



OPEN ACCESS

EDITED BY

Allan Wang,
Miraikan – The National Museum of Emerging
Science and Innovation, Japan

REVIEWED BY

Suresh Kumar Jayaraman,
Carnegie Mellon University, United States
Yigit Yildirim,
Bogazici Universitesi Muhendislik
Fakultesi, Türkiye

*CORRESPONDENCE

Rashid Alyassi,
✉ ralyassi@ethz.ch

RECEIVED 02 July 2025

REVISED 15 October 2025

ACCEPTED 27 October 2025

PUBLISHED 11 December 2025

CITATION

Alyassi R, Cadena C, Riener R and
Paez-Granados D (2025) Social robot
navigation: a review and benchmarking of
learning-based methods.
Front. Robot. AI 12:1658643.
doi: 10.3389/frobt.2025.1658643

COPYRIGHT

© 2025 Alyassi, Cadena, Riener and
Paez-Granados. This is an open-access article
distributed under the terms of the [Creative
Commons Attribution License \(CC BY\)](https://creativecommons.org/licenses/by/4.0/). The
use, distribution or reproduction in other
forums is permitted, provided the original
author(s) and the copyright owner(s) are
credited and that the original publication in
this journal is cited, in accordance with
accepted academic practice. No use,
distribution or reproduction is permitted
which does not comply with these terms.

Social robot navigation: a review and benchmarking of learning-based methods

Rashid Alyassi^{1,2,3*}, Cesar Cadena⁴, Robert Riener² and
Diego Paez-Granados^{1,2,3}

¹Spinal Cord Injury and Artificial Intelligence Lab, D-HEST, ETH Zurich, Zürich, Switzerland,

²Sensory-Motor Systems Lab, Institute of Robotics and Intelligent Systems, ETH Zurich, Zürich,

Switzerland, ³Digital Healthcare and Rehabilitation, Swiss Paraplegic Research, Nottwil, Switzerland,

⁴Robotics Systems Lab, Institute of Robotics and Intelligent Systems, ETH Zurich, Zürich, Switzerland

For autonomous mobile robots to operate effectively in human environments, navigation must extend beyond obstacle avoidance to incorporate social awareness. Safe and fluid interaction in shared spaces requires the ability to interpret human motion and adapt to social norms—an area that is being reshaped by advances in learning-based methods. This review examines recent progress in learning-based social navigation methods that deal with the complexities of human-robot coexistence. We introduce a taxonomy of navigation methods and analyze core system components, including realistic training environments and objectives that promote socially compliant behavior. We conduct a comprehensive benchmark of existing frameworks in challenging crowd scenarios, showing their advantages and shortcomings, while providing critical insights into the architectural choices that impact performance. We find that many learning-based approaches outperform model-based methods in realistic coordination scenarios such as navigating doorways. A key highlight is the end-to-end models, which achieve strong performance by directly planning from raw sensor input, enabling more efficient and adaptive navigation. This review also maps current trends and outlines ongoing challenges, offering a strategic roadmap for future research. We emphasize the need for models that accurately anticipate human movement, training environments that realistically simulate crowded spaces, and evaluation methods that capture real-world complexity. Advancing these areas will help overcome current limitations and move social navigation systems closer to safe, reliable deployment in everyday environments. Additional resources are available at: <https://socialnavigation.github.io>.

KEYWORDS

social navigation, human-robot interaction, reinforcement learning, robot learning, human-aware navigation, path planning

1 Introduction

Social navigation enables robots to move safely and efficiently in human-shared environments while respecting social norms and prioritizing human comfort. It builds on standard collision avoidance navigation by incorporating behaviors such as maintaining social distance, interpreting social cues, and predicting human movements.

As a key component of *Human-Robot Interaction (HRI)*, social navigation focuses on understanding and enhancing interactions between humans and robots in shared environments.

The importance of social navigation was recognized as early as the 1990s with pioneering robots like *RHINO* (Burgard et al., 1999) and *MINERVA* (Thrun et al., 2000), which operated in dynamic environments such as museums, requiring socially aware navigation systems to interact effectively with visitors. Since then, social navigation has gained research interest, leading to steady advancements over the past years.

Several review papers reflect the interdisciplinary nature of social navigation. Sociological and human factors are addressed by Rios-Martinez et al. (2015), who apply proxemics theory, and Thomaz et al. (2016), who review computational human-robot interaction. Perception and mapping in social contexts are discussed by Charalampous et al. (2017), while safety in human-robot interaction is analyzed by Lasota et al. (2017). Path planning and navigation are extensively reviewed by Mohanan and Salgoankar (2018), Sánchez-Ibáñez et al. (2021), and Zhou et al. (2022), although mainly for classical methods. For social navigation specifically, recent surveys cover human-aware navigation (Kruse et al., 2013), conflict prevention (Mirsky et al., 2021), visual navigation (Möller et al., 2021), evaluation (Gao and Huang, 2022; Mavrogiannis et al., 2023), and taxonomy (Singamaneni et al., 2024). Human motion prediction surveys include Rudenko et al. (2020a), Sighencea et al. (2021), and Korbmacher and Tordeux (2022), comparing data-driven and model-based approaches. However, there remains a gap for a comprehensive survey focused on learning-based social navigation approaches.

This survey advances learning-based social navigation by comprehensively reviewing recent methods and introducing a novel taxonomy that categorizes algorithms into five groups by neural network architecture and system modules, expanding on earlier works like Zhu and Zhang (2021). We examine key system components, including human detection, tracking, prediction, and crowd simulation. Furthermore, our conclusions are grounded in an experimental benchmark over state-of-the-art social navigation algorithms, featuring challenging scenarios such as corridors, doorways, and intersections—areas often overlooked in previous surveys (Mavrogiannis et al., 2023). By rigorously comparing existing methods, we identify best practices, evaluate algorithm performance on new scenarios, and highlight open challenges and future directions, providing a comprehensive guide for developing learning-based social navigation systems.

The structure of this survey is as follows: Section 1 introduces a taxonomy of the social navigation problem. Section 2 presents the proposed taxonomy of social navigation algorithms and reviews recent learning-based methods. In Section 3, we examine training processes for navigation models, including discussions on objective functions, crowd simulation, and methods for human detection, tracking, and prediction. Section 4 presents an experimental comparison to validate our analysis by evaluating multiple algorithms across various simulated scenarios. Finally, Section 5 provides a discussion of existing challenges and proposes future research directions to advance social navigation.

1.1 Social navigation problem

Social navigation refers to a robot's ability to navigate environments while considering human presence, social norms, and behaviors. This field encompasses a variety of navigation tasks, broadly classified into three main categories: independent, assistive, and collaborative navigation (Singamaneni et al., 2024).

1.1.1 Independent

Independent crowd-aware navigation involves robots autonomously reaching goals in human-populated environments while minimizing disruption, as seen with service robots in malls or airports integrating into pedestrian flows (Yao et al., 2019). This includes systems designed for joining moving groups (Truong and Ngo, 2017) or avoiding stationary crowds (Tsoi et al., 2022). Independent navigation is the most widely studied and versatile form of social navigation.

1.1.2 Assistive

Assistive navigation tasks involve robots directly supporting humans, such as follower robots in airports (Gupta et al., 2016), shopping assistants (Chen Y. et al., 2017), interactive guides (Burgard et al., 1999; Thrun et al., 2000), and systems aiding visually impaired individuals (Chuang et al., 2018), or accompanying people and groups (Ferrer et al., 2017; Repiso et al., 2020). Some tasks include proactively offering guidance (Kato et al., 2015). These tasks require detecting, following, and interpreting human cues for safe and seamless assistance.

1.1.3 Collaborative

Collaborative navigation features robots and humans working together on shared tasks, either physically or through shared control. In industry, cobots assist on assembly lines (Matheson et al., 2019), while human mobility robots use shared-control systems, model-based (Gonon et al., 2021) or learning-based (Zhang et al., 2023) to integrate human input and dynamically adapt to real-time feedback.

In addition to task-based classification, social navigation can be categorized by communication strategies, focusing on how robots interact with humans through signals. For a more in-depth discussion on taxonomy, refer to Singamaneni et al. (2024) and Mirsky et al. (2021).

This review focuses on independent (crowd-aware) navigation due to its broad applicability. Its core principles can be extended to assistive and collaborative tasks, making it a more general foundation for various social navigation tasks.

2 Social navigation algorithms

This section explores a range of learning-based social navigation algorithms designed for crowd-aware robot navigation. These methods function as local planners and require integration with a global planner for long-term navigation. Learning-based social navigation enables robots to navigate safely around humans through trial and error or imitation. The algorithms are categorized based on their neural network architecture and the specific modules they require, such as human detection, tracking, and prediction. This classification organizes social navigation

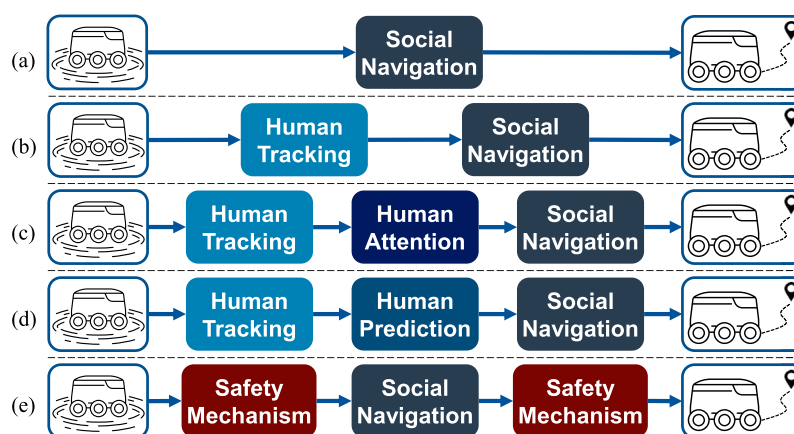


FIGURE 1

Taxonomy of Social Navigation Based on Architecture and Components outlined in Sections 2.1–2.5: (a) End-to-End, (b) Human Position-based, (c) Human Attention-based, (d) Human Prediction-based, (e) Safety-aware.

strategies into five main categories, ranging from simpler end-to-end models to sophisticated multi-policy and prediction-based methods (see Figure 1). Furthermore, within each category, we outline several subtopics that describe common methodological themes. These themes are prevalent in certain categories but are not necessarily unique to them.

2.1 End-to-end navigation

End-to-end reinforcement learning (RL) (see Table 1) has proven highly effective across domains like robot navigation and autonomous driving (Bojarski, 2016). In end-to-end RL, the policy maps observations directly to actions, bypassing predefined intermediary steps and enabling complex behavior learning through trial and error. Typically, the robot's state $s_t = [z, v_x, v_y, d_{gx}, d_{gy}]$, where z represents sensory inputs, and other navigation-related parameters such as velocity v and goal-relative distance d_g .

Q-learning is one of the earliest learning-based navigation methods, initially designed for static environments (Smart and Kaelbling, 2000; Smart and Kaelbling, 2002; Yang et al., 2004) and later extended to dynamic settings (Yen and Hickey, 2004; Costa and Gouvea, 2010; Jaradat et al., 2011). For instance, Wang et al. (Wang Y. et al., 2018) use a two-stream Q-network (Simonyan and Zisserman, 2014) that processes spatial (current LiDAR) and temporal (scan-difference) inputs to explicitly capture obstacle motion. These streams are processed and combined via fully connected layers, enabling effective detection of moving obstacles. While historically notable, end-to-end Q-learning is now rarely used in social navigation due to its difficulty in handling the continuous action spaces needed for smooth, realistic motion.

Actor-critic methods are widely used for continuous action spaces, addressing Q-learning's limitations. Actor-critic models have been applied to both static (Tai et al., 2017; Zhang et al., 2017; Gao et al., 2020) and dynamic environments (Faust et al., 2018; Chiang et al., 2019). For instance, Hoeller et al. (2021) employ the PPO algorithm in combination with an LSTM network to train a robot to navigate a simulated environment. To train for dynamic

collision avoidance, the environment is populated with both static and dynamic (constant-velocity) obstacles.

An alternative to using dynamic obstacles for collision avoidance training is multi-agent reinforcement learning (MARL). MARL often leverages the concept of *centralized learning with decentralized execution* to develop cooperative navigation policies (Zhang et al., 2021). In this setup, all agents are trained within a shared environment, with each agent aiming to reach its designated goal while avoiding collisions with others (Chen W. et al., 2019; Tan et al., 2020). It's *decentralized* since there is no direct communication between agents; however, the training is *centralized* since agents share the same policy parameters and update their experiences collectively during training. For instance, Long et al. (Long et al., 2018) implemented a parallel PPO algorithm to train multiple agents to navigate in simulation. The policy is conditioned on relative goal information and 2D LiDAR data from the past three time steps, which is processed by a 1D CNN. This approach was later validated with real-world scenarios (Fan et al., 2018). Although agents trained through MARL efficiently learn to avoid collisions with other agents running an identical policy, the approach is often sub-optimal in social navigation contexts, since we assume that all agents exhibit similar behaviors, which may not reflect the diverse and adaptive behaviors in real social interactions.

An alternative to MARL is training navigation policies with simulated crowds. Here, simulated humans exhibit cooperative or reactive behaviors resembling real crowds, enabling agents to adapt to diverse social settings. For instance, Liang et al. (2021) uses PPO to train agents among cooperative, human-like agents that follow predefined paths and preferred velocities, adjusting their speed based on available space (Narang et al., 2015). Conversely, Jin et al. (2020) trains a DDPG-based policy in simulation with non-cooperative, ORCA-modeled humans (Van Den Berg et al., 2011), who react to obstacles and others without considering the robot's path. The agent's state is captured by multiple 2D LiDAR scans, decoupled from its motion and adjusted for heading differences over time, effectively highlighting dynamic obstacles independently of the robot's motion.

TABLE 1 End-to-end social navigation algorithms.

Method	Input	Architecture	Algorithm	Output	Training sim	Real-world demo	Code	Training type
Wang et al. (2018a)	2D LiDAR	1D CNN + FC	Q-learning	Discrete (v, ω)	Custom Sim	None	No	Constant velocity (overlapping) obstacles
Hoeller et al. (2021)	RGB-D	VAE + LSTM + FC	PPO	(v_x, v_y, ω)	Isaac Gym	Demo + Video	No	Constant velocity obstacles
CrowdMove (Fan et al., 2018)	2D LiDAR	1D CNN + FC	PPO	(v, ω)	Stage simulator	Demo + Video	No	Multi-agent
Liang et al. (2021)	RGB-D + 2D LiDAR	2D CNN + 1D CNN + FC	PPO	(v, ω)	Gazebo	Demo + Video	No	Agent-Based Crowd Sim (Narang et al., 2015)
Jin et al. (2020)	2D LiDAR	2D CNN + FC	DDPG	(v, ω)	Custom Sim	Demo + Video	No	ORCA (non-cooperative) crowd
Tai et al. (2018)	Depth image	ResNet-50 + FC + 2D CNN	GAIL + TRPO	(v, ω)	Gazebo	Demo + Video	Yes	SFM crowd sim
NavRep (Dugas et al., 2021)	2D LiDAR	LSTM + VAE + FC	World Models + PPO	(v_x, v_y, ω)	Custom Sim	Demo + Video	Yes	ORCA + Constant Velocity + Global planning crowd
Cui et al. (2021)	2D LiDAR	LSTM + 3D CNN + FC	TD3	(v, ω)	Gazebo	Demo	Yes	Multi-agent
Han et al. (2022a)	RGB + 2D LiDAR	2D CNN + FC	PPO	(v, ω)	Stage simulator	Demo	No	Multi-agent
Choi et al. (2019)	2D LiDAR	2D CNN + 1D CNN + LSTM + FC	SAC	(v, ω)	Custom Sim	Demo + Video	No	Multi-agent
Cheng et al. (2023)	2D LiDAR + Preference	FC	Q-Learning	Discrete (v, ω)	Custom Sim	None	No	SFM crowd sim
Choi et al. (2020)	2D LiDAR	GRU + FC	SAC	(v, ω)	Custom Unity Sim	Demo + Video	No	Multi-agent
Lee et al. (2023)	2D LiDAR	1D CNN + FC	SAC + HER	(v, ω)	Custom Sim	Demo	Yes	Learning-based (Long et al., 2018) + global planning crowd sim

2.1.1 Learning from demonstration

Imitation learning (IL) enables learning an end-to-end policy directly from expert demonstrations, bypassing the need for hand-crafted rewards. While inverse reinforcement learning (IRL) infers a reward function from human demonstrations or pedestrian datasets (Kim and Pineau, 2016; Fahad et al., 2018) then learns a policy, behavioral cloning (BC) learns actions directly from demonstrations but struggles in dynamic settings due to its reliance on fixed data. More advanced IL approaches aim to overcome these limitations. One of the earliest data-driven approaches for static

obstacle navigation, proposed by Pfeiffer et al. (2017), uses a goal-conditioned model with 1D CNN and pooling layers trained using BC. The model takes in 2D LiDAR readings and goal information to predict actions and is trained on demonstration data collected using the dynamic window approach (DWA) planner (Fox et al., 1997). While effective in static environments, this approach does not incorporate past observations, reducing its effectiveness in dynamic obstacle scenarios. Similarly, CANet (Long et al., 2017) applies behavioral cloning to learn a navigation policy from multi-agent data generated using ORCA planner (Van Den Berg et al., 2011).

The model is an MLP trained to output a probability distribution over 61 pre-defined 2D velocity clusters, capturing a range of socially aware navigational behaviors. A value iteration network (VIN)-based planner, proposed by Liu et al. (2018), applies VIN (Tamar et al., 2016) to social navigation. VIN introduces a neural network architecture with a differentiable planning module that approximates the classical value iteration algorithm. Given a reward map and local transition model, VIN iteratively maps rewards and previous value estimates into Q-values using convolutional layers, where each channel corresponds to an action's outcome. A channel-wise max pooling layer retrieves the maximum over actions, yielding the updated value function, which is then used by a greedy reactive policy network (e.g., softmax) to generate an action distribution. Liu et al. (2018) extend VIN by adding an MLP that combines the VIN output with the robot's velocity to predict actions. Trained in a supervised manner on real and synthetic maps with demonstration actions derived from a reactive optimization-based planner, this approach provides a novel perspective on navigation. However, it's limited to static environments and needs to be extended to dynamic settings with crowds. Another approach, GAIL, is used by Tai et al. (2018) to train a navigation policy. GAIL employs a generator (policy) that processes depth images to predict actions, while a discriminator distinguishes between the generator's actions and expert demonstrations. To stabilize training, the discriminator is defined as a regression network, inspired by WGAN (Arjovsky et al., 2017), rather than a standard classifier. Initially, the policy is pre-trained with behavioral cloning on expert data and then fine-tuned using TRPO with the discriminator. The main advantage of GAIL is its use of online simulation-based training, which helps mitigate generalization issues. MuSoHu (Nguyen et al., 2023) addresses data scarcity in data-driven navigation by providing a large-scale dataset of 100 km of human navigation patterns collected with a helmet-mounted sensor suite. Applying behavioral cloning on this dataset produces a human-like path-planning policy that mitigates behavior modeling inaccuracies and shows strong real-world performance. DeepMoTion (Hamandi et al., 2019) aims to mimic human pedestrian behavior by using imitation learning to train a navigation policy. The approach uses pedestrian datasets to simulate human-centric LiDAR data, training an LSTM-based policy through supervised learning. The model predicts the pedestrian's future direction and velocity based on its LiDAR data and final goal. To account for variability in human behavior, it employs a Gaussian distribution for direction prediction, enabling the capture of diverse movement patterns in similar scenarios.

2.1.2 Model-based RL

World models provide agents with internal representations of environment dynamics, enabling more informed, end-to-end decision-making. One prominent example is NavRep (Dugas et al., 2021), which integrates the World Model framework (Ha and Schmidhuber, 2018) with the PPO algorithm to train a policy. NavRep introduces *rings*, a novel 2D LiDAR representation that arranges data into exponentially spaced radial intervals within a polar coordinate grid, enhancing close-range resolution. Similarly, Cui et al. (2021) applies world models with the TD3 algorithm in a MARL framework, with the state represented by stacked 2D obstacle maps generated from multiple LiDAR scans.

2.1.3 Enhanced perception methods

Most methods discussed so far rely on a single sensor input, which can be prone to noise and limited in accuracy. To enhance perception robustness for end-to-end systems, sensor fusion techniques are employed. For example, Liang et al. (2021) processes 2D LiDAR data using a 1D CNN and depth images using a 2D CNN, with inputs collected over three consecutive time steps, and combines the outputs through concatenation. In another approach, Han et al. (Han Y. et al., 2022) propose a fusion network that integrates RGB images and 2D LiDAR data to produce depth information. The 2D LiDAR data is first transformed into the camera's coordinate frame, then combined with RGB data through an encoder-decoder CNN network (Ma and Karaman, 2018) to produce a depth image. The depth image is processed by a self-attention module, which prioritizes pixels based on factors such as robot type, goal position, and velocity, thus enhancing the agent's situational awareness. Some navigation systems focus on optimizing performance with sensors that have limited fields of view. In these setups, self-supervised and supervised approaches are used to improve the agent's situational awareness. For example, Choi et al. (2019) employ an actor-critic algorithm where the actor network uses an LSTM, while the critic receives additional information, such as a local 2D map. This approach allows the actor to rely on temporal cues, while the critic aids in evaluating action choices more accurately. Similarly, Monaci et al. (2022) introduce a method where an initial policy is trained using privileged information, such as precise human positions within the environment. This policy is subsequently distilled into a non-privileged policy that learns to approximate the privileged information through supervised learning.

