, pages 34980–35017, 2024.

[30] Meng Chu, Zhedong Zheng, Wei Ji, Tingyu Wang, and Tat-Seng Chua. Towards natural language-guided drones: GeoText-1652 benchmark with spatial relation matching. In *European Conference on Computer Vision*, pages 213–231, 2024.

[31] Ao Liang et al. Perspective-invariant 3D object detection. In *IEEE/CVF International Conference on Computer Vision*, pages 27725–27738, 2025.

[32] Rong Li, Yuhao Dong, Tianshuai Hu, et al. 3EED: Ground everything everywhere in 3D. *arXiv preprint arXiv:2511.01755*, 2025.