2.1.4 Multi-objective and hierarchical RL

Multi-objective reinforcement learning (MORL) (Roijers et al., 2013) frameworks are increasingly applied in end-to-end navigation tasks where agents must balance multiple, often conflicting, objectives. MORL allows a policy to be trained on multiple different objectives, enabling the adjustment of objective weightings, referred to as a preference vector, during deployment (Hayes et al., 2022). This flexibility is particularly beneficial in dynamic social environments, where safety, efficiency, and comfort are key yet sometimes competing. For example, Cheng et al. (2023) implement a vectorized Q-learning-based MORL algorithm to train a policy with a simulated crowd. Meanwhile, Choi et al. (2020) use the SAC MORL algorithm to train a navigation policy with a preference vector learned from human feedback, sampled through a Bayesian neural network (Blundell et al., 2015). Hierarchical reinforcement learning (HRL) divides complex tasks into manageable sub-tasks or sub-goals, allowing an agent to focus on different levels of decision-making. In HRL architectures, the high-level policy selects sub-goals, while the low-level policies execute these sub-goals through specific navigation actions. For instance, Lee et al. (2023) propose an HRL framework in which the high-level policy focuses on reaching the goal efficiently, minimizing time-to-goal. This policy generates a skill vector, which is then interpreted by the low-level policy to execute specific navigation skills, such as collision avoidance, goal-reaching, and maintaining a safe distance. Both levels of policy utilize 2D LiDAR data and goal state information. Other HRL approaches offer variations in task distribution and

shared information. [Zhu and Hayashibe \(2022\)](#) use a high-level policy as a safety controller to halt the low-level policy if necessary, while [Wang et al. \(2021\)](#) implement an HRL framework in which the high-level policy shares a sub-goal with the low-level navigation policy.

2.1.5 Vision-based navigation

In vision-based end-to-end navigation, RGB or RGB-D cameras provide input for agents to reach goals specified by relative position (PointGoal), target images (ImageGoal), or instructions (Vision-Language Navigation). These planners excel in visually rich settings without global maps, relying solely on relative goal information. Policies typically use CNN-RNN architectures, where CNNs process images and RNNs build an internal map ([Kulhánek et al., 2019](#)). Even *blind* agents, lacking vision but using memory-based policies, can navigate efficiently via spatial awareness and wall-following strategies ([Wijmans et al., 2023](#)). Such methods use photorealistic simulators based on real-world scans ([Chang et al., 2017](#)) and often employ discrete actions for training efficiency. Vision-based social navigation is emerging, with proximity-aware ([Cancelli et al., 2023](#)) and Falcon ([Gong et al., 2024](#)) methods using auxiliary tasks to better anticipate and navigate around pedestrians and obstacles.

2.1.6 Language models in navigation

Vision-language models (VLMs) are powerful multimodal models with the ability to support navigation through reasoning, visual grounding, and contextual understanding. Early work on vision-language navigation (VLN) ([Anderson et al., 2018a](#)) introduced text-based high-level planning, which can be extended to social navigation for local decision-making ([Li et al., 2024](#)). Beyond high-level planning, several recent hybrid methods integrate VLMs directly into the social navigation pipeline. [Song et al. \(2024\)](#) use a VLM to select high-level direction and speed, which are integrated with goal and obstacle costs in a model-based planner, with weights determined through an additional VLM prompt. GSON ([Luo et al., 2025](#)) leverages VLMs to detect social groups and integrates the results into an MPC planner to generate paths that avoid them. OLiVia-Nav ([Narasimhan et al., 2025](#)) distills social context from a large VLM into lightweight encoders that provide semantic inputs to a trajectory planner, which then generates candidate motions and selects the one most aligned with captions distilled from expert demonstrations. OLiVia-Nav further incorporates lifelong learning to update its encoders with new data. Related to this, [Okunevich et al. \(2025\)](#) introduce an online learning approach that adapts a social module in real time, updating the social cost function during deployment. Alternatively, coding-capable large language models (LLMs) have been prompted to generate reward functions from natural language preference descriptions ([Ma et al., 2023](#)), with applications in navigation and preference alignment ([Wang et al., 2024](#)). Social-LLaVA ([Payandeh et al., 2024](#)) leverages a VLM fine-tuned for social robot navigation to directly map decisions onto a predefined set of low-level navigation primitives. Despite this progress, the slow inference and high computational demands of VLMs currently limit their use for real-time reactive social navigation. As a result, they are mostly applied as global planners, semantic encoders, or social-context modules, while their broader potential remains underexplored.

2.1.7 Self-supervised learning

Beyond RL, self-supervised methods enable partial or full training of navigation policies using generated labels. For example, [Hoeller et al. \(2021\)](#) train a VAE to encode depth data, filter noise, and enhance sim-to-real transfer, providing informative representations for faster RL training. [Yang et al. \(2023\)](#) propose a bi-level framework where a neural network predicts waypoints optimized through a differentiable ESDF-based cost function; while deployment is simplified by using a spline to fit waypoints. [Roth et al. \(2024\)](#) further incorporate semantic costmaps, though dynamic obstacle avoidance remains unevaluated.

Overall, end-to-end navigation directly maps sensor inputs to actions and supports continuous actions, multi-agent training, model-based RL, multi-objective and hierarchical frameworks, VLMs, and self-supervised learning. However, challenges remain in ensuring safety and robustness.

2.2 Human position-based navigation

The challenging nature of collision avoidance in navigation has led to methods that rely on known positions and velocities of dynamic obstacles, such as humans (see [Table 2](#)). These positions are obtained through a detection and tracking module (see [Section 3.3](#)), allowing the robot to account for surrounding agents in its navigation decisions. In this setup, the human state is often represented as $s^h = [p_x^h, p_y^h, v_x^h, v_y^h]$, with the position p^h and velocity v^h defined in the robot frame.

A foundational approach in human position-based navigation is Collision Avoidance with Deep Reinforcement Learning (CADRL), introduced by [Chen et al. \(2017b\)](#). CADRL uses a model-based RL framework to learn a value function over the joint state space of the robot and surrounding agents. The optimal action is derived as $\pi(s) = \arg \max_{a \in \mathcal{A}} V(\hat{\mathcal{T}}(s, a))$, where \mathcal{A} is a set of sampled actions, V is the learned value function, and $\hat{\mathcal{T}}$ represents the estimated transition dynamics. In CADRL, the transition dynamics model for humans is estimated using a simplified constant velocity model.

Building on CADRL, Socially Aware CADRL (SA-CADRL) ([Chen et al., 2017c](#)) incorporates social norms, such as overtaking, directly into the reward function. The value function in SA-CADRL is computed over a fixed set of agents and is trained similarly to CADRL using the multi-agent reinforcement learning (MARL) framework. Further advancements, such as GA3C-CADRL ([Everett et al., 2018](#)), extend SA-CADRL by applying the A3C algorithm and integrating an LSTM layer, enabling the policy to process an arbitrary number of agents as input, thereby increasing scalability in crowded environments. Additionally, GA3C-CADRL simplifies the reward structure by removing explicit social norms. Further research by [Everett et al. \(2021\)](#) explores the impact of the LSTM on this model's performance in complex, multi-agent scenarios. While GA3C-CADRL performs well, using an LSTM to encode multiple agents may affect consistency due to LSTM's sensitivity to input order.

A range of methods leverage the concept of velocity obstacles (VO) in state or reward functions to promote collision avoidance in navigation policies. [Han R. et al. \(2022\)](#) propose an RL policy that uses reciprocal velocity obstacles (RVO) ([Van den Berg et al., 2008](#)) to model agent interactions. The policy processes RVO

TABLE 2 Human position-based social navigation algorithms.

Method	Input	Architecture	Algorithm	Output	Training simulator	Real-world demo	Code	Training type	Detection/Tracking
CADRL (Chen et al., 2017b)	Human pos. and vel	FC	Deep V-learning	Sampled (v_x, v_y)	CADRL Sim	Demo + Video	Yes	Multi-agent	LiDAR clustering (Campbell et al., 2013)
SA-CADRL (Chen et al., 2017c)	Human pos. and vel	FC	Deep V-learning	Sampled (v_x, v_y)	CADRL Sim	Demo + Video	Yes	Multi-agent	LiDAR + RGB detection (Miller et al., 2016) LiDAR tracking (Campbell et al., 2013)
GA3C-CADRL (Everett et al., 2018)	Human pos. and vel	LSTM + FC	A3C	Sampled (v_x, v_y)	CADRL Sim	Demo + Video	Yes	Multi-agent	LiDAR + RGB detection (Miller et al., 2016) (Liu et al., 2016), LiDAR tracking (Campbell et al., 2013)
DenseCAvoid (Sathyamoorthy et al., 2020a)	RGB-D + 2D LiDAR + Human pos. and vel	2D CNN + FC + 1D CNN	PPO	(v, ω)	Custom Gazebo Sim	Demo + Video	No	Random non-cooperative crowd	YOLOv3 detection (Redmon, 2018) RobustTP prediction (Chandra et al., 2019)
Han et al. (2022b)	RVO 6D vector + Distance+ Reciprocal collision time	BiGRU + FC	PPO	($\Delta v_x, \Delta v_y$)	Custom Sim	Demo (Robot-Robot) + Video	Yes	Multi-agent	None
DRL-VO (Xie and Dames, 2023)	2D LiDAR + Human pos. and vel	2D CNN + FC	PPO	(v, ω)	Custom Gazebo Sim	Demo + Video	Yes	SFM crowd (Gloor, 2016)	YOLOv3 detection (Redmon, 2018) + MHT tracking (Yoon et al., 2018)
ILPP (Qin et al., 2021)	Human position and velocity + 2D LiDAR + Global path	2D CNN + FC + Attn Block	BC	Confidence Map (guides A*planner)	None	Demo	No	Data-driven	LiDAR Detection + Kalman filter and Hungarian Algorithm
De Heuvel et al. (2022)	Human pos. and heading	FC	TD3	(v, ω)	Pybullet Simulator	Demo + Video	No	Static human	None
De Heuvel et al. (2023)	Depth image + Human pos	VAE (CNN) + LSTM + FC	TD3	(v, ω)	iGibson simulator	None	No	Static human	None
Marta et al. (2023)	2D LiDAR (labeled rays) + Human pos. and vel	FC	PPO	(a_x, a_y) + SFM force	Custom Unity Sim	None	No	SFM crowd	None
De Heuvel et al. (2024)	2D LiDAR + Human pos	FC	MORL-TD3	(v, ω)	iGibson sim	Demo + Video	No	Static human	None

parameters, including a 6D vector (preferred velocity and boundary velocities), distance, and reciprocal collision time for each human, using a bi-directional RNN (BiGRU). The reward function penalizes overlapping RVO areas. Some approaches, such as DRL-VO (Xie and Dames, 2023) and DenseCAvoid (Sathyamoorthy et al., 2020a), incorporate both human positions and sensor data to handle static obstacle avoidance in navigation. DRL-VO combines human positions with 2D LiDAR data, leveraging a VO-based reward function to encourage collision-free trajectories. This fusion of human position data with LiDAR enables effective static and dynamic obstacle avoidance. Similarly, DenseCAvoid uses the PPO algorithm to train a policy that fuses 2D LiDAR and RGB-D data for enhanced static obstacle detection. Building on an architecture similar to Liang et al. (2021), DenseCAvoid integrates single-step human motion predictions using RobustTP (Chandra et al., 2019), enabling the model to anticipate human movements in dynamic environments.

ILPP (Qin et al., 2021) applies imitation learning to generate a navigation confidence map that modifies the global path to incorporate collision avoidance. To produce a confidence map, the model takes LiDAR data, global path, pedestrian positions and velocities, and robot odometry. Additionally, ILPP predicts when global re-planning is necessary, especially if the expert path deviates from the global path. The model is trained using 1.3 h of a human driver operating a motorized wheelchair. To derive a path from the confidence map, the destination is set where the goal path meets the grid edge, and an A* planner finds the lowest-cost route to the destination, which is then smoothed using Gaussian filtering before being executed by a low-level controller.

2.2.1 Preference-aware navigation

Approaches that incorporate human demonstrations and preferences into policy training have proven effective for aligning robot behavior with human expectations in social navigation. De Heuvel et al. (2022) use the SAC algorithm with behavioral cloning to train a policy in simulation, closely fitting human demonstration trajectories collected via a VR pointer. This work is extended in De Heuvel et al. (2023) by adding a perception pipeline that predicts future human positions. Building on this, De Heuvel et al. (2024) employ MORL-TD3 with multiple objectives, including a human demonstration distilled into a reward function using D-REX. Lastly, Marta et al. (2023) adopt a multi-objective approach to balance an expert-designed objective with a human preference objective derived from a reward model trained on pairwise human trajectory comparisons.

Overall, human position-based navigation utilizes explicit knowledge of human positions and velocities to enable safer and more socially-aware navigation policies. Techniques such as CADRL-based methods establish foundational frameworks by learning interaction-aware value functions. Moreover, incorporating human preferences and demonstrations ensures policies align closely with human expectations.

2.3 Human attention-based navigation

Human attention-based navigation approaches explicitly model the attention between humans within a crowd. Human

Attention-based approaches have become a key component in social navigation, enabling policies that adapt to both individual and crowd dynamics, and achieving significant performance improvement (see Table 3). These methods explicitly model relationships between human features using pooling layers or graph neural networks (GNNs) to represent mutual influences. Pooling layers provide a compact, unified representation of human features, which, when combined with individual features, encodes human-human attention. In graph-based approaches, the robot and humans are nodes in the input graph, generating node embeddings that capture human-human and robot-human relationships.

SARL (Chen et al., 2019b) builds on CADRL (Chen et al., 2017b) by introducing an attention and a pooling module to explicitly capture human-human attention. The attention module encodes features of each human relative to surrounding humans using a human-centered local map. In this local map, each human's surrounding individuals are divided into grid cells concatenated with the human and robot states, then the features are passed into an MLP to produce a human embedding vector. To capture human-human attention and transform an arbitrary number of human embeddings into a fixed-size vector, SARL uses a self-attention pooling module, an attention mechanism adapted from Transformers. This attention mechanism assigns scalar weights to each human embedding vector and computes a unified output by summing the weighted embeddings across all humans. This dual-stage position-based encoding via the local map and self-attention pooling improves social navigation performance compared to methods without explicit attention encoding, though local maps offered a slight performance improvement during testing. During deployment, SARL may also be adapted to use a single-step human trajectory prediction model to estimate the next state, offering a more accurate alternative to the constant velocity model used in CADRL.

SOADRL (Liu et al., 2020a) extends SARL to a model-free RL setup, introducing a two-policy switching mechanism to address both dynamic and static obstacles. When humans are present, SOADRL combines SARL's output with a robot-centric angular map or 2D occupancy grid for static obstacle encoding. In the absence of humans, SOADRL switches to a policy that relies solely on the map input, ensuring efficient navigation through static obstacles.

NaviGAN (Tsai and Oh, 2020) introduces a learning-based social force model (SFM) for navigation using a dual LSTM-based GAN architecture. The model's first LSTM generates an intention force based on the robot's goal and past state sequence, while the second LSTM generates a social force that accounts for human interactions. It uses a pooling layer similar to the one in Social-GAN (Gupta et al., 2018) to encode human history. It also incorporates a fluctuation force for randomness. The combined intention and social forces determine the robot's future actions. A discriminator is used during training to encourage realistic behavior, distinguishing between generated actions and expert actions from a real-world pedestrian dataset. To incorporate temporal information, DS-RNN (Liu et al., 2021) uses a three-RNN architecture trained with PPO for social navigation. One RNN encodes each human's past positions relative to the robot; another encodes the robot's past velocities. These embeddings are combined via attention pooling (without modeling human-human attentions) and, along with the robot's state, fed into a third RNN that outputs the policy action and value function.

TABLE 3 Human-human interaction-based social navigation algorithms.

Method	Input	Architecture	Algorithm	Output	Training sim	Real-world demo	Code	Detection/Tracking
SARL (Chen et al., 2019b)	Human pos. and vel	Self-Attn + LSTM + FC	Deep V-learning	80 discrete (v, θ)	CrowdNav	Demo + Video	Yes	Depth projection detection
Liu et al. (2020a)	2D LiDAR grid/map + Human pos. and vel	Self-Attn + LSTM + 2D CNN + FC	A3C	Discrete (v, ω)	Custom Sim	Demo + Video	No	MobileNetSSD + Block matching detection (Eppenger et al., 2020)
NaviGAN (Tsai and Oh, 2020)	Human path	GAN + LSTM + Pooling + FC	GAN	$(p_x, p_y)_1, \dots$	None	Demo + Video	No	LiDAR + Kalman filter
DS-RNN (Liu et al., 2021)	Human position	S-RNN (LSTM/GRU) + FC	PPO	(v_x, v_y)	CrowdNav	Demo + Video	Yes	YOLOv3 detection (Redmon, 2018) + DeepSORT tracking (Wojke et al., 2017)
GazeNav (Chen et al., 2020a)	Human pos. and vel	GCN + FC	Deep V-learning	Sampled (v, θ)	Custom Sim	None	No	None
Navistar (Wang et al., 2023a)	Human pos. and vel	Multi-Head Attn + GCN + FC	SAC	(v_x, v_y)	CrowdNav	Lab experiment + Video	Yes	Velocity predictor (YOLO + DeepSORT) (Pramanik et al., 2021) + Distance estimator (Bertoni et al., 2021)
Liu et al. (2023a)	Human occupancy grid	GAT + 2D CNN + LSTM + FC	DQN	5 discrete (v_x, v_y)	Custom Sim	Demo + Video	No	Yolov3 detection (Redmon, 2018) + pointcloud tracking (Liu et al., 2020b)

2.3.1 Graph neural network-based navigation

GazeNav (Chen et al., 2020a) employs a model-based RL approach with gaze-based attention that uses 2 two-layer Graph Convolutional Networks (GCNs) to define its value function. The first GCN, an attention network, treats the robot and humans as graph nodes with uniform edge weights, predicting attention weights for each connection. The second GCN is an aggregation network that uses the predicted attention weights as edge values to compute embedding vectors for each human-robot pair, which are then passed into an MLP-based value function. To train the attention network, GazeNav introduces three supervised methods: uniform weights, distance-based weights, and gaze-modulated weights. The gaze-modulated weights are obtained by tracking human gaze in a simulated environment, assigning higher attention to humans within the gaze direction. Experiments show that gaze-modulated weights outperform uniform, distance-based, and self-attention-based weights (Chen C. et al., 2019), demonstrating the

benefits of incorporating human gaze data. For a more expressive representation, Navistar (Wang W. et al., 2023) uses a three-block architecture to model spatio-temporal crowd interactions. A spatial block (GCN plus multi-head attention) creates spatial embeddings; a temporal block applies multi-head attention with positional encoding for each human. A multi-modal transformer block then merges these outputs using cross-attention and self-attention to produce the final action and value outputs. In a related approach, Liu Z. et al. (2023) integrate GNNs with occupancy grids to capture spatial-temporal characteristics. At each time step, the environment is divided into a robot-centered grid and an obstacle-centered grid for each human, both processed by a CNN. The CNN outputs are then passed through an LSTM to capture temporal patterns, feeding into a Graph Attention Network (GAT) that produces interaction-aware embeddings. The control policy uses an MLP to generate action distributions from the GAT's aggregated output.

TABLE 4 Human prediction-based social navigation algorithms.

Method	Input	Architecture	Algorithm	Output	Training sim	Real-world demo	Code	Detection/Tracking	Prediction
MCTS-RNN (Eiffert et al., 2020a)	Human path	LSTM Enc-Dec + FC	MCTS (Model-based)	Path $(p_x, p_y)_1, \dots$	CrowdNav	Demo + Video	Yes	None	Learned RNN
MP-RGL (Chen et al., 2020b)	Human pos. and vel.	GCN + FC	MCTS (Model-based)	Discrete paths $(v_x, v_y)_1, \dots$	CrowdNav	Demo + Video	Yes	YOLO detection + EKF tracking	Learned GCN
GO-MPC (Brito et al., 2021)	Human pos. and vel.	LSTM + FC	PPO + MPC	(v, ω)	CADRL Sim	None	Yes	None	Constant velocity model
Poddar et al. (2023)	Human path	LSTM + GAN + FC	MPC (Model-based)	Discrete (v_x, v_y)	CrowdNav	Lab experiment + Video	Yes	None	S-GAN (Gupta et al., 2018)
SARL-SGAN-KCE (Li et al., 2020)	Human pos. and vel.	GAN + LSTM + Self-Attn	Deep V-Learning	(v, ω)	CrowdNav	None	No	None	S-GAN (Gupta et al., 2018)
Liu et al. (2023b)	Human path	Multi-Head-Attn + GRU + FC	PPO	(v_x, v_y)	CrowdNav	Demo + Video	Yes	DR-SPAAM (Jia et al., 2020)	GST (Huang et al., 2021)

To summarize, human attention-based navigation methods explicitly model human-human and human-robot attentions to enable socially-aware and adaptive policies. Approaches utilizing pooling layers, GNNs and RNNs, provide improved social compliance by capturing spatial and temporal relationships.

2.4 Human prediction-based navigation

Human Prediction-based Social Navigation (see Table 4) leverages human trajectory prediction to enable more strategic, optimal navigation in dynamic environments (see Section 3.3.2). This approach aligns with model-based RL principles, where the human prediction model serves as a dynamics model, guiding decision-making by simulating future states. To leverage this predictive capability, the navigation system should plan over a similar multi-second horizon rather than just single-step actions. Early work in this area applied techniques like Monte Carlo Tree Search (MCTS) for high-level decision-making in autonomous vehicles (Paxton et al., 2017) and optimization-based planners such as MPC for robots (Finn and Levine, 2017). One notable example is Chen et al. (2018), who use a Social-LSTM (Alahi et al., 2016) to predict human trajectories, incorporating this into an optimization-based timed elastic band (TEB) planner (Rösmann et al., 2015) with adaptive travel modes that adjust based on crowd density and movement direction.

2.4.1 MCTS-based navigation

MCTS-RNN (Eiffert et al., 2020a) is a model-based RL navigation system that uses an LSTM encoder-decoder human prediction model as its dynamics model. The LSTM model is trained on pedestrian datasets and outputs a Gaussian distribution over future human states. Planning is conducted using MCTS with a receding horizon, performing single-step rollouts from each node to reduce runtime, which increases state uncertainty. To handle this, the reward function includes both goal proximity and prediction uncertainty. MP-RGL (Chen et al., 2020) integrates MCTS planning with a GCN-based human prediction model. The GCN operates on a fully connected graph comprising humans and the robot, where edge weights are computed using Gaussian similarity in the node embedding space (Wang X. et al., 2018). Planning is performed through a simplified MCTS (Oh et al., 2017), with a d -step planning horizon, leveraging value function estimates instead of explicit rollouts.

2.4.2 MPC-based navigation

GO-MPC (Brito et al., 2021) is a hybrid framework that integrates RL and nonlinear MPC for navigation, where an LSTM-based RL model proposes sub-goals (as Gaussians) and the MPC computes optimal, collision-free trajectories to these sub-goals. The RL model is first supervised-trained with MPC-generated labels, then fine-tuned with PPO, aiming to maximize goal-reaching and minimize collisions. The MPC minimizes distance and control costs, enforcing constraints to avoid predicted human paths. Poddar et al. (2023) propose a hybrid approach that integrates

a Social-GAN (Gupta et al., 2018) human prediction model with an MPC planner. This approach uses discrete MPC to optimize a cost function that balances goal distance, social distance, and alignment with Social-GAN predictions to encourage human-like behavior. While Social-GAN can generate multiple predictions per human, results indicate that single and multiple prediction scenarios perform comparably to simpler constant-velocity estimates.

SARL-SGAN-KCE (Li et al., 2020) combines Social-GAN predictions with the SARL model (Chen C. et al., 2019) to choose optimal single-step actions. To ensure smooth motion, the planner constrains the action space by limiting angular velocity and penalizing rapid acceleration changes. Experimental results show that a higher number of trajectory predictions per human achieves performance comparable to a lower number of predictions. Finally, Liu S. et al. (2023) propose a model-free PPO RL approach that incorporates off-the-shelf human prediction models like GST (Huang et al., 2021). Human predictions are processed with multi-head human-human attention, then through robot-human attention with the robot's state, followed by a GRU that outputs the value and action. The reward penalizes intersecting predicted human paths, reducing collision risk despite prediction uncertainty.

In summary, human prediction-based navigation enhances decision-making by anticipating future human movements, enabling more strategic and socially compliant planning. Challenges include managing uncertainty from the robot's impact on human behavior and the computational cost of tree-based methods like MCTS, which require repeated action sampling and forward simulation.

2.5 Safety-aware navigation

Considering that learning-based approaches are, in some sense, viewed as black-box methods, researchers have attempted to embed safety and functionality through purposefully designed algorithms (see Table 5). These approaches are classified as safety-aware when they introduce an additional module, training strategy, or feature primarily dedicated to safety.

2.5.1 Multi-policy navigation

Hybrid multi-policy planning combines multiple strategies, where robots switch policies based on context and uncertainty. For example, Sun et al. (2019) switches between RL and RVO when a collision is imminent. Katyal et al. (2020) build on this with risk-averse and aggressive policies. By default, the system follows the aggressive policy but switches to the risk-averse policy in novel social scenarios, identified by an LSTM-based probabilistic pedestrian prediction module that uses goal intent prediction to generate a set of possible trajectories. The policy selector computes uncertainty from these predictions, with higher uncertainty indicating unfamiliar situations where the risk-averse policy is preferred. Extending this approach, Fan et al. (2020) develop a three-policy system with a scenario classifier to switch between a PID controller, a standard RL policy (Long et al., 2018), and a safe RL policy with clipped velocity. The classifier relies on two parameters, the *safe radius* and *risk radius*, based on the distance to nearby obstacles. When within the safe radius, the PID policy is used. In the risk radius, the RL policy takes over, and outside both, the safe policy is employed. To address more complex scenarios, Amano and Kato (Amano and Kato, 2022) add a fourth policy to this

setup, a reset policy to move the robot toward a larger unoccupied space if it detects a freezing robot scenario. This extension ensures the robot can navigate out of potentially freezing situations. Furthermore, Linh et al. (2022) propose a multi-policy system with three policies, using an RL-based policy selector to choose the most appropriate policy dynamically. Policies include both learning-based (RL) and model-based (TEB) planners (Rösmann et al., 2015). The selector is trained to optimize rewards by picking the best policy for a given context, combining flexibility with performance for complex navigation tasks.

Nishimura and Yonetani (2020) introduce Learning-to-Balance (L2B), a single-policy RL system that dynamically switches between two behaviors: passive crowd avoidance or active path-clearing through audible signals. The robot action is defined by a velocity vector and a binary mode indicator, with a reward function that discourages excessive path-clearing while promoting social distancing. To simulate the impact of path-clearing sounds on human behavior during training, L2B uses a simplified version of emotional reciprocal velocity obstacles (ERVO) (Xu M. et al., 2019), which accounts for emotional reactions to perceived threats. IAN (Dugas et al., 2020) is a multi-policy navigation system that uses Monte Carlo Tree Search (MCTS) to choose among three planning policies: intend (RVO planner (Alonso-Mora et al., 2013) for reactive avoidance), say (verbal path announcement with lower speed and assumed human cooperation), and nudge (DWA planner (Fox et al., 1997) for cautious progress). MCTS evaluates paths by crowdedness, perceptivity, and permissivity, selecting the lowest-cost route and adapting plans based on each policy's success probability. Both L2B and IAN require the robot to have a speaker and operate where its audio signals are audible.

Lütjens et al. (2019) propose a hybrid safe RL system based on discrete MPC, optimizing a cost function that accounts for estimated goal-reaching time and predicted collision probability. An ensemble of LSTMs predicts collision probabilities of motion primitives, with MC-dropout (Gal and Ghahramani, 2016) used for uncertainty estimation. The collision prediction model is trained as a binary classifier in simulation, penalizing uncertainty to encourage safe exploration. However, this approach heavily depends on collision model accuracy, and inaccuracies can lead to overly conservative behavior. Sathiamoorthy et al. (2020b) introduce Frozone, which prevents robot freezing by detecting potential freezing zones (PFZs) using pedestrian positions and velocities. A convex hull is constructed around predicted pedestrian locations, and the robot computes a deviation angle to avoid these regions. However, in confined spaces like corridors, Frozone may lead the robot toward other obstacles. XAI-N (Roth et al., 2021) leverages decision trees to create an interpretable navigation policy. XAI-N distills an RL policy (Fan et al., 2018) into a single decision tree using the VIPER method (Bastani et al., 2018), prioritizing modifiability and transparency over continuous action control. To enhance performance, the approach incorporates decision rules to address safety challenges such as freezing and oscillation, making it a more reliable option for social navigation.

Bansal et al. (2020) propose a Hamilton–Jacobi reachability-based framework that augments the human state with a belief over future intent, producing a forward reachable set that includes all likely pedestrian states for fixed time-horizon with probability above threshold λ . These sets are incorporated as time-dependent obstacles

TABLE 5 Safety-aware social navigation algorithms.

Method	Input	Architecture	Algorithm	Output	Safety mechanism	Training sim	Real-world demo	Code
Sun et al. (2019)	2D LiDAR (labeled rays)	LSTM + FC	PPO	(v, ω)	Multi-policy (VO-based switch)	Unity Custom Sim	None	No
Katyal et al. (2020)	Human path	VAE + LSTM + FC	Deep V-Learning	2D velocity	Multi-policy (uncertainty switch)	CrowdNav	None	No
Fan et al. (2020)	2D LiDAR	1D CNN + FC	PPO	(v, ω)	Multi-policy (distance switch)	Stage Sim	Lab experiment + Video	No
Linh et al. (2022)	Static and dynamic 2D LiDAR	2D CNN + FC	A3C	(v, ω)	Multi-policy (RL policy switch)	Arena-Rosnav	None	Yes
Nishimura and Yonetani (2020)	Human pos. and vel.	Self-Attn + FC	Deep V-Learning	8 discrete (v_x, v_y) + Clearing beep	Crowd clearing beep	Custom Sim	None	Yes
IAN (Dugas et al., 2020)	Grid map	None	MCTS	(v, ω)	Multi-policy (MCTS switch)	Custom Sim	Demo + Video	Yes
Lütjens et al. (2019)	Human pos. and vel.	LSTM + FC	MPC	11 discrete (v, θ)	Uncertainty collision prediction	Custom Sim	None	No
Sathyamoorthy et al. (2020b)	2D LiDAR + human pos	1D CNN + FC	PPO	(v, ω)	Heading Safety filter	Gazebo Sim	Demo + Video	No
XAI-N (Roth et al., 2021)	2D LiDAR	Decision Tree	PPO + VIPER	Discrete (v, ω)	Decision Tree	Gazebo Sim	Demo + Video	Yes
Jang and Ghaffari (2024)	Human pos. and vel.	None	MPC	(v_x, v_y)	Control-Barrier Function	2D sim	None	No
CASRL (Zhou et al., 2023)	Human pos. and vel.	HGAT + MLP	TD3	(v_x, v_y)	Multi-task	CrowdNav	Demo	No
Pfeiffer et al. (2018)	2D LiDAR	FC	CPO	(v, ω)	Collision avoidance constraint	Custom Sim	Demo	Yes
SoNIC (Yao et al., 2024)	Human pos. and vel.	GRU + MLP	PPO-Lag	(v_x, v_y)	Constraint	CrowdNav	Demo + Video	Yes

and avoided with a spline-based trajectory planner. However, in highly populated scenes, the predicted reachable sets may overlap heavily for a low threshold, effectively blocking all routes and causing the robot to freeze, and while the human model parameters can be learned from data, the overall prediction approach is model-based. [Jang and Ghaffari \(2024\)](#) learn social zones from pedestrian data by relating distance to line-of-sight angle, then approximate them with speed-dependent ellipses enforced through a control-barrier function within a hybrid MPC planner. The

learned zones are front-biased—indicating humans prefer more space ahead—and slightly tilted, reflecting overtaking behavior. While enforcing them as hard constraints improves safety, it can be overly conservative, since people typically tolerate reduced spacing in crowded settings. CASRL ([Zhou et al., 2023](#)) frames safety in navigation as a multi-task RL problem: goal reaching and collision avoidance tasks. It extends an off-policy RL algorithm (TD3) with separate critics for each task, while the actor is updated using a conflict-averse rule that maximizes the minimum improvement

TABLE 6 Crowd simulation libraries.

Library	Crowd behavior	Language
RVO2 (Van Den Berg et al., 2025)	ORCA	Python (Stüvel, 2025) or C++
UMANS (van Toll et al., 2020)	SFM, PLEdetrans, RVO ORCA, PowerLaw, Vision	C++
PySocialForce (Gao, 2025)	SFM with groups (Moussaïd et al., 2010)	Python
DeepSocialForce (Kreiss, 2021)	Deep Social Force	Python
CROMOSIM (Faure, 2025)	SFM, CA, Granular	Python
CrowdDynamics (Group, 2025)	SFM with Path planning	Python
JuPedSim (Dynamics, 2025)	SFM, Centrifugal, CFSM	Python & C++
Mesa (Masad, and Kazil, 2015)	ABM	Python
Agents.jl (Datseris et al., 2024)	ABM	Julia
Vadere (Kleinmeier et al., 2019)	CA, SFM, OSM (Seitz and Köster, 2012)	Java

across tasks. This reduces performance loss when gradient updates disagree. However, the reported simulation gains fall short and may require further tuning.

2.5.2 Constrained RL

Constrained RL provides a natural framework for enforcing safety, as constraints take precedence over the reward objective when violated. For instance, Pfeiffer et al. (2018) introduce a safe RL navigation policy that defines a collision constraint, trained using constrained policy optimization (CPO) (Achiam et al., 2017), which maximizes reward while constraining the expected number of collisions. SoNIC (Yao et al., 2024) introduces a safety constraint derived from Adaptive Conformal Inference (ACI), which quantifies the uncertainty of predicted pedestrian trajectories. Similarly, Zhu et al. (2025) propose a confidence-weighted trajectory prediction model, where a Bayesian β parameter adapts the uncertainty region based on prediction errors. In their method, uncertainty is incorporated through a robust dynamical distance constraint that estimates time-to-collision, rather than relying on simple distance-based thresholds. However, both approaches employ trajectory predictors that neglect the robot's presence, resulting in predictions where pedestrians are assumed to move independently of the robot. This leads to overly conservative robot behavior.

In conclusion, safety-aware navigation improves reliability in learning-based systems through structured mechanisms, but further work is needed to balance safety with efficiency and ensure adaptability to diverse real-world scenarios.

3 Navigation model training

Training social navigation policies equips robots with safe, efficient, and socially aware navigation in human environments. This section outlines key training components (see Figure 2),

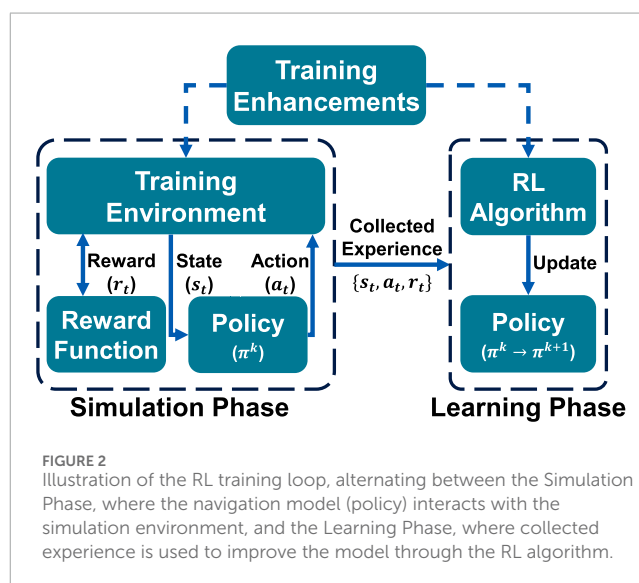


FIGURE 2 Illustration of the RL training loop, alternating between the Simulation Phase, where the navigation model (policy) interacts with the simulation environment, and the Learning Phase, where collected experience is used to improve the model through the RL algorithm.

including the objective function, environments with static and dynamic obstacles, including realistic crowd simulation. Advanced strategies, such as pre-training, enhance training efficiency. We also examine human detection, tracking, prediction, and broader scene understanding and activity recognition, which are leveraged by navigation policies to improve performance. Finally, we cover evaluation methods for social navigation, including metrics and real-world experiments.

3.1 Objective function

The objective or reward function in most reinforcement learning (RL) problems is typically formulated as $\max \mathbb{E}[\sum_{t=0}^h \gamma^t r_t]$, where the goal is to maximize the expected cumulative reward. This

formulation typically incorporates a discount factor $\gamma \in [0, 1]$ to prioritize earlier rewards over delayed ones. The reward function, denoted by $r(s, a)$, is often defined as a function of the current state and action but can also incorporate the subsequent state $r(s, a, s')$, to capture transition dynamics. In navigation tasks, the objective is often a weighted sum of reward components:

$$r(s, a, s') = \sum_i w_i r_i(s, a, s')$$

where $w_i \in \mathbb{R}^+$ are weights and r are scalar and indicator functions. The section details possible reward components, but it's worth noting that combining multiple objectives can create local minima (Everett et al., 2021).

Many reward functions are *sparse*, providing feedback only at key milestones like reaching a goal. To improve learning, *reward shaping* introduces dense rewards, giving intermediate feedback at each timestep. While dense rewards speed up learning, they must be carefully designed to avoid suboptimal strategies.

3.1.1 Goal reward

The reward function for reaching a goal state is the main component of any navigation task. It is often defined as an indicator function $[\|p^r - g^r\| \leq g_{tol}]$, which returns 1 when the Euclidean distance between the robot's current position p and the goal position g^r is within a specified tolerance g_{tol} . A dense reward formulation for goal-reaching provides feedback based on progress, calculated as $\|p_t^r - g\| - \|p_{t-1}^r - g\|$, reflecting incremental advancement toward the goal (Long et al., 2018; Tan et al., 2020). Other approaches reward the agent for moving along grid cells that align with the global path (Liu Z. et al., 2023).

3.1.2 Collision-avoidance reward

The reward function for collision avoidance is often defined as an indicator function, $-[p^r \in \mathcal{C}]$, which returns 1 when the robot's position p^r is within a collision set \mathcal{C} , or when a collision is detected using other means. Alternative definitions penalize proximity to obstacles, such as $-[d_{obs} | d_{obs} \leq d_{tol}]$ (Chen et al., 2017b) or $[\frac{d_{obs}}{d_{tol}} - 1 | d_{obs} \leq d_{tol}]$ (Cui et al., 2021) where d_{obs} is the minimum distance to nearby obstacles and d_{tol} is the tolerance distance. These formulations may apply different thresholds for static and dynamic obstacles, such as humans (Cui et al., 2021). Other approaches use step functions to gradually increase the penalty as the robot gets closer to obstacles, encouraging safer navigation.

3.1.3 Efficiency reward

To encourage efficient and timely navigation, reward functions often include terms that promote higher speeds. This may take the form of a gradual step function that provides a higher reward for increased velocity (Lee and Jeong, 2023) or a negative step-cost applied at each timestep to minimize time taken to reach the goal (Wang Y. et al., 2018; Choi et al., 2019).

3.1.4 Smoothness reward

For smooth trajectory generation, a negative reward proportional to the rotational velocity, $-|\omega|$ is applied for differential drive robots (Tan et al., 2020; Xie and Dames, 2023). Additionally, Hoeller et al. (2021) penalizes lateral and backward velocities by adding a negative reward proportional to their

squared magnitudes, encouraging smoother and more consistent forward motion.

3.1.5 Social reward

Social norms can be integrated into the reward function to promote behaviors like passing, crossing, and overtaking in socially appropriate ways (Chen et al., 2017c). This reward function is typically defined as a conditional function based on human parameters relative to the robot, including x-axis position, velocity, distance, relative heading angle, and heading angle difference. For instance, to promote overtaking from the left, the robot is rewarded when certain conditions are met: the goal distance exceeds 3, the human is positioned within $0 < p_x < 3$, and $0 < p_y < 1$, the robot's velocity surpasses the human's, and their heading angle difference is under $\pi/4$.

3.1.6 Geometric collision-avoidance reward

Model-based or geometric rewards using human position have enabled more robust navigation. For instance, DRL-VO (Xie and Dames, 2023) uses velocity obstacles (VOs) to model human motion, rewarding alignment with the optimal heading direction, where VOs are computed during training only. Han R. et al. (2022) incorporate VOs into both state and reward, with rewards based on joint VO area, velocity differences, and estimated minimum time to collision. Zhu et al. (2022) employ an oriented bounding capsule (OBC) model, where human velocity adds a buffer in front of the OBC, and the reward is the minimum distance to the OBC; OBC parameters are included in the robot's state for better learning. Additionally, Samsani and Muhammad (2021) define a danger zone (DZ) as an extended sector around humans, accounting for uncertainty in position and velocity predictions.

3.1.7 Human preference reward

Reinforcement Learning from Human Feedback (RLHF) provides a framework to simultaneously learn a policy and a reward function using human input (Christiano et al., 2017), with applications spanning various domains, including language models like GPT-3 (Ouyang et al., 2022). In social navigation, Wang R. et al. (2022) applies RLHF to learn a reward function based on pairwise human preferences over trajectory segments.

3.1.8 Human prediction reward

For planners that utilize human trajectory predictions, a negative reward is often used to discourage the robot from intruding into human-predicted zones (Liu S. et al., 2023). Additionally, a negative reward can be defined over prediction uncertainty, as in (Eiffert et al., 2020b), where the reward is the negative square root of the determinant of the covariance matrix, $\sqrt{\det(\Sigma)}$, encouraging actions that lead to more predictable crowd behavior.

3.1.9 Exploration reward

Exploration rewards are designed to encourage the robot to explore a wide range of actions or states. Action-based exploration rewards promote action diversity by maximizing the policy's entropy (Schulman et al., 2017), while state-based exploration rewards encourage the robot to explore new areas. For instance, the intrinsic curiosity module (ICM) (Pathak et al., 2017), applied to navigation tasks (Shi H. et al., 2019; Martinez-Baselga et al., 2023)

to reward the robot for discovering novel states, thereby enhancing its learning process.

3.1.10 Task-specific reward

Task-specific rewards are custom-designed to achieve the requirements of a particular navigation task. For example, in social navigation with a human companion, Li et al. (2018) define a distance-based reward that penalizes the robot for straying from its companion, encouraging it to stay close and coordinate its movement with the human partner.

3.1.11 Learning rewards from demonstrations

Inverse reinforcement learning (IRL) infers a reward function from expert demonstrations, either by using handcrafted state-action features (Okal and Arras, 2016; Kim and Pineau, 2016) or by learning feature representations directly with neural networks (Fahad et al., 2018). For instance, Vasquez et al. (2014) learn a reward function expressed as a weighted combination of features that capture local crowd density, relative velocities and orientations of nearby pedestrians, the robot's own velocity, and social force interactions. More recently, methods like disturbance-based reward extrapolation (D-REX) (Brown et al., 2020) learn reward functions from suboptimal or unlabeled data. D-REX applies behavioral cloning, adds increasing ϵ -greedy noise, and trains the reward to favor less noisy trajectories, promoting desirable behavior, an approach effective for social navigation (De Heuvel et al., 2024). Similar techniques include T-REX (Brown et al., 2019) and SSRR (Chen et al., 2021).

3.1.12 Learning reward weights

Various techniques have been developed to automatically determine the optimal values of each objective weight w_i , eliminating the need for manual tuning. One such method is inverse reinforcement learning (IRL), which infers the weights w_i by matching the robot's behavior to expert demonstrations (Ziebart et al., 2008). Additionally, AutoRL (Chiang et al., 2019; Parker-Holder et al., 2022) employs automated hyperparameter tuning to optimize the reward weights w_i during training to enhance task-specific performance metrics.

3.2 Training environment

This section reviews key components of training environments for social navigation, focusing on crowd data and physics-based simulators that replicate robot dynamics and sensory feedback to ensure realistic training conditions. Furthermore, crowd simulation libraries (see Table 6) provide controllable and realistic human behaviors that can be used to populate training environments and replicate crowd datasets.

3.2.1 Crowd data

Crowd datasets play a critical role in advancing data-driven approaches for both crowd behavior simulation and human trajectory prediction. They provide the necessary information to model realistic crowd interactions and dynamics, as detailed in the Appendix. Additionally, these datasets support the training of human prediction methods, as explored in Section 3.3.2. Table 7

organizes these datasets by their sensory platforms, including stationary sensors, moving robots, and moving vehicles, each serving distinct purposes and applications. While long-term crowd tracking datasets such as the ATC dataset (Brščić et al., 2013) exist, they lack the scale and diversity needed to support social navigation research.

3.2.2 Simulation platform

Simulators provide a controlled virtual environment for developing and evaluating social navigation algorithms by modeling human-robot interactions and crowd behaviors. Table 8 categorizes simulators based on key attributes, such as the supported sensor types, the human model ranging from simple cylindrical shapes to detailed 3D figures, supported crowd behaviors, evaluation metrics based on implementation specifics.

Many simulators share a common emphasis on creating realistic environments. For instance, Habitat (Szot et al., 2021) and Gibson (Li et al., 2021), widely used in embodied AI research, render highly detailed indoor spaces using real 3D scans. However, these environments are typically limited to smaller areas like apartments or offices, making them less suitable for large-scale crowd simulations. Additionally, several simulators, such as Isaac Sim (Makoviychuk et al., 2021), prioritize achieving high FPS, which is crucial for training performance. Simulators also vary in the complexity of crowd behaviors, with some supporting basic movement patterns and others providing sophisticated, behavior-rich models that more accurately capture crowd dynamics, such as NavRep (Dugas et al., 2021). However, current simulators remain limited, as an efficient RL-supported crowd simulation with diverse scenarios is still missing, which we aim to address with our benchmark.

3.3 Human detection, tracking, and prediction

In social navigation, a robot often relies on human detection, tracking, or prediction for better social awareness and generalizability. *Human Detection* provides the robot-centric human positions, which are essential for position-based planners (see Section 2.2). *Human Tracking* estimates human positions and velocities over time, supporting planners requiring human speeds or trajectories. *Human Prediction* utilizes tracking data to forecast future human movements, which are utilized by prediction-based planners (see Section 2.4).

3.3.1 Human detection and tracking

Human detection methods are typically tailored to specific sensors, including 2D LiDAR, 3D LiDAR, RGB, and RGB-D sensors. Tracking enhances detection by assigning unique identifiers to individuals and addressing challenges such as sensor occlusions, which is essential for reliable multi-object tracking (MOT). While human detection provides the robot-centric positions of each detected person, tracking maintains a history of these positions over time, enabling the estimation of their velocities.

TABLE 7 Pedestrian datasets.

Dataset name	Collection location	Data volume	Annotations	Sensor platform
ETH (Pellegrini et al., 2009)	ETH Zurich, and Hotel in Zurich	25 min, 2 tracks	Human position	Stationary top-view camera
UCY (Lerner et al., 2007)	University campus and Zara store	16 min, 6 tracks	Human position	Stationary top-view camera
Central Station (Zhou et al., 2012)	New York train station	33 min	Human position	Stationary top-view camera
Edinburgh (Majecka, 2009)	Outdoor University campus	92,000 trajectories	Human position	Stationary top-view camera
Stanford Drone (Robicquet et al., 2016)	Stanford University campus outdoor	5 h, 60 tracks	Bounding box	Stationary drone top-view camera
VIRAT (Oh et al., 2011)	Indoor and outdoor	25 h	Bounding box	Stationary surveillance camera
Oxford Town (Benfold and Reid, 2011)	Oxford town center	5 min	Bounding box	Stationary surveillance camera
ATC (Bršćić et al., 2013)	Shopping mall	92 days	Human position	Stationary top-mounted radar
Ko-PER (Strigel et al., 2014)	Crossing in Aschaffenburg, Germany	6.5 min	Human position	Stationary top-view camera and Laser
inD (Bock et al., 2020)	Crossing in Aachen, Germany	10 h	Human position	Stationary drone top-view camera
CITR and DUT (Yang et al., 2019)	University campus	13 min, 66 tracks	Human position	Stationary drone top-view camera
WILDTRACK (Chavdarova et al., 2018)	ETH Zurich, Switzerland	200 s	Human position	Stationary tilted camera
STCCrowd (Cong et al., 2022)	Outdoor streets and walkways	84 tracks	Human position	Stationary vehicle with RGB and 3D LiDAR
THOR Dataset (Rudenko et al., 2020b)	Indoor large room	60 min	Human position	Stationary/moving robot with 3D LiDAR and tracking
L-CAS (Yan et al., 2017)	Indoor university building	49 min	Human position	Stationary and moving robot with 3D LiDAR
SCAND (Karnan et al., 2022)	Indoor and outdoor	8.7 h	Social interaction	Robot with RGB-D, and 3D LiDAR
JRDB (Martin-Martin et al., 2021)	Indoor and outdoor	64 min	Human position	Robot with RGB-D, and 3D LiDAR
FLOBOT (Yan et al., 2020)	Airport and supermarket in Italy/France	27 min	Human position	Autonomous robot with RGB-D, and 3D LiDAR
NCLT (Carlevaris-Bianco et al., 2016)	University campus indoor and outdoor	35 h	None	Robot with RGB-D, and 3D LiDAR
MuSoHu (Nguyen et al., 2023)	Indoor and outdoor	20 h	Social interaction	Walking human + helmet with RGB-D and 3D LiDAR
CrowdBot (Paez-Granados et al., 2021)	3 streets in Lausanne, Switzerland	200 min	Human position	Semi/Autonomous Robot with RGB-D and 3D LiDAR
HuRoN (Hirose et al., 2023)	5 office environments	75 h	Human position	Autonomous robot with RGB-D and 2D LiDAR
SiT (Bae et al., 2024)	60 scenes, indoor and outdoor, Seoul	470,000 frames	Human position	Robot with RGB-D, and 3D LiDAR
KITTI (Geiger et al., 2012)	Karlsruhe, Germany	6 h	Human position	Moving vehicle with RGB, 3D LiDAR and more
nuScenes (Caesar et al., 2020)	Boston, MA and Singapore	5.5 h, 20s tracks	Human position	Moving vehicle with RGB, 3D LiDAR and more
Waymo (Ettinger et al., 2021)	San Francisco, Phoenix, and more	570 h, 20s tracks	Human position	Moving vehicle with RGB, 3D LiDAR and more
BDD100K (Yu et al., 2020)	New York, Berkeley, and more	1100 h, 40s tracks	Human position	Moving vehicle with RGB, 3D LiDAR and more
A2D2 (Geyer et al., 2020)	South of Germany	40,000 frames	Human position	Moving vehicle with RGB, 3D LiDAR and more

TABLE 8 Simulation platforms.

Simulator	Description	Human behavior	Sensors	Human model	Maps	Metrics
Menge (Curtis et al., 2016)	Supports ROS (Aroor et al., 2017)	Path planning + ORCA and SFM	None	Circle	2D Maps	No
SEAN (Tsoi et al., 2022; Tsoi et al., 2020)	Based on Unity game engine	SFM and ORCA	RGB-D	3D humanoid	3D maps	Navigation and Social
SEAN-EP (Tsoi et al., 2021)	Build on SEAN simulator For human feedback collection	SFM and ORCA	RGB-D	3D humanoid	3D maps	Navigation and Social
UnrealCV (Qiu et al., 2017)	Build on Unreal Engine 4	None	RGB	3D humanoid	3D maps	No
MORSE (Echeverria et al., 2011)	Based on Blender engine	None	RGB-D and 3D LiDAR	3D humanoid	3D maps	No
Gazebo (Koenig and Howard, 2004)	ROS compatible simulator	None	RGB-D and 3D LiDAR	3D humanoid	3D maps	No
Isaac Sim (Makoviychuk et al., 2021; Mittal et al., 2023)	Based on PhysX engine	None	RGB-D and 3D LiDAR	3D Rigid human (Isaac GYM)	None	No
Webots Sim (Michel, 2004)	Based on OpenGL engine	None	RGB-D and 3D LiDAR	3D humanoid	3D maps	No
A12-THOR (Kolve et al., 2017)	Based on Unity Engine Only supports LoCoBot	None	RGB-D	None	3D maps	No
Habitat 2.0 (Szot et al., 2021)	Based Habitat Sim	None	RGB-D	None	3D Scans	Navigation
Habitat 3.0 (Puig et al., 2023)	Based Habitat Sim	Path Planning	RGB-D	3D humanoid	3D Scans	Navigation and Social
HabiCrowd (Vuong et al., 2023)	Based Habitat Sim	UPL (Karamouzas et al., 2014)	RGB-D	3D humanoid	3D Scans	Navigation and Social
SAPIEN (Xiang et al., 2020)	Based on PhysX engine	None	RGB-D	None	3D Scans	No
CrowdBot Sim (Grzeskowiak et al., 2021)	Based on Unity Engine	UMANS (van Toll et al., 2020)	RGB-D and 3D LiDAR	3D humanoid	3D maps	Navigation and Social
iGibson (Li et al., 2021)	Based on PyBullet	ORCA	RGB-D and 3D LiDAR	3D Rigid human	3D Scans	No
CrowdNav (Chen et al., 2019b)	2D openspace simulator	SFM and ORCA	None	Circle	None	No
Arena-Rosnav (Kästner et al., 2021)	Uses Gazebo, Unity and Flatland	SFM and ORCA	RGB-D and 3D LiDAR	Circle or 3D humanoid	2D and 3D maps	Navigation and Social
nav-gym (Lee et al., 2023)	Use CrowdNav with Flatland	ORCA	2D LiDAR	Circle or 3D humanoid	2D maps	No

(Continued on the following page)

TABLE 8 (Continued) Simulation platforms.

Simulator	Description	Human behavior	Sensors	Human model	Maps	Metrics
gym-collision-avoidance (Everett et al., 2018)	Supports multiple robots and humans	ORCA	None	Circle	None	No
navrep (Dugas et al., 2021)	Multiple social simulators	Global planner + ORCA	2D LiDAR	Circle or human with moving legs	2D maps	No
gym ped sim (Tai et al., 2018)	A plugin for Gazebo	SFM	RGB-D and 3D LiDAR	3D humanoid	3D maps	No
Social Gym (Sprague et al., 2023)	A multi-agent 2D simulator	RL planner	None	Circle	2D maps	Navigation and Social
RDS Sim (Gonon et al., 2021)	RDS planner evaluation code	ORCA	None	Circle	None	Navigation and Social

3.3.1.1 Human detection

RGB-based human detection leverages general object detection techniques, which can be broadly categorized into classical and deep learning approaches. Classical methods such as histogram of oriented gradients (HOG) (Dalal and Triggs, 2005) and deformable part model (DPM) (Felzenszwalb et al., 2008), often struggle with accuracy and robustness in complex or dynamic environments. Deep learning-based methods, on the other hand, are divided into one-stage and two-stage approaches. Two-stage or *coarse-to-fine* methods, like Faster R-CNN (Ren et al., 2016) and FPN (Lin et al., 2017), typically offer higher accuracy by refining proposals. While one-stage detectors, such as YOLO (Redmon, 2016), SSD (Liu et al., 2016), and DETR (Carion et al., 2020), prioritize speed, making them ideal for real-time applications in social navigation. For further details, refer to Zou et al. (2023). The output of RGB-based object detection provides a bounding box in the image plane, which requires conversion to robot-centered coordinates for accurate spatial positioning. To estimate 3D pose parameters from 2D detections, some methods, such as Multi-fusion (Xu and Chen, 2018) and ROI-10D (Manhardt et al., 2019), incorporate depth estimation modules to approximate distance. Meanwhile, techniques like Deep3DBox (Mousavian et al., 2017), MonoGRnet (Qin et al., 2019), and Hu et al. (2019) apply geometric reasoning techniques for 3D localization based on 2D information.

Early methods for 2D LiDAR-based human detection relied on hand-crafted features, identifying humans by detecting both legs within a segment (Arras et al., 2007) or by tracking individual legs over time (Leigh et al., 2015). The first deep learning-based detector, DROW (Beyer et al., 2016), was subsequently enhanced by incorporating temporal information to improve tracking consistency (Beyer et al., 2018). Building upon DROW, DR-SPAAM (Jia et al., 2020) introduced faster processing capabilities for handling long-term temporal data. Additionally, Dequaire et al. (2018) employed an occupancy grid-based approach combined with an RNN to capture temporal patterns effectively. Current 3D LiDAR detection approaches are categorized into Bird's Eye View (BEV) methods, point-based methods, voxel-based methods, multi-view methods, and range-view-based methods. BEV methods provide

fast, top-down 2D projections of the environment, making them popular for quick processing tasks in robotics. Examples include PIXOR (Yang et al., 2018a) and HDNet (Yang et al., 2018b). However, they often miss critical vertical details essential for detecting objects like pedestrians. *Point-based* methods directly process raw point cloud data, offering higher accuracy. Notable examples are PointNet++ (Qi et al., 2017) and PointRCNN (Shi S. et al., 2019). However, these methods are computationally intensive and less suitable for real-time applications. *Voxel-based* methods transform point clouds into 3D voxel grids, effectively balancing accuracy and computational efficiency by reducing processing loads while preserving essential details. Notable examples include VoxelNet (Zhou and Tuzel, 2018) and SECOND (Yan et al., 2018). *Multi-view* methods, such as MV3D (Chen X. et al., 2017) and SE-SSD (Zheng et al., 2021), combine multiple point cloud representations to leverage their respective advantages and enhance detection performance. *Range-view-based* methods convert LiDAR data into 2D range images, preserving vertical details and achieving high processing speeds, making them well-suited for applications like social navigation. Approaches include RangeNet++ (Milioto et al., 2019) and RSN (Sun et al., 2021). RGB-D-based human detection combines RGB data with depth information, which can also be acquired from a 3D LiDAR for sensor fusion. Techniques like PointPainting (Vora et al., 2020) fuse RGB semantic data onto LiDAR points, while PointNet (Qi et al., 2018) leverage 3D bounding frustums, focusing detection within the RGB-D space. For further details, refer to Mao J. et al. (2023).

3.3.1.2 Human tracking

Human tracking involves identifying detected objects, assigning each object a unique ID, and continuously updating their location through state estimation filters, even during brief sensor occlusions. This section centers on the Tracking-by-Detection framework, which performs detection before tracking, as other tracking frameworks are less common for human tracking. Trackers vary by association metrics and tracking dimensionality. In general, MOT relies on motion prediction techniques such as Kalman filters, particle filters, or multi-hypothesis tracking (MHT) (Yoon et al.,

2018), combined with application-specific association metrics (Rakai et al., 2022). For vision-based MOT, popular methods include DEEPSort (Wojke et al., 2017) which integrates deep association metrics, ByteTrack (Zhang et al., 2022) which relies on hierarchical association for accurate initial detection and faster performance, and other methods (Xu Y. et al., 2019). For 3D MOT, approaches like AB3DMOT (Weng et al., 2020), which utilizes 3D bounding boxes and Kalman filtering, and other approaches like SimpleTrack (Pang et al., 2022) and CAMO-MOT (Wang L. et al., 2023) which enhance tracking accuracy and efficiency. Fusion-based MOT combines 2D and 3D detections from multiple sensors to enhance tracking robustness. EagerMOT (Kim et al., 2021) fuses information from multiple detectors, while DeepfusionMOT (Wang X. et al., 2022) applies deep learning-based association for enhanced consistency. For further details, refer to Peng et al. (2024).

3.3.2 Human trajectory prediction

Predicting human trajectories is critical for effective social navigation. Traditionally relying on knowledge-based methods, the field has shifted towards learning-based approaches, which consistently outperform traditional methods on metrics such as average displacement error (ADE) (Pellegrini et al., 2009) and final displacement error (FDE) (Alahi et al., 2016). Learning-based methods leverage crowd datasets (see Section 3.2.1) and typically employ CNN, LSTM, or GAN architectures (Korbmacher and Tordeux, 2022).

3.3.2.1 CNN-based predictors

CNNs, initially designed for spatial tasks, have been adapted to sequential pedestrian prediction by representing trajectories spatially. Early approaches such as Behavior-CNN (Yi et al., 2016) encode pedestrian trajectories into displacement volumes processed by CNN layers. More recent models, such as Social-STGCNN (Mohamed et al., 2020), incorporate graph convolutions to effectively model pedestrian interactions, while scene context integration further enhances predictions (Ridel et al., 2020). Overall, CNNs efficiently process data in parallel but typically require reprocessing the full input history for each prediction, limiting their efficiency in real-time navigation.

3.3.2.2 LSTM-based predictors

LSTM networks excel at capturing temporal dependencies in sequential data. Social-LSTM (Alahi et al., 2016) introduced social pooling to account for pedestrian interactions during prediction. Enhancements include integrating environmental context via semantic information (Lisotto et al., 2019) and employing attention mechanisms (Fernando et al., 2018). Graph-based methods like STGAT (Huang et al., 2019) further improve interaction modeling. Transformers have recently emerged as powerful alternatives, better capturing complex interactions and limited sensing scenarios (Huang et al., 2021). In contrast, LSTMs, despite slower batch processing, efficiently leverage hidden states for incremental, real-time predictions, making them ideal for social navigation.

3.3.2.3 GAN-based predictors

GAN-based models generate diverse and realistic trajectories, addressing human behavior's multi-modality. Influential methods

include Social-GAN (Gupta et al., 2018), which combines LSTMs with GAN frameworks, and SoPhie (Sadeghian et al., 2019), which integrates social and physical context through attention modules. Recent advancements like probabilistic crowd GAN (PCGAN) (Eiffert et al., 2020b) and diffusion-based models (Gu et al., 2022; Mao W. et al., 2023) further enhance multi-modal, safety-compliant predictions. Despite the computational demand, GANs' diverse trajectory predictions significantly contribute to robust and safe decision-making in social navigation scenarios.

3.4 Scene understanding and activity recognition

Scene understanding and activity recognition are perception modules that provide information beyond human detection and trajectory prediction. Scene understanding includes object detection, pose estimation, semantic segmentation, saliency prediction, affordance prediction, and captioning (Naseer et al., 2018).

Object detection and pose estimation, detailed in Section 3.3 for humans, can be generalized to other classes for broader scene understanding. Beyond object detection, 2D and 3D semantic segmentation assign semantic labels to pixels or points in images and LiDAR scans, producing detailed maps of the environment (Kirillov et al., 2023; Cen et al., 2023) with applications to navigation (Roth et al., 2024). Affordance prediction further interprets the scene by modeling possible interactions; for navigation, this is useful for identifying robot-traversable areas (Yuan et al., 2024). Saliency prediction models human visual attention by estimating focus regions in a scene (Lou et al., 2022), allowing vision models to ignore irrelevant input and prioritize informative areas. Finally, 3D dense captioning methods, such as Vote2Cap-DETR (Chen et al., 2023), extend scene classification or 2D captioning by generating multiple localized captions, offering richer scene descriptions for context-aware navigation.

In parallel, activity recognition interprets dynamic human behaviors at both the individual and group levels. At the individual level, this involves human action classification (Girdhar et al., 2017), while at the group level it includes group activity classification (Choi et al., 2009) often supported by group detection methods (Wang Q. et al., 2018; Li et al., 2022). More recently, LLM-based classifiers have been introduced for activity recognition (Qu et al., 2024; Liu et al., 2025). Current navigation approaches primarily use activity recognition to estimate proxemics (Charalampous et al., 2016; Narayanan et al., 2020), though its potential for richer context-aware decision-making remains unexplored.

Vision-language models (VLMs) (Liu H. et al., 2023) are large multimodal models with broad capabilities, including object recognition, reasoning, and contextual understanding. By jointly leveraging visual and textual inputs, they provide a natural bridge between scene understanding, activity recognition, and navigation guidance. Despite this potential, their use in social navigation remains limited, with only a few recent methods exploring VLM-based decision making (Song et al., 2024; Munje et al., 2025).

3.5 Training enhancement techniques

Efficient training is essential for robust social navigation policies, since large-scale RL training is often limited by computational resources. While extensive training, such as training a DD-PPO policy for 2 billion steps (Wijmans et al., 2019), can boost performance, more efficient approaches exist. Task-specific techniques, such as leveraging problem symmetries by flipping path topologies (Chen et al., 2017c) can improve exploration. This section highlights general, task-agnostic methods for enhancing training efficiency and performance.

3.5.1 Pre-training techniques

Pre-training techniques, such as behavioral cloning from demonstrations (Pfeiffer et al., 2018; Chen C. et al., 2019), accelerate training by providing basic navigation skills and reducing RL exploration. Self-supervised methods, like VAEs with reconstruction loss (Dugas et al., 2021; Hoeller et al., 2021), improve state representation, while transfer learning from pretrained CNNs enhances RGB input processing (Hong et al., 2021). Policy transfer from existing models is also used (Wijmans et al., 2019). These approaches improve training efficiency, convergence, and generalization.

3.5.2 Auxiliary tasks

Auxiliary tasks are additional tasks or objectives incorporated during training to support learning the main task. This offers better training signal and model performance. Auxiliary tasks have been shown to improve navigation performance by training models to predict features such as depth, loop closures (Mirowski et al., 2016), and location estimation (Tongloy et al., 2017). Additional tasks include predicting immediate reward prediction and learning to control specific regions in the input image (Jaderberg et al., 2016) or predicting image segmentation (Kulhánek et al., 2019). In social navigation, auxiliary tasks are used to improve understanding of social dynamics. For instance, *Proximity-Aware* (Cancelli et al., 2023) incorporates tasks to estimate the distance and direction of surrounding humans, while *Falcon* (Gong et al., 2024) incorporates tasks for predicting the number of nearby humans, tracking their locations, and estimating their future trajectories. These tasks enable the model to acquire valuable insights into the environment's social dynamics, leading to more efficient and informed planning.

3.5.3 Curriculum learning

Curriculum learning gradually increases task difficulty during training, aiding convergence in challenging social navigation tasks. In RL, this process involves three steps: task generation, sequencing, and transfer learning (Narvekar et al., 2020). *Task generation* creates scenarios of varying difficulty by adjusting obstacles, goal distances, or map complexity, using parameter sampling or grid search. *Sequencing* organizes tasks by increasing difficulty, either at a fixed rate or adaptively based on agent performance, and may involve modifying reward functions or start/goal distributions (Riedmiller et al., 2018; Florensa et al., 2018), optimization strategies (Matiisen et al., 2019), Curriculum MDPs (Narvekar et al., 2017), or human feedback (Bengio et al., 2009). *Transfer learning* adapts agents when intermediate tasks differ in state/action spaces, rewards, or dynamics, such as transitioning from precise states to noisy sensors,

or from indoor to outdoor navigation. This combination allows agents to efficiently learn complex social navigation skills.

3.5.4 Teacher-student framework

The teacher-student framework enables a teacher model, often trained with privileged information, to guide a student via real-time feedback, reward shaping, or action labels. Knowledge transfer is achieved through policy distillation (Rusu et al., 2015), using labeled paths or actions from the teacher, student, or both (Czarnecki et al., 2018), allowing the student to imitate and refine its navigation policy, which can later be fine-tuned with RL. Teachers may also provide reward signals to enhance exploration (Czarnecki et al., 2019) and corrective action feedback (Ross et al., 2011). Model-based teachers like MPC are also used (Lowrey et al., 2018). Asymmetric actor-critic methods allow the critic to use privileged information to guide the actor (Pinto et al., 2017). In teacher-student curriculum learning, teachers assign progressively harder tasks and are rewarded for student improvement (Matiisen et al., 2019), while multi-teacher approaches combine skills from specialized teachers (Rusu et al., 2015). For social navigation, non-optimal teachers (e.g., PID planners) can be combined with RL, accelerating training by switching to the higher Q-value source (Xie et al., 2018).

3.5.5 Sim-to-real

Sim-to-real transfer for navigation tackles the challenge of adapting a simulation-trained policy to perform reliably in real-world environments. Achieving sim-to-real transfer requires a highly realistic simulator (refer to Section 3.2.2) and the implementation of techniques like domain randomization and domain adaptation. These techniques operate at different levels: scenario-level randomization and adaptation (see Appendix for details) modify various aspects of the simulated environment, while sensor-level noise enables the policy to handle discrepancies in real-world sensor data. *Domain adaptation* adjusts simulation-trained models to real-world domains. For RGB data, this uses real-world samples and methods like discrepancy minimization, adversarial alignment, or reconstruction methods for feature alignment (Wang and Deng, 2018). For depth sensors, techniques such as depth completion and refinement address real-world limitations, improving consistency with simulated data (Khan et al., 2022). *Domain randomization* narrows the sim-to-real gap by introducing simulated variability, allowing policies to generalize to real-world conditions (Tobin et al., 2017). For RGB inputs, this includes varying visual features to simulate lighting and color changes (Anderson et al., 2021); for depth sensors, it involves adding noise, occlusions, warping, and quantization (Muratore et al., 2022; Thalhammer et al., 2019). Active domain randomization further improves robustness by focusing on model-affecting variations (Mehta et al., 2020; Zakharov et al., 2019).

3.6 Navigation model evaluation

Evaluating social navigation policies requires a robust approach to ensure reliable and safe robot operation in human environments. This section covers policy evaluation by outlining real-world experiments that validate a robot's capabilities in realistic, dynamic settings and by presenting metrics that offer structured, quantifiable

insights into both navigation performance and social compliance. For a more comprehensive overview of social navigation evaluation, see [Francis et al. \(2023\)](#) and [Gao and Huang \(2022\)](#).

3.6.1 Real-world experiments

Evaluating social navigation policies in real-world settings is crucial for assessing their robustness, adaptability, and social acceptability. Experiments typically fall into three categories: experimental demonstrations, lab studies, and field studies ([Mavrogiannis et al., 2023](#)). Experimental demonstrations offer proof-of-concept with limited reproducibility ([Chen et al., 2017b](#); [Chen C. et al., 2019](#)), while lab studies provide structured, repeatable tests in controlled environments with systematic reporting ([Tsai and Oh, 2020](#); [Mavrogiannis et al., 2019](#)). Field studies are the most comprehensive, deploying robots in public spaces among uninstructed pedestrians ([Kato et al., 2015](#); [Kim and Pineau, 2016](#)). Real-world evaluations combine quantitative metrics with qualitative observations, such as participant feedback or questionnaires, to assess social adaptability and compliance ([Pirk et al., 2022](#)).

3.6.2 Metrics

Navigation and social navigation metrics provide a structured framework to assess robot performance in crowded environments. Traditional navigation metrics assess robots' fundamental abilities such as reaching targets and avoiding obstacles, while social navigation metrics focus on interactions with humans, including maintaining personal space and minimizing disruptions to bystanders. Together these metrics, as detailed in [Table 9](#), guide the development of navigation systems that achieve task objectives efficiently while adhering to socially appropriate behaviors, promoting safer and widely accepted robot deployments.

4 Social navigation benchmarking

This section benchmarks state-of-the-art social navigation planners from 7 categories, assessing their performance in realistic and challenging scenarios. We achieve efficient and consistent training and evaluation processes by leveraging GPU-based simulation. Additionally, planners are adapted to handle static obstacles such as walls, as most planners only process human positions. We benchmark each planner over 6 scenarios to provide insights into the strengths, limitations, and real-world applicability.

4.1 Benchmark setup

A significant challenge in learning-based robotics, including social navigation, is the demanding computational cost of training and evaluation. To address this, we developed a benchmark that leverages GPU parallel computing to accelerate simulation and computation, significantly reducing training time and enabling more extensive experimentation and efficient benchmarking of social navigation planners.

The benchmark comprises three main components: kinematic motion simulation, sensor simulation, and crowd behavior modeling. Kinematic simulation is fully implemented on the GPU,

TABLE 9 Navigation and social navigation metrics.

Metric	Description
Success	Goal-reaching success rate
Timeout	Exceeded time limit runs
Path Length	Total path distance
Run Time	Completion time
SPL (Anderson et al., 2018b)	Path efficiency-weighted success
Static Obstacle Collision	Static obstacle collisions count
Velocity Metrics	Min, max, and avg. velocity
Speed efficiency (Fraichard and Levesy, 2020)	Ratio of nominal and actual speed
Acceleration Metrics	Min, max, and avg. acceleration
Jerk Metrics	Min, max, and avg. jerk
Obstacle Distance	Min, max, and avg. distance to obstacle
Path Irregularity (Guzzi et al., 2013)	Total deviation from a straight path
Topological complexity (Mavrogiannis et al., 2018)	Amount of entanglement within trajectory
Path Efficiency	Actual vs. straight-line ratio
Failure To Progress (Francis et al., 2023)	Goal progress failure over time period
Human Collision	Number of collisions with humans
Social Distance	Min, max, and avg. distance to humans
Min Time To Collide (Francis et al., 2023)	Min time to projected human collision
Crowd Density	Mean/max crowd density around robot
Virtual Collision (Paez-Granados et al., 2022)	Virtual boundary violations count
Personal Space (Wang et al., 2022c)	Avg time in minimum personal space
Legibility (Dragan et al., 2013)	Goal matches human expectation given robot motion
Predictability (Dragan et al., 2013)	Motion matches human expectation given known goal
Projected Path (Wang et al., 2022c)	Avg duration of path overlap with pedestrian
Following Rate (Puig et al., 2023)	Steps with maintained distance in human-follow task
SPS (Puig et al., 2023)	Path-weighted finding success in human find and follow task

including all computations for rewards and metrics, allowing efficient calculation of agent positions with respect to the map and robot frame. Sensor simulation is also performed on the GPU using Habitat Sim (Savva et al., 2019), which supports RGB and depth camera emulation (see Figures 3e,f), and we generate 2D LiDAR observations via ray casting. The Habitat 3.0 (Puig et al., 2023) codebase further enables photorealistic rendering of 3D moving humans at high frame rates, achieving around 600 FPS for crowds of 40 humans. Existing crowd behavior models are primarily CPU-based, relying on well-established libraries. For diversity and robustness, we incorporate two models: SFM, using the implementation from Gao (2025) with parameters from Helbing et al. (2005), and ORCA, using the implementation from Stüvel (2025) and parameters based on Chen C. et al. (2019).

To enhance training efficiency and success, we employed curriculum learning during training. This technique gradually increases the difficulty of the scenarios as the robot improves. Initially, training focuses on less challenging configurations. As the training progresses, parameters such as crowd density and goal distance are systematically increased. Additionally, training focuses on scenarios where the robot performs poorly, ensuring that the robot performs well across all scenarios.

During evaluation, scenario parameters, including crowd density, goal distances, and map complexity, are randomly and uniformly sampled to ensure diverse testing conditions.

4.2 Benchmark scenarios

Benchmark scenarios, illustrated in Figures 3a–d, are designed to comprehensively evaluate social navigation by simulating a range of real-world challenges robots may encounter in crowded indoor and outdoor environments (Gao and Huang, 2022; Francis et al., 2023; Stratton et al., 2024). The robot's start and goal positions, environment size, and crowd density are randomized within defined bounds, with safety constraints to avoid infeasible or unsafe initializations. Six representative scenarios are included: a static scenario with only obstacles to test navigation in narrow spaces; a doorway scenario evaluating interactions at chokepoints (Singamaneni et al., 2022); a corridor scenario capturing integration into unidirectional or bidirectional crowd flows; an intersection scenario representing complex areas where two flows meet; and open space scenarios that simulate unconstrained environments using both random and data-driven human motion, incorporating realistic pedestrian behavior from ETH (Pellegrini et al., 2009) and UCY (Lerner et al., 2007) datasets.

4.3 Benchmark planners

To evaluate social navigation strategies, we selected planners based on relevance, novelty, performance, and available implementations. The benchmark features three baselines and six learning-based planners, each covering a distinct social navigation category, along with an imitation-learning method. Since many learning-based planners do not natively handle static obstacles, we extend them with a LiDAR network (Fan et al., 2020), ensuring fair evaluation in environments with both dynamic and static obstacles.

4.3.1 Baseline planners

The baseline planners include ORCA (Van den Berg et al., 2008), the SFM (Helbing and Molnar, 1995), and DWA (Fox et al., 1997). They serve as classical foundations for comparison with advanced methods. Each is given privileged access to the map layout and all human positions, ensuring optimal performance under ideal conditions.

4.3.2 End-to-end planner

The end-to-end planner is based on the RL policy from Fan et al. (2020), which processes recent 2D LiDAR scans with a 1D CNN, combines them with the robot's state, and uses an MLP for action selection. Due to its suboptimal performance, we adopt an RNN-enhanced architecture (Hoeller et al., 2021), where the CNN output and robot state are fed into a GRU network, improving results. This end-to-end model learns navigation directly from sensor data, without using human state information.

4.3.3 Imitation learning-based planner

We implement Behavioral Cloning (BC) for imitation learning, offering a simple alternative to methods like GAIL (Ho and Ermon, 2016) without needing simulated environments. Trained on 35,000 successful human attention-based planner episodes, matching the planner's performance would signal robust generalization from real-world data. The network architecture mirrors the human attention-based planner.

4.3.3.1 Human position-based planner

The GA3C-CADRL (Everett et al., 2018) planner uses an actor-critic policy with an LSTM to process human positions and velocities. We extend it with a LiDAR network (Fan et al., 2020) for static obstacle handling, enabling navigation in mixed environments. The LSTM input is zero-padded, and in scenarios without humans, the LSTM layer is skipped.

4.3.4 Human attention-based planner

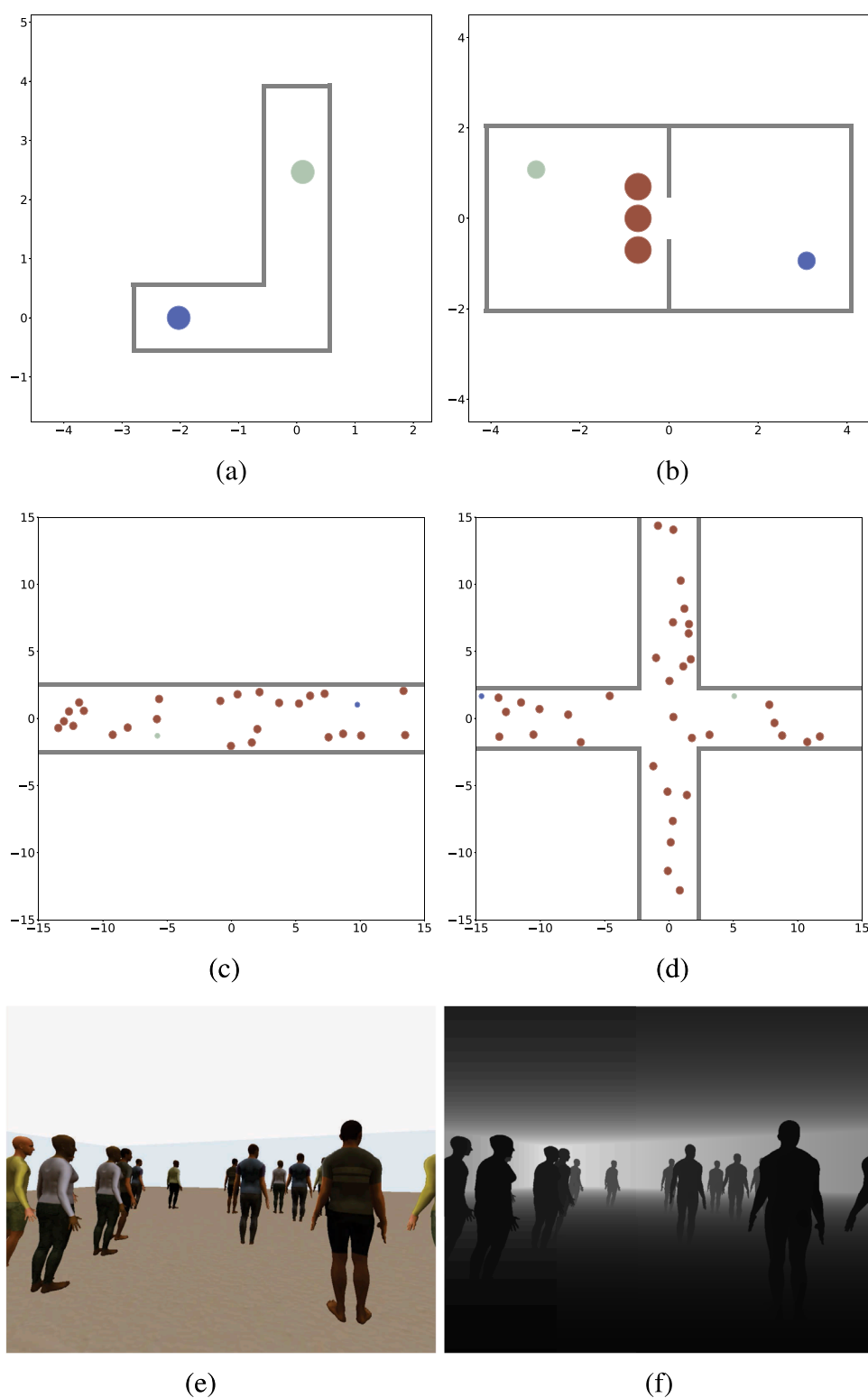
The SARL planner (Chen C. et al., 2019) employs an attention-based network to model robot-human attentions. We extend the original value network to an actor-critic framework and add a LiDAR network (Fan et al., 2020) for static obstacle handling. Unlike Liu et al. (Liu L. et al., 2020), which switches between separate policies for human and non-human scenarios, our approach uses a learned embedding to pad human input when no humans are present.

4.3.5 Human prediction-based planner

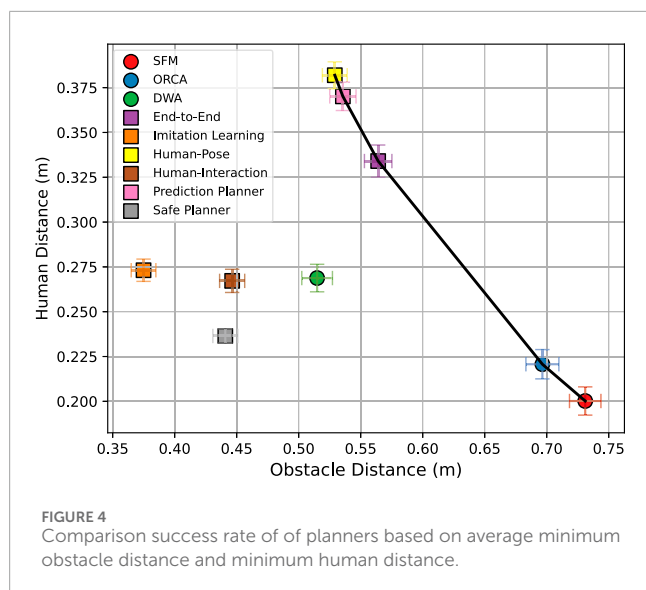
The prediction-based planner adapts the RGL model (Chen C. et al., 2020), integrating robot state and LiDAR input to predict human trajectories in the robot frame. These predicted trajectories are processed by an actor-critic policy, following the SARL planner (Chen C. et al., 2019), to handle fixed-size trajectories. When no humans are present, a learned embedding pads the input for consistency.

4.3.6 Safety-aware planner

Inspired by Linh et al. (2022), the safety-aware planner combines ORCA (Van den Berg et al., 2008) for static environments and the human attention-based planner for dynamic settings, using a policy switcher based on obstacle proximity. This hybrid approach

**FIGURE 3**

Top-down illustrations of the navigation scenarios **(a–d)**, where the robot is shown in blue, the goal in green, and humans in red. Example RGB and depth images from the benchmark rendered using Habitat Sim are shown in **(e,f)**.



balances safety and efficiency by adapting to both static and human-dense scenarios.

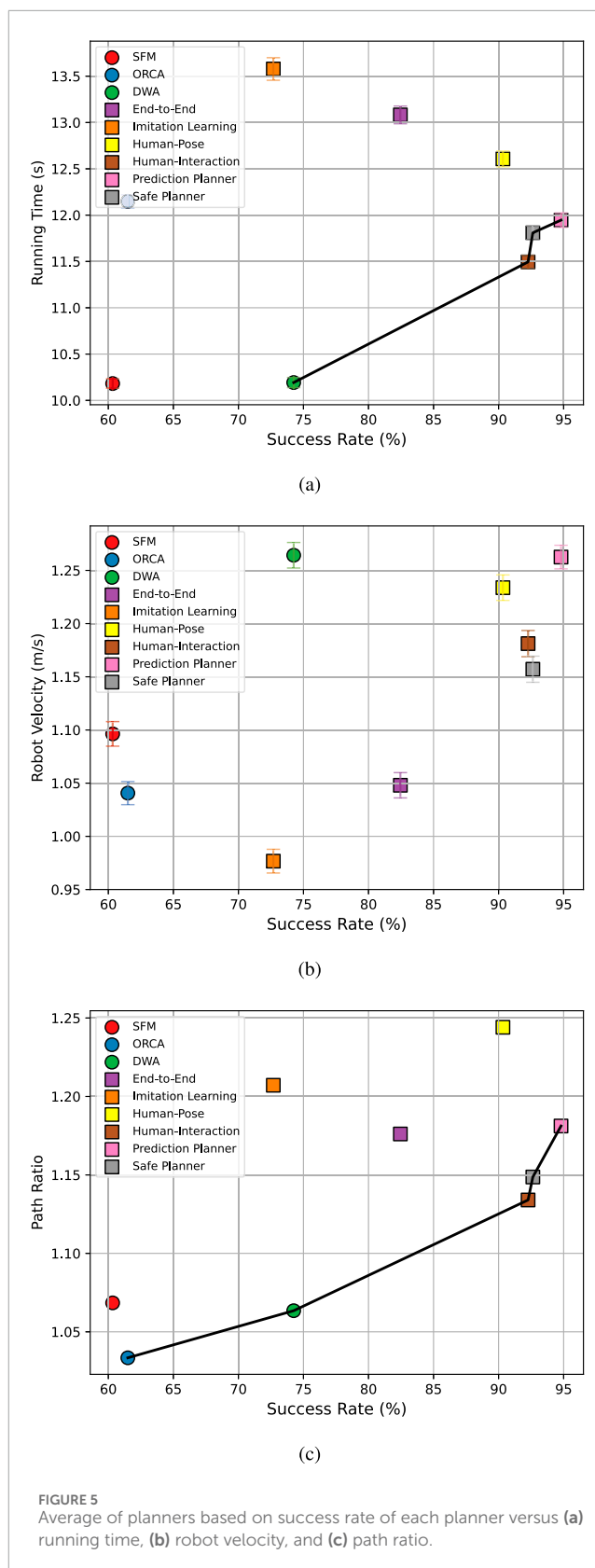
4.4 Results

Across six scenarios, learning-based planners consistently outperform model-based methods. In terms of success rate and safety, many of these learned policies consistently outperform traditional approaches. Unlike model-based planners, which prioritize obstacle avoidance, learning-based planners tend to emphasize maintaining a safe distance from humans, as illustrated in Figure 4.

In the static scenario, all methods avoid collisions, so success rate and runtime distinguish performance. Model-based planners like ORCA achieve high success but are slower, while learning-based planners are overall faster, sometimes at the expense of a higher timeout rate. Imitation Learning struggles to generalize here. In the doorway scenario, where human-robot interactions are frequent, learning-based planners adapt better, leading to safer navigation and fewer collisions.

In the corridor scenario, both model-based and learning-based planners perform comparably, managing high success rates, efficiency, and safety distances. In contrast, in the intersection scenario, learning-based methods, particularly the prediction-based planner, achieve higher success rates.

In the open space random scenario, learning-based planners achieve higher success rates and smoother navigation by adapting to dynamic human movement, reducing congestion. Model-based methods, while faster, incur more collisions due to riskier behavior. This pattern holds across most scenarios as shown in Figures 5a,c. In the open space data-driven scenario, learning-based planners remain safer while matching the running times of model-based approaches.



Among learning-based methods, end-to-end RL is notably conservative and prioritizes safety. Imitation Learning generalizes well in open spaces but struggles in constrained settings. The human position-based planner excels in open areas through direct spatial awareness, while the Human attention-based planner adapts best in crowded environments using attention mechanisms. The safety-aware planner balances efficiency and safety but remains limited by its learning-based component. The prediction-based planner, with its prediction module and expressive architecture, achieves the highest overall success rate and velocity, as shown in [Figure 5b](#).

5 Discussion and future directions

Despite the progress in social navigation, several challenges remain for learning-based social navigation to achieve safe and reliable real-world deployment. We organize this discussion around three priority levels: foundational requirements for safety and robustness, socially aligned behaviors for human acceptance, and capabilities that improve transparency and versatility.

5.1 Foundations for safety and realism

Ensuring safe and robust navigation is the highest priority for real-world deployment.

5.1.1 Safety and robustness

Most safety-oriented planners, like multi-policy approaches ([Sun et al., 2019](#); [Katyal et al., 2020](#); [Fan et al., 2020](#)), assume that reducing speed enhances safety, but this is not always valid; rapid maneuvers may be needed in dynamic, crowded settings. Relying solely on speed reduction can compromise safety in complex environments. Similarly, several human-prediction-based methods ([Yao et al., 2024](#); [Zhu et al., 2025](#)) primarily forecast human motion without explicitly modeling the robot's influence on the crowd, which limits their ability to generate safe and adaptive plans. Instead, planners should learn context-aware safe behaviors, adjusting speed as needed and responding to emergencies, to achieve both safety and efficiency without unnecessary conservativeness.

5.1.2 Scenario diversity and generalization

A major limitation to model generalizability is the limited diversity of training scenarios. Future benchmarks should incorporate a wider range of realistic, data-driven scenarios reflecting true pedestrian distributions and start-goal configurations. Long-term crowd tracking datasets similar to, or even larger than ATC ([Bršćić et al., 2013](#)), which capture varied environments in a shopping mall, can help provide such diversity.

5.1.3 Physics and sensor realism

Physics simulators range from simple kinematic to detailed dynamic models, with high-fidelity simulation improving sim-to-real transfer and enabling robot-specific planners that can integrate low-level control, such as direct wheel velocities. Likewise, accurate sensor simulation enhances robustness; while most simulators use generic models for simplicity ([Inc D, 2025](#)), sensor-specific models that replicate real-world parameters and noise can significantly improve generalization and sim-to-real performance.

5.1.4 Realistic crowd simulation

Most crowd simulation methods focus on human-human and human-obstacle interactions, but accurately modeling human-robot interactions remains a challenge. Some approaches ignore the robot's presence ([Chen Y. et al., 2020](#); [Dugas et al., 2021](#)), leading to unrealistic and overly conservative behavior, while others treat robots as humans or add randomness for robustness ([Chen C. et al., 2019](#); [Stratton et al., 2024](#)). However, these do not fully capture the diverse ways humans respond to robots, which depend on robot-specific factors like size, shape, and movement. More advanced crowd models that reflect these characteristics are needed for realistic social navigation simulation.

5.1.5 Robust evaluation

Advancing social navigation requires robust benchmarking methods that can accurately represent the planner's performance. Key directions include adopting realistic crowd simulation, conducting real-world evaluations, and refining social metrics ([Francis et al., 2023](#); [Gao and Huang, 2022](#)). Automated, objective real-world evaluation frameworks are increasingly important, as subjective user feedback is impractical to standardize. Future evaluations could use objective, non-verbal indicators, such as body language or facial expressions to better assess human comfort and social acceptance, ensuring planners are both effective and socially appropriate.

5.2 Social alignment and preferences

Beyond safety, social navigation must align with human expectations and adapt to cultural and individual differences.

5.2.1 Social norms and compliance

Social norms are informal rules guiding behavior in shared spaces, extending beyond collision avoidance and proxemics ([Hall, 1963](#)). For instance, smoothly avoiding social groups is addressed by some crowd prediction methods ([Bisagno et al., 2018](#); [Fernando et al., 2019](#)), but is incorporated into only a few navigation algorithms ([Bhaskara et al., 2023](#)). Other norms, such as culturally specific conventions ([Chen et al., 2017c](#)), are context-sensitive and not universal, suggesting the value of learning social norms directly from large-scale crowd data rather than relying solely on handcrafted heuristics. Vision-language models (VLMs) open an additional pathway by enabling robots to ground these norms in natural language, reason about complex social contexts, and even communicate intentions to humans in interpretable ways. Effective social navigation will likely require a combination of data-driven norm learning and VLM-based reasoning, alongside intention communication that may be verbal ([Dugas et al., 2020](#); [Nishimura and Yonetani, 2020](#)) or conveyed through non-verbal cues, as highlighted in autonomous vehicle research ([Habibovic et al., 2018](#)).

5.2.2 Human preferences

Social navigation is not a one-size-fits-all solution. Individuals and crowds vary in preferred comfort distance, speed, and interaction style. Future work should emphasize preference-aware navigation, where robots learn and adapt to individual users or cultural groups, potentially combining reinforcement learning with preference learning, feedback, or large language models

that capture human expectations and feedback. Although current approaches consider human preferences during training (Choi et al., 2020), accommodating post-deployment feedback and achieving continuous learning remain open challenges.

5.3 Transparency and reasoning

To ensure long-term acceptance, learning-based systems must be interpretable, communicative, and capable of reasoning based on context.

5.3.1 Explainability and transparency

A major challenge in learning-based planners is the difficulty of interpreting the reasoning behind their decisions, which is often referred to as *explainability* (Vouros, 2022). Integrating explainability improves user trust, allows better debugging, and clarifies the decision-making process. Several techniques exist, such as *saliency maps*, which visually indicate influential regions within image-based inputs (Huber et al., 2021), and approaches that provide verbal explanations for their decisions (Dugas et al., 2020). Integrating these explainability methods into learning-based social navigation can create more transparent, interpretable, and user-friendly systems.

5.3.2 Social vision-language navigation

Recent advances in vision-language navigation (VLNs) (An et al., 2022) highlight opportunities to enrich social navigation with multimodal reasoning capabilities and improve functional versatility. Beyond instruction following (Anderson et al., 2018a), VLNs can support a wide range of tasks such as visual question answering (Wu et al., 2024), describing social situations, or embodied dialog (Hahn et al., 2020). Social VLN could allow robots to interpret human intent, infer social norms from linguistic context, and communicate their own decisions in interpretable ways.

Author contributions

RA: Writing – original draft, Software, Writing – review and editing. CC: Writing – review and editing. RR: Writing – review and editing. DP-G: Writing – review and editing, Methodology.

Funding

The author(s) declare that financial support was received for the research and/or publication of this article. This

research work was partially supported by the Innosuisse Project 103.421 IP-IC “Developing an AI-enabled Robotic Personal Vehicle for Reduced Mobility Population in Complex Environments”.

Acknowledgements

We acknowledge the support of S. Dey for providing feedback on the initial draft of the manuscript. ChatGPT-4 was used to assist with grammar checks and basic fact-checking in this review.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declare that Generative AI was used in the creation of this manuscript. ChatGPT-4 was used to assist with grammar checks and basic fact-checking in this review.

Any alternative text (alt text) provided alongside figures in this article has been generated by Frontiers with the support of artificial intelligence and reasonable efforts have been made to ensure accuracy, including review by the authors wherever possible. If you identify any issues, please contact us.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/frobt.2025.1658643/full#supplementary-material>

References

- Achiam, J., Held, D., Tamar, A., and Abbeel, P. (2017). “Constrained policy optimization,” in *International Conference on Machine Learning*. Sydney, Australia: PMLR, 22–31.
- Alahi, A., Goel, K., Ramanathan, V., Robicquet, A., Fei-Fei, L., and Savarese, S. (2016). “Social lstm: human trajectory prediction in crowded spaces,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 961–971.
- Alonso-Mora, J., Breitenmoser, A., Rufli, M., Beardsley, P., and Siegwart, R. (2013). “Optimal reciprocal collision avoidance for multiple non-holonomic robots,” in *Distributed autonomous robotic systems: the 10th international symposium*. Springer, 203–216.
- Amano, K., and Kato, Y. (2022). “Autonomous Mobile robot navigation for complicated environments by switching multiple control policies,” in *IECON 2022–48th annual conference of the IEEE industrial electronics Society (IEEE)*, 1–6.

- An, D., Qi, Y., Li, Y., Huang, Y., Wang, L., Tan, T., et al. (2022). Bevdet: multimodal map pre-training for language-guided navigation. *arXiv Prepr. arXiv:2212.04385*.
- Anderson, P., Wu, Q., Teney, D., Bruce, J., Johnson, M., Sünderhauf, N., et al. (2018a). "Vision-and-language navigation: interpreting visually-grounded navigation instructions in real environments," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3674–3683.
- Anderson, P., Chang, A., Chaplot, D. S., Dosovitskiy, A., Gupta, S., Koltun, V., et al. (2018b). On evaluation of embodied navigation agents. *arXiv preprint arXiv:1807.06757*.
- Anderson, P., Shrivastava, A., Truong, J., Majumdar, A., Parikh, D., Batra, D., et al. (2021). "Sim-to-real transfer for vision-and-language navigation," in *Conference on robot learning (PMLR)*, 671–681.
- Arjovsky, M., Chintala, S., and Bottou, L. (2017). "Wasserstein generative adversarial networks," in *International Conference on Machine Learning*. Sydney, Australia: PMLR, 214–223.
- Aroor, A., Epstein, S. L., and Korpan, R. (2017). "Mengeros: a crowd simulation tool for autonomous robot navigation," in *AAAI fall symposium series*.
- Arras, K. O., Mozos, O. M., and Burgard, W. (2007). "Using boosted features for the detection of people in 2d range data," in *Proceedings 2007 IEEE international conference on robotics and automation (IEEE)*, 3402–3407.
- Bae, J. W., Kim, J., Yun, J., Kang, C., Choi, J., Kim, C., et al. (2024). Sit dataset: socially interactive pedestrian trajectory dataset for social navigation robots. *Adv. Neural Inf. Process. Syst.* 36.
- Bansal, S., Bajcsy, A., Ratner, E., Dragan, A. D., and Tomlin, C. J. (2020). "A hamilton-jacobi reachability-based framework for predicting and analyzing human motion for safe planning," in *2020 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 7149–7155.
- Bastani, O., Pu, Y., and Solar-Lezama, A. (2018). Verifiable reinforcement learning via policy extraction. *Adv. neural Inf. Process. Syst.* 31.
- Benfold, B., and Reid, I. (2011). Stable multi-target tracking in real-time surveillance video. *CVPR 2011 (IEEE)*, 3457–3464. doi:10.1109/cvpr.2011.5995667
- Bengio, Y., Louradour, J., Collobert, R., and Weston, J. (2009). "Curriculum learning," in *Proceedings of the 26th annual international conference on machine learning*, 41–48.
- Bertoni, L., Kreiss, S., Mordan, T., and Alahi, A. (2021). Monstereo: when monocular and stereo meet at the tail of 3d human localization. *IEEE International Conference on Robotics and Automation ICRA*, 5126–5132.
- Beyer, L., Hermans, A., and Leibe, B. (2016). Drow: Real-Time deep learning-based wheelchair detection in 2-d range data. *IEEE Robotics Automation Lett.* 2, 585–592. doi:10.1109/lra.2016.2645131
- Beyer, L., Hermans, A., Linder, T., Arras, K. O., and Leibe, B. (2018). Deep person detection in 2d range data. *arXiv Prepr. arXiv:1804.02463*.
- Bhaskara, R., Chiu, M., and Bera, A. (2023). Sg-lstm: social group lstm for robot navigation through dense crowds. *IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, 3835–3840.
- Bisagno, N., Zhang, B., and Conci, N. (2018). "Group lstm: group trajectory prediction in crowded scenarios," in *Proceedings of the European conference on computer vision (ECCV) workshops*.
- Blundell, C., Cornebise, J., Kavukcuoglu, K., and Wierstra, D. (2015). "Weight uncertainty in neural network," in *International Conference on Machine Learning*. Lille, France: PMLR, 1613–1622.
- Bock, J., Krajewski, R., Moers, T., Runde, S., Vater, L., and Eckstein, L. (2020). "The ind dataset: a drone dataset of naturalistic road user trajectories at german intersections," in *2020 IEEE intelligent vehicles symposium (IV)*. IEEE, 1929–1934.
- Bojarski, M. (2016). End to end learning for self-driving cars. *arXiv Prepr. arXiv:1604.07316*.
- Brito, B., Everett, M., How, J. P., and Alonso-Mora, J. (2021). Where to go next: learning a subgoal recommendation policy for navigation in dynamic environments. *IEEE Robotics Automation Lett.* 6, 4616–4623. doi:10.1109/lra.2021.3068662
- Brown, D., Goo, W., Nagarajan, P., and Niekum, S. (2019). "Extrapolating beyond suboptimal demonstrations via inverse reinforcement learning from observations," in *International Conference on Machine Learning*. Long Beach, California, United States: PMLR, 783–792.
- Brown, D. S., Goo, W., and Niekum, S. (2020). "Better-than-demonstrator imitation learning via automatically-ranked demonstrations," in *Conference on robot learning (PMLR)*, 330–359.
- Bršćić, D., Kanda, T., Ikeda, T., and Miyashita, T. (2013). Person tracking in large public spaces using 3-d range sensors. *IEEE Trans. Human-Machine Syst.* 43, 522–534. doi:10.1109/thms.2013.2283945
- Burgard, W., Cremers, A. B., Fox, D., Hähnel, D., Lakemeyer, G., Schulz, D., et al. (1999). Experiences with an interactive museum tour-guide robot. *Artif. Intell.* 114, 3–55. doi:10.1016/s0004-3702(99)00070-3
- Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., et al. (2020). "Nuscenes: a multimodal dataset for autonomous driving," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11621–11631.
- Campbell, T., Liu, M., Kulis, B., How, J. P., and Carin, L. (2013). Dynamic clustering via asymptotics of the dependent dirichlet process mixture. *Adv. Neural Inf. Process. Syst.* 26.
- Cancelli, E., Campari, T., Serafini, L., Chang, A. X., and Ballan, L. (2023). "Exploiting proximity-aware tasks for embodied social navigation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 10957–10967.
- Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. (2020). "End-to-end object detection with transformers," in *European conference on computer vision*. Springer, 213–229.
- Carlevaris-Bianco, N., Ushani, A. K., and Eustice, R. M. (2016). University of michigan north campus long-term vision and lidar dataset. *Int. J. Robotics Res.* 35, 1023–1035. doi:10.1177/0278364915614638
- Cen, J., Zhang, S., Pei, Y., Li, K., Zheng, H., Luo, M., et al. (2023). Cmdfusion: bidirectional fusion network with cross-modality knowledge distillation for lidar semantic segmentation. *IEEE Robotics Automation Lett.* 9, 771–778. doi:10.1109/lra.2023.3335771
- Chandra, R., Bhattacharya, U., Roncal, C., Bera, A., and Manocha, D. (2019). "Robusttp: end-to-end trajectory prediction for heterogeneous road-agents in dense traffic with noisy sensor inputs," in *Proceedings of the 3rd ACM computer science in cars symposium*, 1–9.
- Chang, A., Dai, A., Funkhouser, T., Halber, M., Niessner, M., Savva, M., et al. (2017). Matterport3d: learning from rgb-d data in indoor environments. *arXiv Prepr. arXiv:1709.06158*, 667–676. doi:10.1109/3dv.2017.00081
- Charalampous, K., Kostavelis, I., and Gasteratos, A. (2016). Robot navigation in large-scale social maps: an action recognition approach. *Expert Syst. Appl.* 66, 261–273. doi:10.1016/j.eswa.2016.09.026
- Charalampous, K., Kostavelis, I., and Gasteratos, A. (2017). Recent trends in social aware robot navigation: a survey. *Robotics Aut. Syst.* 93, 85–104. doi:10.1016/j.robot.2017.03.002
- Chavdarova, T., Baqué, P., Bouquet, S., Maksai, A., Jose, C., Bagautdinov, T., et al. (2018). "Wildtrack: a multi-camera hd dataset for dense unscripted pedestrian detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5030–5039.
- Chen, Y., Wu, F., Shuai, W., and Chen, X. (2017a). Robots serve humans in public places—keja robot as a shopping assistant. *Int. J. Adv. Robotic Syst.* 14, 172988141770356. doi:10.1177/1729881417703569
- Chen, Y. F., Liu, M., Everett, M., and How, J. P. (2017b). "Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning," in *2017 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 285–292.
- Chen, Y. F., Everett, M., Liu, M., and How, J. P. (2017c). Socially aware motion planning with deep reinforcement learning. *IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, 1343–1350.
- Chen, X., Ma, H., Wan, J., Li, B., and Xia, T. (2017d). "Multi-view 3d object detection network for autonomous driving," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1907–1915.
- Chen, Z., Song, C., Yang, Y., Zhao, B., Hu, Y., Liu, S., et al. (2018). Robot navigation based on human trajectory prediction and multiple travel modes. *Appl. Sci.* 8, 2205. doi:10.3390/app8112205
- Chen, W., Zhou, S., Pan, Z., Zheng, H., and Liu, Y. (2019a). Mapless collaborative navigation for a multi-robot system based on the deep reinforcement learning. *Appl. Sci.* 9, 4198. doi:10.3390/app9204198
- Chen, C., Liu, Y., Kreiss, S., and Alahi, A. (2019b). "Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning," in *2019 international conference on robotics and automation (ICRA) (IEEE)*, 6015–6022.
- Chen, Y., Liu, C., Shi, B. E., and Liu, M. (2020a). Robot navigation in crowds by graph convolutional networks with attention learned from human gaze. *IEEE Robotics Automation Lett.* 5, 2754–2761. doi:10.1109/lra.2020.2972868
- Chen, C., Hu, S., Nikdel, P., Mori, G., and Savva, M. (2020b). "Relational graph learning for crowd navigation," in *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 10007–10013.
- Chen, L., Paleja, R., and Gombolay, M. (2021). "Learning from suboptimal demonstration via self-supervised reward regression," in *Conference on robot learning (PMLR)*, 1262–1277.
- Chen, S., Zhu, H., Chen, X., Lei, Y., Yu, G., and Chen, T. (2023). "End-to-end 3d dense captioning with vote2cap-detr," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11124–11133.
- Cheng, G., Wang, Y., Dong, L., Cai, W., and Sun, C. (2023). Multi-objective deep reinforcement learning for crowd-aware robot navigation with dynamic human preference. *Neural Comput. Appl.* 35, 16247–16265. doi:10.1007/s00521-023-08385-4
- Chiang, H. T. L., Faust, A., Fiser, M., and Francis, A. (2019). Learning navigation behaviors end-to-end with autorl. *IEEE Robotics Automation Lett.* 4, 2007–2014. doi:10.1109/lra.2019.2899918
- Choi, W., Shahid, K., and Savarese, S. (2009). "What are they doing? collective activity classification using spatio-temporal relationship among people," in *2009 IEEE*

- 12th international conference on computer vision workshops, ICCV workshops (IEEE), 1282–1289.
- Choi, J., Park, K., Kim, M., and Seok, S. (2019). Deep reinforcement learning of navigation in a complex and crowded environment with a limited field of view. *Int. Conf. Robotics Automation (ICRA) (IEEE)*, 5993–6000. doi:10.1109/icra.2019.8793979
- Choi, J., Dance, C., Kim, J., Park, K., Han, J., Seo, J., et al. (2020). “Fast adaptation of deep reinforcement learning-based navigation skills to human preference,” in *2020 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 3363–3370.
- Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S., and Amodei, D. (2017). Deep reinforcement learning from human preferences. *Adv. neural Inf. Process. Syst.* 30.
- Chuang, T. K., Lin, N. C., Chen, J. S., Hung, C. H., Huang, Y. W., Teng, C., et al. (2018). “Deep trail-following robotic guide dog in pedestrian environments for people who are blind and visually impaired-learning from virtual and real worlds,” in *2018 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 5849–5855.
- Cong, P., Zhu, X., Qiao, F., Ren, Y., Peng, X., Hou, Y., et al. (2022). “Stcrowd: a multimodal dataset for pedestrian perception in crowded scenes,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 19608–19617.
- Costa, E. D. S., and Gouvea, Jr M. M. (2010). “Autonomous navigation in dynamic environments with reinforcement learning and heuristic,” in *2010 ninth international conference on machine learning and applications*. IEEE, 37–42.
- Cui, Y., Zhang, H., Wang, Y., and Xiong, R. (2021). “Learning world transition model for socially aware robot navigation,” in *2021 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 9262–9268.
- Curtis, S., Best, A., and Menge, M. D. (2016). A modular framework for simulating crowd movement. *Collect. Dyn.* 1, 1–40.
- Czarnecki, W., Jayakumar, S., Jaderberg, M., Hasenclever, L., Teh, Y. W., Heess, N., et al. (2018). “Mix and match agent curricula for reinforcement learning,” in *International Conference on Machine Learning*. Stockholm: PMLR, 1087–1095.
- Czarnecki, W. M., Pascanu, R., Osindero, S., Jayakumar, S., Swirszcz, G., and Jaderberg, M. (2019). “Distilling policy distillation,” in *The 22nd International Conference on Artificial Intelligence and Statistics*. Okinawa, Japan: PMLR, 1331–1340.
- Dalal, N., and Triggs, B. (2005). Histograms of oriented gradients for human detection. *2005 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR'05) (IEEE)* 1, 886–893. doi:10.1109/cvpr.2005.177
- Datseris, G., Vahdati, A. R., and DuBois, T. C. (2024). Agents. jl: a performant and feature-full agent-based modeling software of minimal code complexity. *Simulation* 100, 1019–1031. doi:10.1177/00375497211068820
- de Heuvel, J., Corral, N., Bruckschen, L., and Bennewitz, M. (2022). “Learning personalized human-aware robot navigation using virtual reality demonstrations from a user study,” in *2022 31st IEEE international conference on robot and human interactive communication (RO-MAN)*. IEEE, 898–905.
- de Heuvel, J., Corral, N., Kreis, B., Conrad, J., Driemel, A., and Bennewitz, M. (2023). *Learning depth vision-based personalized robot navigation from dynamic demonstrations in virtual reality*. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, 6757–6764.
- de Heuvel, J., Sethuraman, T., and Bennewitz, M. (2024). Learning adaptive multi-objective robot navigation with demonstrations. *arXiv preprint arXiv:2404.04857*.
- Dequaire, J., Ondruška, P., Rao, D., Wang, D., and Posner, I. (2018). Deep tracking in the wild: end-to-end tracking using recurrent neural networks. *Int. J. Robotics Res.* 37, 492–512. doi:10.1177/0278364917710543
- Dragan, A. D., Lee, K. C., and Srinivasa, S. S. (2013). Legibility and predictability of robot motion. *8th ACM/IEEE Int. Conf. Human-Robot Interact. (HRI) (IEEE)*, 301–308. doi:10.1109/hri.2013.6483603
- Dugas, D., Nieto, J., Siegwart, R., and Chung, J. J. (2020). *Ian: multi-behavior navigation planning for robots in real, crowded environments*. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, 11368–11375.
- Dugas, D., Nieto, J., Siegwart, R., and Chung, J. J. (2021). “Navrep: unsupervised representations for reinforcement learning of robot navigation in dynamic human environments,” in *2021 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 7829–7835.
- Dynamics, P. (2025). *Jupedsim*. Available online at: <https://github.com/PedestrianDynamics/jupedsim>.
- Echeverria, G., Lassabe, N., Degroote, A., and Lemaignan, S. (2011). “Modular open robots simulation engine: morse,” in *2011 IEEE international conference on robotics and automation (IEEE)*, 46–51.
- Eiffert, S., Kong, H., Pirmarzdashti, N., and Sukkari, S. (2020a). “Path planning in dynamic environments using generative rnns and monte carlo tree search,” in *2020 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 10263–10269.
- Eiffert, S., Li, K., Shan, M., Worrall, S., Sukkari, S., and Nebot, E. (2020b). Probabilistic crowd gan: multimodal pedestrian trajectory prediction using a graph vehicle-pedestrian attention network. *IEEE Robotics Automation Lett.* 5, 5026–5033. doi:10.1109/lra.2020.3004324
- Eppeberger, T., Cesari, G., Dymczyk, M., Siegwart, R., and Dubé, R. (2020). “Leveraging stereo-camera data for real-time dynamic obstacle detection and tracking,” in *IEEE/RSJ international conference on intelligent robots and systems (IROS) (IEEE)*, 10528–10535.
- Ettinger, S., Cheng, S., Caine, B., Liu, C., Zhao, H., Pradhan, S., et al. (2021). “Large scale interactive motion forecasting for autonomous driving: the waymo open motion dataset,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 9710–9719.
- Everett, M., Chen, Y. F., and How, J. P. (2018). *Motion planning among dynamic, decision-making agents with deep reinforcement learning*. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, 3052–3059.
- Everett, M., Chen, Y. F., and How, J. P. (2021). Collision avoidance in pedestrian-rich environments with deep reinforcement learning. *Ieee Access* 9, 10357–10377. doi:10.1109/access.2021.3050338
- Fahad, M., Chen, Z., and Guo, Y. (2018). *Learning how pedestrians navigate: a deep inverse reinforcement learning approach*. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, 819–826.
- Fan, T., Cheng, X., Pan, J., Manocha, D., and Yang, R. (2018). Crowdmove: autonomous mapless navigation in crowded scenarios. *arXiv preprint arXiv:1807.07870*.
- Fan, T., Long, P., Liu, W., and Pan, J. (2020). Distributed multi-robot collision avoidance via deep reinforcement learning for navigation in complex scenarios. *Int. J. Robotics Res.* 39, 856–892. doi:10.1177/0278364920916531
- Faure, S. (2025). *Cromosim*. Available online at: <https://www.cromosim.fr>.
- Faust, A., Oslund, K., Ramirez, O., Francis, A., Tapia, L., Fiser, M., et al. (2018). “Prm-rl: long-range robotic navigation tasks by combining reinforcement learning and sampling-based planning,” in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 5113–5120.
- Felzenszwalb, P., McAllester, D., and Ramanan, D. (2008). “A discriminatively trained, multiscale, deformable part model,” in *2008 IEEE conference on computer vision and pattern recognition (IEEE)*, 1–8.
- Fernando, T., Denman, S., Sridharan, S., and Fookes, C. (2018). Soft+ hardwired attention: an lstm framework for human trajectory prediction and abnormal event detection. *Neural Netw.* 108, 466–478. doi:10.1016/j.neunet.2018.09.002
- Fernando, T., Denman, S., Sridharan, S., and Fookes, C. (2019). “Gd-gan: generative adversarial networks for trajectory prediction and group detection in crowds,” in *Computer Vision-ACCV 2018: 14th Asian conference on computer vision, Perth, Australia, December 2–6, 2018, revised selected papers, part I 14*. Springer, 314–330.
- Ferrer, G., Zulueta, A. G., Cotarello, F. H., and Sanfeliu, A. (2017). Robot social-aware navigation framework to accompany people walking side-by-side. *Aut. robots* 41, 775–793. doi:10.1007/s10514-016-9584-y
- Finn, C., and Levine, S. (2017). “Deep visual foresight for planning robot motion,” in *2017 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 2786–2793.
- Florensa, C., Held, D., Geng, X., and Abbeel, P. (2018). “Automatic goal generation for reinforcement learning agents,” in *International Conference on Machine Learning*. Stockholm: PMLR, 1515–1528.
- Fox, D., Burgard, W., and Thrun, S. (1997). The dynamic window approach to collision avoidance. *IEEE Robotics and Automation Mag.* 4, 23–33. doi:10.1109/100.580977
- Fraichard, T., and Levesy, V. (2020). From crowd simulation to robot navigation in crowds. *IEEE Robotics Automation Lett.* 5, 729–735. doi:10.1109/lra.2020.2965032
- Francis, A., Pérez-d’Arpino, C., Li, C., Xia, F., Alahi, A., Alami, R., et al. (2023). Principles and guidelines for evaluating social robot navigation algorithms. *arXiv Prepr. arXiv:2306.16740*.
- Gal, Y., and Ghahramani, Z. (2016). “Dropout as a bayesian approximation: representing model uncertainty in deep learning,” in *International conference on machine learning (PMLR)*, 1050–1059.
- Gao, Y. (2025). *Pysocialforce*. Available online at: <https://github.com/yuxiang-gao/PySocialForce>.
- Gao, Y., and Huang, C. M. (2022). Evaluation of socially-aware robot navigation. *Front. Robotics AI* 8, 721317. doi:10.3389/frobt.2021.721317
- Gao, J., Ye, W., Guo, J., and Li, Z. (2020). Deep reinforcement learning for indoor mobile robot path planning. *Sensors* 20, 5493. doi:10.3390/s20195493
- Geiger, A., Lenz, P., and Urtasun, R. (2012). “Are we ready for autonomous driving? The kitti vision benchmark suite,” in *2012 IEEE conference on computer vision and pattern recognition (IEEE)*, 3354–3361.
- Geyer, J., Kassahun, Y., Mahmudi, M., Ricou, X., Durgesh, R., Chung, A. S., et al. (2020). A2d2: Audi autonomous driving dataset. *arXiv preprint arXiv:2004.06320*.
- Girdhar, R., Ramanan, D., Gupta, A., Sivic, J., and Russell, B. (2017). “Actionvlad: learning spatio-temporal aggregation for action classification,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 971–980.
- Gloor, C. (2016). *Pedsim: pedestrian crowd simulation*. Available online at: <http://pedsim.silmaril.org>.

- Gong, Z., Hu, T., Qiu, R., and Liang, J. (2024). From cognition to precognition: a future-aware framework for social navigation. *arXiv preprint arXiv:2409.13244*.
- Gonon, D. J., Paez-Granados, D., and Billard, A. (2021). Reactive navigation in crowds for non-holonomic robots with convex bounding shape. *IEEE Robotics Automation Lett.* 6, 4728–4735. doi:10.1109/lra.2021.3068660
- Group, C. D. (2025). Crowd dynamics. Available online at: <https://github.com/crowddynamics/crowddynamics>.
- Grzeskowiak, F., Gonon, D., Dugas, D., Paez-Granados, D., Chung, J. J., Nieto, J., et al. (2021). “Crowd against the machine: a simulation-based benchmark tool to evaluate and compare robot capabilities to navigate a human crowd,” in *2021 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 3879–3885.
- Gu, T., Chen, G., Li, J., Lin, C., Rao, Y., Zhou, J., et al. (2022). “Stochastic trajectory prediction via motion indeterminacy diffusion,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 17113–17122.
- Gupta, M., Kumar, S., Behera, L., and Subramanian, V. K. (2016). A novel vision-based tracking algorithm for a human-following mobile robot. *IEEE Trans. Syst. Man, Cybern. Syst.* 47, 1415–1427. doi:10.1109/tsmc.2016.2616343
- Gupta, A., Johnson, J., Fei-Fei, L., Savarese, S., and Alahi, A. (2018). “Social gan: socially acceptable trajectories with generative adversarial networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2255–2264.
- Guzzi, J., Giusti, A., Gambardella, L. M., Theraulaz, G., and Di Caro, G. A. (2013). “Human-friendly robot navigation in dynamic environments,” in *2013 IEEE international conference on robotics and automation (IEEE)*, 423–430.
- Ha, D., and Schmidhuber, J. (2018). Recurrent world models facilitate policy evolution. *Adv. neural Inf. Process. Syst.* 31.
- Habibovic, A., Lundgren, V. M., Andersson, J., Klingegård, M., Lagström, T., Sirkka, A., et al. (2018). Communicating intent of automated vehicles to pedestrians. *Front. Psychol.* 9, 1336. doi:10.3389/fpsyg.2018.01336
- Hahn, M., Krantz, J., Batra, D., Parikh, D., Reh, J. M., Lee, S., et al. (2020). Where are you? Localization from embodied dialog. 806, 822. doi:10.18653/v1/2020.emnlp-main.59
- Hall, E. T. (1963). A system for the notation of proxemic behavior. *Am. Anthropol.* 65, 1003–1026. doi:10.1525/aa.1963.65.5.02a00020
- Hamandi, M., D’Arcy, M., and Fazli, P. (2019). “Deepmotion: learning to navigate like humans,” in *2019 28th IEEE international conference on robot and human interactive communication (RO-MAN)*. IEEE, 1–7.
- Han, Y., Zhan, I. H., Zhao, W., Pan, J., Zhang, Z., Wang, Y., et al. (2022a). Deep reinforcement learning for robot collision avoidance with self-state-attention and sensor fusion. *IEEE Robotics Automation Lett.* 7, 6886–6893. doi:10.1109/lra.2022.3178791
- Han, R., Chen, S., Wang, S., Zhang, Z., Gao, R., Hao, Q., et al. (2022b). Reinforcement learned distributed multi-robot navigation with reciprocal velocity obstacle shaped rewards. *IEEE Robotics Automation Lett.* 7, 5896–5903. doi:10.1109/lra.2022.3161699
- Hayes, C. F., Rădulescu, R., Bargiacchi, E., Källström, J., Macfarlane, M., Reymond, M., et al. (2022). A practical guide to multi-objective reinforcement learning and planning. *Aut. Agents Multi-Agent Syst.* 36, 26. doi:10.1007/s10458-022-09552-y
- Helbing, D., and Molnar, P. (1995). Social force model for pedestrian dynamics. *Phys. Rev. E* 51, 4282–4286. doi:10.1103/physreve.51.4282
- Helbing, D., Buzna, L., Johansson, A., and Werner, T. (2005). Self-organized pedestrian crowd dynamics: experiments, simulations, and design solutions. *Transp. Sci.* 39, 1–24. doi:10.1287/trsc.1040.0108
- Hirose, N., Shah, D., Sridhar, A., and Levine, S. (2023). Sacson: scalable autonomous control for social navigation. *IEEE Robotics Automation Lett.* 9, 49–56. doi:10.1109/lra.2023.3329626
- Ho, J., and Ermon, S. (2016). Generative adversarial imitation learning. *Adv. neural Inf. Process. Syst.* 29.
- Hoeller, D., Wellhausen, L., Farshidian, F., and Hutter, M. (2021). Learning a state representation and navigation in cluttered and dynamic environments. *IEEE Robotics Automation Lett.* 6, 5081–5088. doi:10.1109/lra.2021.3068639
- Hong, Y., Wu, Q., Qi, Y., Rodriguez-Opazo, C., and Gould, S. (2021). “Vln bert: a recurrent vision-and-language bert for navigation,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 1643–1653.
- Hu, H. N., Cai, Q. Z., Wang, D., Lin, J., Sun, M., Krahenbuhl, P., et al. (2019). Joint monocular 3d vehicle detection and tracking. *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 5390–5399.
- Huang, Y., Bi, H., Li, Z., Mao, T., and Wang, Z. (2019). “Stgat: modeling spatial-temporal interactions for human trajectory prediction,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 6272–6281.
- Huang, Z., Li, R., Shin, K., and Driggs-Campbell, K. (2021). Learning sparse interaction graphs of partially detected pedestrians for trajectory prediction. *IEEE Robotics Automation Lett.* 7, 1198–1205. doi:10.1109/lra.2021.3138547
- Huber, T., Weitz, K., André, E., and Amir, O. (2021). Local and global explanations of agent behavior: integrating strategy summaries with saliency maps. *Artif. Intell.* 301, 103571. doi:10.1016/j.artint.2021.103571
- Inc D (2025). Velodyne simulator. Available online at: https://wiki.ros.org/velodyne_simulator.
- Jaderberg, M., Mnih, V., Czarnecki, W. M., Schaul, T., Leibo, J. Z., Silver, D., et al. (2016). Reinforcement learning with unsupervised auxiliary tasks. *arXiv Prepr. arXiv:1611.05397*.
- Jang, J., and Ghaffari, M. (2024). Social zone as a barrier function for socially-compliant robot navigation. *IFAC-PapersOnLine* 58, 157–162. doi:10.1016/j.ifacol.2025.01.173
- Jaradat, M. A. K., Al-Rousan, M., and Quadan, L. (2011). Reinforcement based mobile robot navigation in dynamic environment. *Robotics Computer-Integrated Manuf.* 27, 135–149. doi:10.1016/j.rcim.2010.06.019
- Jia, D., Hermans, A., and Leibe, B. (2020). *Dr-spaam: a spatial-attention and auto-regressive model for person detection in 2d range data*. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, 10270–10277.
- Jin, J., Nguyen, N. M., Sakib, N., Graves, D., Yao, H., and Jagersand, M. (2020). “Mapless navigation among dynamics with social-safety-awareness: a reinforcement learning approach from 2d laser scans,” in *2020 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 6979–6985.
- Karamouzas, I., Skinner, B., and Guy, S. J. (2014). Universal power law governing pedestrian interactions. *Phys. Rev. Lett.* 113, 238701. doi:10.1103/physrevlett.113.238701
- Karnan, H., Nair, A., Xiao, X., Warnell, G., Pirk, S., Toshev, A., et al. (2022). Socially compliant navigation dataset (scand): a large-scale dataset of demonstrations for social navigation. *IEEE Robotics Automation Lett.* 7, 11807–11814. doi:10.1109/lra.2022.3184025
- Kästner, L., Buiyan, T., Jiao, L., Le, T. A., Zhao, X., Shen, Z., et al. (2021). “Arenarosnav: towards deployment of deep-reinforcement-learning-based obstacle avoidance into conventional autonomous navigation systems,” in *2021 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 6456–6463.
- Kato, Y., Kanda, T., and Ishiguro, H. (2015). May i help you? Design of human-like polite approaching behavior. *Proc. Tenth Annu. ACM/IEEE Int. Conf. Human-Robot Interact.*, 35–42. doi:10.1145/2696454.2696463
- Katyal, K. D., Hager, G. D., and Huang, C. M. (2020). “Intent-aware pedestrian prediction for adaptive crowd navigation,” in *2020 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 3277–3283.
- Khan, M. A. U., Nazir, D., Pagani, A., Mokayed, H., Liwicki, M., Stricker, D., et al. (2022). A comprehensive survey of depth completion approaches. *Sensors* 22, 6969. doi:10.3390/s22186969
- Kim, B., and Pineau, J. (2016). Socially adaptive path planning in human environments using inverse reinforcement learning. *Int. J. Soc. Robotics* 8, 51–66. doi:10.1007/s12369-015-0310-2
- Kim, A., Oşep, A., and Leal-Taixé, L. (2021). “Eagermot: 3d multi-object tracking via sensor fusion,” in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 11315–11321.
- Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., et al. (2023). “Segment anything,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 4015–4026.
- Kleinmeier, B., Zönnchen, B., Gödel, M., and Köster, G. (2019). Vadere: an open-source simulation framework to promote interdisciplinary understanding. *arXiv Prepr. arXiv:1907.09520* 4, A21. doi:10.17815/cd.2019.21
- Koenig, N., and Howard, A. (2004). Design and use paradigms for gazebo, an open-source multi-robot simulator. *2004 IEEE/RSJ Int. Conf. intelligent robots Syst. (IROS) (IEEE Cat. No. 04CH37566) (Ieee)* 3, 2149–2154. doi:10.1109/iros.2004.1389727
- Kolve, E., Mottaghi, R., Han, W., VanderBilt, E., Weihs, L., Herrasti, A., et al. (2017). Ai2-thor: an interactive 3d environment for visual ai. *arXiv Prepr. arXiv:1712.05474*.
- Korbmacher, R., and Tordeux, A. (2022). Review of pedestrian trajectory prediction methods: comparing deep learning and knowledge-based approaches. *IEEE Trans. Intelligent Transp. Syst.* 23, 24126–24144. doi:10.1109/tits.2022.3205676
- Kreiss, S. (2021). Deep social force. *arXiv preprint arXiv:2109.12081*.
- Kruse, T., Pandey, A. K., Alami, R., and Kirsch, A. (2013). Human-aware robot navigation: a survey. *Robotics Aut. Syst.* 61, 1726–1743. doi:10.1016/j.robot.2013.05.007
- Kulhánek, J., Derner, E., De Bruin, T., and Babuška, R. (2019). “Vision-based navigation using deep reinforcement learning,” in *2019 european conference on mobile robots (ECMR) (IEEE)*, 1–8.
- Lasota, P. A., Fong, T., and Shah, J. A. (2017). A survey of methods for safe human-robot interaction. *Found. Trends Robotics* 5, 261–349. doi:10.1561/23000000052
- Lee, H., and Jeong, J. (2023). Velocity range-based reward shaping technique for effective map-less navigation with lidar sensor and deep reinforcement learning. *Front. Neurobotics* 17, 1210442. doi:10.3389/fnbot.2023.1210442
- Lee, K., Kim, S., and Choi, J. (2023). *Adaptive and explainable deployment of navigation skills via hierarchical deep reinforcement learning*. IEEE International Conference on Robotics and Automation ICRA, 1673–1679.

- Leigh, A., Pineau, J., Olmedo, N., and Zhang, H. (2015). "Person tracking and following with 2d laser scanners," in *2015 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 726–733.
- Lerner, A., Chrysanthou, Y., and Lischinski, D. (2007). "Crowds by example," *Comput. Graph. forum*, 26, 655–664. doi:10.1111/j.1467-8659.2007.01089.x
- Li, M., Jiang, R., Ge, S. S., and Lee, T. H. (2018). Role playing learning for socially concomitant mobile robot navigation. *CAAI Trans. Intell. Technol.* 3, 49–58. doi:10.1049/trit.2018.0008
- Li, K., Shan, M., Narula, K., Worrall, S., and Nebot, E. (2020). "Socially aware crowd navigation with multimodal pedestrian trajectory prediction for autonomous vehicles," in *2020 IEEE 23rd international conference on intelligent transportation systems (ITSC)*. IEEE, 1–8.
- Li, C., Xia, F., Martín-Martín, R., Lingelbach, M., Srivastava, S., Shen, B., et al. (2021). Igbison 2.0: object-centric simulation for robot learning of everyday household tasks. *arXiv Prepr. arXiv:2108.03272*.
- Li, J., Han, R., Yan, H., Qian, Z., Feng, W., and Wang, S. (2022). "Self-supervised social relation representation for human group detection," in *European conference on computer vision*. Springer, 142–159.
- Li, H., Li, M., Cheng, Z. Q., Dong, Y., Zhou, Y., He, J. Y., et al. (2024). Human-aware vision-and-language navigation: bridging simulation to reality with dynamic human interactions. *Adv. Neural Inf. Process. Syst.* 37, 119411–119442.
- Liang, J., Patel, U., Sathyaamoorthy, A. J., and Manocha, D. (2021). "Crowd-steer: real-time smooth and collision-free robot navigation in densely crowded scenarios trained using high-fidelity simulation," in *Proceedings of the twenty-ninth international conference on international joint conferences on artificial intelligence*, 4221–4228.
- Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017). "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2117–2125.
- Linh, K. U., Cox, J., Buiyan, T., and Lambrecht, J. (2022). "All-in-one: a drl-based control switch combining state-of-the-art navigation planners," in *2022 International Conference on Robotics and Automation (ICRA)*, 2861–2867. doi:10.1109/icra46639.2022.9811797
- Lisotto, M., Coscia, P., and Ballan, L. (2019). "Social and scene-aware trajectory prediction in crowded spaces," in *Proceedings of the IEEE/CVF international conference on computer vision workshops*.
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., et al. (2016). "Ssd: single shot multibox detector," in *Computer Vision—ECCV 2016: 14th European conference, Amsterdam, the Netherlands, October 11–14, 2016, Proceedings, part I 14*. Springer, 21–37.
- Liu, Y., Xu, A., and Chen, Z. (2018). *Map-based deep imitation learning for obstacle avoidance*. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, 8644–8649.
- Liu, L., Dugas, D., Cesari, G., Siegwart, R., and Dubé, R. (2020a). "Robot navigation in crowded environments using deep reinforcement learning," in *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS) (IEEE)*, 5671–5677.
- Liu, Z., Suo, C., Liu, Y., Shen, Y., Qiao, Z., Wei, H., et al. (2020b). Deep learning-based localization and perception systems: approaches for autonomous cargo transportation vehicles in large-scale, semiclosed environments. *IEEE Robotics and Automation Mag.* 27, 139–150. doi:10.1109/mra.2020.2977290
- Liu, S., Chang, P., Liang, W., Chakraborty, N., and Driggs-Campbell, K. (2021). "Decentralized structural-rnn for robot crowd navigation with deep reinforcement learning," in *2021 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 3517–3524.
- Liu, Z., Zhai, Y., Li, J., Wang, G., Miao, Y., and Wang, H. (2023a). Graph relational reinforcement learning for mobile robot navigation in large-scale crowded environments. *IEEE Trans. Intelligent Transp. Syst.* 24, 8776–8787. doi:10.1109/tits.2023.3269533
- Liu, S., Chang, P., Huang, Z., Chakraborty, N., Hong, K., Liang, W., et al. (2023b). "Intention aware robot crowd navigation with attention-based interaction graph," in *IEEE international conference on robotics and automation (ICRA)*. IEEE, 12015–12021.
- Liu, H., Li, C., Wu, Q., and Lee, Y. J. (2023c). Visual instruction tuning. *Adv. neural Inf. Process. Syst.* 36, 34892–34916.
- Liu, Y., Lerch, L., Palmieri, L., Rudenko, A., Koch, S., Ropinski, T., et al. (2025). Context-aware human behavior prediction using multimodal large language models: challenges and insights. *arXiv Prepr. arXiv:2504.00839*.
- Long, P., Liu, W., and Pan, J. (2017). Deep-learned collision avoidance policy for distributed multiagent navigation. *IEEE Robotics Automation Lett.* 2, 656–663. doi:10.1109/lra.2017.2651371
- Long, P., Fan, T., Liao, X., Liu, W., Zhang, H., and Pan, J. (2018). "Towards optimally decentralized multi-robot collision avoidance via deep reinforcement learning," in *2018 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 6252–6259.
- Lopez, N. G., Nuin, Y. L. E., Moral, E. B., Juan, L. U. S., Rueda, A. S., Vilches, V. M., et al. (2019). gym-gazebo2, a toolkit for reinforcement learning using ros 2 and gazebo. *arXiv Prepr. arXiv:1903.06278*.
- Lou, J., Lin, H., Marshall, D., Saupe, D., and Liu, H. (2022). Transalnet: towards perceptually relevant visual saliency prediction. *Neurocomputing* 494, 455–467. doi:10.1016/j.neucom.2022.04.080
- Lowrey, K., Rajeswaran, A., Kakade, S., Todorov, E., and Mordatch, I. (2018). Plan online, learn offline: efficient learning and exploration via model-based control. *arXiv preprint arXiv:1811.01848*.
- Luo, S., Sun, P., Zhu, J., Deng, Y., Yu, C., Xiao, A., et al. (2025). Gson: a group-based social navigation framework with large multimodal model. *IEEE Robotics Automation Lett.* 10, 9646–9653. doi:10.1109/lra.2025.3595038
- Lütjens, B., Everett, M., and How, J. P. (2019). Safe reinforcement learning with model uncertainty estimates. *Int. Conf. Robotics Automation (ICRA) (IEEE)*, 8662–8668. doi:10.1109/icra.2019.8793611
- Ma, F., and Karaman, S. (2018). "Sparse-to-dense: depth prediction from sparse depth samples and a single image," in *2018 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 4796–4803.
- Ma, Y. J., Liang, W., Wang, G., Huang, D. A., Bastani, O., Jayaraman, D., et al. (2023). Eureka: human-level reward design via coding large language models. *arXiv preprint arXiv:2310.12931*.
- Majecka, B. (2009). *Statistical models of pedestrian behaviour in the forum*. Citeseer: Ph.D. thesis.
- Makoviychuk, V., Wawrzyniak, L., Guo, Y., Lu, M., Storey, K., Macklin, M., et al. (2021). Isaac gym: high performance gpu-based physics simulation for robot learning. *arXiv preprint arXiv:2108.10470*.
- Manhardt, F., Kehl, W., and Gaidon, A. (2019). "Roi-10d: monocular lifting of 2d detection to 6d pose and metric shape," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2069–2078.
- Mao, J., Shi, S., Wang, X., and Li, H. (2023a). 3d object detection for autonomous driving: a comprehensive survey. *Int. J. Comput. Vis.* 131, 1909–1963. doi:10.1007/s11263-023-01790-1
- Mao, W., Xu, C., Zhu, Q., Chen, S., and Wang, Y. (2023b). "Leapfrog diffusion model for stochastic trajectory prediction," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 5517–5526.
- Marta, D., Holk, S., Pek, C., Tumova, J., and Leite, I. (2023). *Aligning human preferences with baseline objectives in reinforcement learning*. IEEE International Conference on Robotics and Automation ICRA, 7562–7568.
- Martin-Martín, R., Patel, M., Rezafooghi, H., Shenoi, A., Gwak, J., Frankel, E., et al. (2021). Jrdb: a dataset and benchmark of egocentric robot visual perception of humans in built environments. *IEEE Trans. pattern analysis Mach. Intell.* 45, 6748–6765. doi:10.1109/tpami.2021.3070543
- Martinez-Baselga, D., Riazuelo, L., and Montano, L. (2023). Improving robot navigation in crowded environments using intrinsic rewards. *arXiv Prepr. arXiv:2302.06554*, 9428–9434. doi:10.1109/icra48891.2023.10160876
- Masad, D., and Kazil, J. L. (2015). Mesa: an agent-based modeling framework. *SciPy (Citeseer)*, 51–58. doi:10.25080/majora-7b98e3ed-009
- Matheson, E., Minto, R., Zampieri, E. G., Faccio, M., and Rosati, G. (2019). Human-robot collaboration in manufacturing applications: a review. *Robotics* 8, 100. doi:10.3390/robotics8040100
- Matiisen, T., Oliver, A., Cohen, T., and Schulman, J. (2019). Teacher–student curriculum learning. *IEEE Trans. neural Netw. Learn. Syst.* 31, 3732–3740. doi:10.1109/tnnls.2019.2934906
- Mavrogiannis, C. I., Thomason, W. B., and Knepper, R. A. (2018). "Social momentum: a framework for legible navigation in dynamic multi-agent environments," in *Proceedings of the 2018 ACM/IEEE international conference on human-robot interaction*, 361–369.
- Mavrogiannis, C., Hutchinson, A. M., Macdonald, J., Alves-Oliveira, P., and Knepper, R. A. (2019). Effects of distinct robot navigation strategies on human behavior in a crowded environment. *14th ACM/IEEE Int. Conf. Human-Robot Interact. (HRI) (IEEE)*, 421–430. doi:10.1109/hri.2019.8673115
- Mavrogiannis, C., Baldini, F., Wang, A., Zhao, D., Trautman, P., Steinfeld, A., et al. (2023). Core challenges of social robot navigation: a survey. *ACM Trans. Human-Robot Interact.* 12, 1–39. doi:10.1145/3583741
- Mehta, B., Diaz, M., Golemo, F., Pal, C. J., and Paull, L. (2020). "Active domain randomization," in *Conference on robot learning (PMLR)*, 1162–1176.
- Michel, O. (2004). Cyberbotics Ltd. webots™: professional mobile robot simulation. *Int. J. Adv. Robotic Syst.* 1, 5. doi:10.5772/5618
- Milioto, A., Vizzo, I., Behley, J., and Stachniss, C. (2019). "Rangenet++: fast and accurate lidar semantic segmentation," in *2019 IEEE/RSJ international conference on intelligent robots and systems (IROS) (IEEE)*, 4213–4220.
- Miller, J., Hasfura, A., Liu, S. Y., and How, J. P. (2016). *Dynamic arrival rate estimation for campus mobility on demand network graphs*. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, 2285–2292.
- Mirowski, P., Pascanu, R., Viola, F., Soyer, H., Ballard, A. J., Banino, A., et al. (2016). Learning to navigate in complex environments. *arXiv Prepr. arXiv:1611.03673*.

- Mirsky, R., Xiao, X., Hart, J., and Stone, P. (2021). Prevention and resolution of conflicts in social navigation—a survey. *arXiv preprint arXiv:2106.12113*.
- Mittal, M., Yu, C., Yu, Q., Liu, J., Rudin, N., Hoeller, D., et al. (2023). Orbit: a unified simulation framework for interactive robot learning environments. *IEEE Robotics Automation Lett.* 8, 3740–3747. doi:10.1109/lra.2023.3270034
- Mohamed, A., Qian, K., Elhoseiny, M., and Claudel, C. (2020). “Social-stgcnn: a social spatio-temporal graph convolutional neural network for human trajectory prediction,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 14424–14432.
- Mohan, M., and Salgoankar, A. (2018). A survey of robotic motion planning in dynamic environments. *Robotics Aut. Syst.* 100, 171–185. doi:10.1016/j.robot.2017.10.011
- Möller, R., Furnari, A., Battiatto, S., Härmä, A., and Farinella, G. M. (2021). A survey on human-aware robot navigation. *Robotics Aut. Syst.* 145, 103837. doi:10.1016/j.robot.2021.103837
- Monaci, G., Aractingi, M., and Silander, T. (2022). Dipcan: distilling privileged information for crowd-aware navigation. *Robotics Sci. Syst.*
- Mousavian, A., Anguelov, D., Flynn, J., and Kosecka, J. (2017). “3d bounding box estimation using deep learning and geometry,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7074–7082.
- Moussaïd, M., Perozo, N., Garnier, S., Helbing, D., and Theraulaz, G. (2010). The walking behaviour of pedestrian social groups and its impact on crowd dynamics. *PLoS one* 5, e10047. doi:10.1371/journal.pone.0010047
- Munje, M. J., Tang, C., Liu, S., Hu, Z., Zhu, Y., Cui, J., et al. (2025). Socialnav-sub: benchmarking vlms for scene understanding in social robot navigation. *arXiv Prepr. arXiv:2509.08757*.
- Muratore, F., Ramos, F., Turk, G., Yu, W., Gienger, M., and Peters, J. (2022). Robot learning from randomized simulations: a review. *Front. Robotics AI* 9, 799893. doi:10.3389/frobt.2022.799893
- Narang, S., Best, A., Curtis, S., and Manocha, D. (2015). Generating pedestrian trajectories consistent with the fundamental diagram based on physiological and psychological factors. *PLoS one* 10, e0117856. doi:10.1371/journal.pone.0117856
- Narasimhan, S., Tan, A. H., Choi, D., and Nejat, G. (2025). “Olivia-nav: an online lifelong vision language approach for mobile robot social navigation,” in *2025 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 9130–9137.
- Narayanan, V., Manoghar, B. M., Dorabala, V. S., Manocha, D., and Bera, A. (2020). “Proximo: gait-based emotion learning and multi-view proxemic fusion for socially-aware robot navigation,” in *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 8200–8207.
- Narvekar, S., Sinapov, J., and Stone, P. (2017). *Autonomous task sequencing for customized curriculum design in reinforcement learning*. *IJCAI* 2536–2542. doi:10.24963/ijcai.2017/353
- Narvekar, S., Peng, B., Leonetti, M., Sinapov, J., Taylor, M. E., and Stone, P. (2020). Curriculum learning for reinforcement learning domains: a framework and survey. *J. Mach. Learn. Res.* 21, 1–50.
- Naseer, M., Khan, S., and Porikli, F. (2018). Indoor scene understanding in 2.5/3d for autonomous agents: a survey. *IEEE access* 7, 1859–1887. doi:10.1109/access.2018.2886133
- Nguyen, D. M., Nazeri, M., Payandeh, A., Datar, A., and Xiao, X. (2023). “Toward human-like social robot navigation: a large-scale, multi-modal, social human navigation dataset,” in *2023 IEEE/RSJ international conference on intelligent robots and systems (IROS) (IEEE)*, 7442–7447.
- Nishimura, M., and Yonetani, R. (2020). *L2b: learning to balance the safety-efficiency trade-off in interactive crowd-aware robot navigation*. *IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, 11004–11010.
- Oh, S., Hoogs, A., Perera, A., Cuntoor, N., Chen, C. C., Lee, J. T., et al. (2011). “A large-scale benchmark dataset for event recognition in surveillance video,” *CVPR*, 3153–3160.
- Oh, J., Singh, S., and Lee, H. (2017). Value prediction network. *Adv. neural Inf. Process. Syst.* 30.
- Okal, B., and Arras, K. O. (2016). “Learning socially normative robot navigation behaviors with bayesian inverse reinforcement learning,” in *2016 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 2889–2895.
- Okunevich, I., Lombard, A., Krajník, T., Ruichek, Y., and Yan, Z. (2025). Online context learning for socially compliant navigation. *IEEE Robotics Automation Lett.* 10, 5042–5049. doi:10.1109/lra.2025.3557309
- Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., et al. (2022). Training language models to follow instructions with human feedback. *Adv. neural Inf. Process. Syst.* 35, 27730–27744.
- Paez-Granados, D., He, Y., Gonon, D., Huber, L., and Billard, A. (2021). 3d point cloud and rgbd of pedestrians in robot crowd navigation: detection and tracking. *IEEE DataPort* 12.
- Paez-Granados, D., He, Y., Gonon, D., Jia, D., Leibe, B., Suzuki, K., et al. (2022). *Pedestrian-robot interactions on autonomous crowd navigation: reactive control methods and evaluation metrics*. *IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, 149–156.
- Pang, Z., Li, Z., and Wang, N. (2022). “Simpletrack: understanding and rethinking 3d multi-object tracking,” in *European conference on computer vision*. Springer, 680–696.
- Parker-Holder, J., Rajan, R., Song, X., Biedenkapp, A., Miao, Y., Eimer, T., et al. (2022). Automated reinforcement learning (autorl): a survey and open problems. *J. Artif. Intell. Res.* 74, 517–568. doi:10.1613/jair.1.13596
- Pathak, D., Agrawal, P., Efros, A. A., and Darrell, T. (2017). “Curiosity-driven exploration by self-supervised prediction,” in *International Conference on Machine Learning*. Sydney, Australia: PMLR, 2778–2787.
- Paxton, C., Raman, V., Hager, G. D., and Kobilarov, M. (2017). *Combining neural networks and tree search for task and motion planning in challenging environments*. *IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, 6059–6066.
- Payandeh, A., Song, D., Nazeri, M., Liang, J., Mukherjee, P., Raj, A. H., et al. (2024). Social-llava: enhancing robot navigation through human-language reasoning in social spaces. *arXiv Prepr. arXiv:2501.09024*.
- Pellegrini, S., Ess, A., Schindler, K., and Van Gool, L. (2009). “You’ll never walk alone: modeling social behavior for multi-target tracking,” in *2009 IEEE 12th international conference on computer vision (IEEE)*, 261–268.
- Peng, Z., Liu, W., Ning, Z., Zhao, Q., Cheng, S., and Hu, J. (2024). “3d multi-object tracking in autonomous driving: a survey,” in *2024 36th Chinese control and decision conference (CCDC)*. IEEE, 4964–4971.
- Pfeiffer, M., Schaeuble, M., Nieto, J., Siegwart, R., and Cadena, C. (2017). “From perception to decision: a data-driven approach to end-to-end motion planning for autonomous ground robots,” in *2017 IEEE international conference on robotics and automation (icra) (IEEE)*, 1527–1533.
- Pfeiffer, M., Shukla, S., Turchetta, M., Cadena, C., Krause, A., Siegwart, R., et al. (2018). Reinforced imitation: sample efficient deep reinforcement learning for mapless navigation by leveraging prior demonstrations. *IEEE Robotics Automation Lett.* 3, 4423–4430. doi:10.1109/lra.2018.2869644
- Pinto, L., Andrychowicz, M., Welinder, P., Zaremba, W., and Abbeel, P. (2017). Asymmetric actor critic for image-based robot learning. *arXiv preprint arXiv:1710.06542*.
- Pirk, S., Lee, E., Xiao, X., Takayama, L., Francis, A., and Toshev, A. (2022). A protocol for validating social navigation policies. *arXiv preprint arXiv:2204.05443*.
- Poddar, S., Mavrogiannis, C., and Srinivasa, S. S. (2023). *From crowd motion prediction to robot navigation in crowds*. *IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, 6765–6772.
- Pramanik, A., Pal, S. K., Maiti, J., and Mitra, P. (2021). Granulated rcnn and multi-class deep sort for multi-object detection and tracking. *IEEE Trans. Emerg. Top. Comput. Intell.* 6, 171–181. doi:10.1109/tetci.2020.3041019
- Puig, X., Undersander, E., Szot, A., Cote, M. D., Yang, T. Y., Partsey, R., et al. (2023). Habitat 3.0: a co-habitat for humans, avatars and robots. *arXiv preprint arXiv:2310.13724*.
- Qi, C. R., Yi, L., Su, H., and Guibas, L. J. (2017). Pointnet++: deep hierarchical feature learning on point sets in a metric space. *Adv. neural Inf. Process. Syst.* 30.
- Qi, C. R., Liu, W., Wu, C., Su, H., and Guibas, L. J. (2018). “Frustum pointnets for 3d object detection from rgb-d data,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 918–927.
- Qin, Z., Wang, J., and Lu, Y. (2019). Monogonet: a geometric reasoning network for monocular 3d object localization. *Proc. AAAI Conf. Artif. Intell.* 33, 8851–8858. doi:10.1609/aaai.v33i01.33018851
- Qin, L., Huang, Z., Zhang, C., Guo, H., Ang, M., and Rus, D. (2021). *Deep imitation learning for autonomous navigation in dynamic pedestrian environments*. *IEEE International Conference on Robotics and Automation ICRA*, 4108–4115.
- Qiu, W., Zhong, F., Zhang, Y., Qiao, S., Xiao, Z., Kim, T. S., et al. (2017). “Unrealcv: virtual worlds for computer vision,” in *Proceedings of the 25th ACM international conference on multimedia*, 1221–1224.
- Qu, H., Cai, Y., and Liu, J. (2024). “LLMs are good action recognizers,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 18395–18406.
- Rakai, L., Song, H., Sun, S., Zhang, W., and Yang, Y. (2022). Data association in multiple object tracking: a survey of recent techniques. *Expert Syst. Appl.* 192, 116300. doi:10.1016/j.eswa.2021.116300
- Redmon, J. (2021). “You only look once: unified, real-time object detection,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*.
- Redmon, J. (2018). Yolo3: an incremental improvement. *arXiv preprint arXiv:1804.02767*.
- Ren, S., He, K., Girshick, R., and Sun, J. (2016). Faster r-cnn: towards real-time object detection with region proposal networks. *IEEE Trans. pattern analysis Mach. Intell.* 39, 1137–1149. doi:10.1109/tpami.2016.2577031
- Repiso, E., Garrell, A., and Sanfeliu, A. (2020). People’s adaptive side-by-side model evolved to accompany groups of people by social robots. *IEEE Robotics Automation Lett.* 5, 2387–2394. doi:10.1109/lra.2020.2970676

- Ridel, D., Deo, N., Wolf, D., and Trivedi, M. (2020). Scene compliant trajectory forecast with agent-centric spatio-temporal grids. *IEEE Robotics Automation Lett.* 5, 2816–2823. doi:10.1109/lra.2020.2974393
- Riedmiller, M., Hafner, R., Lampe, T., Neunert, M., Degraeve, J., Wiele, T., et al. (2018). “Learning by playing solving sparse reward tasks from scratch,” in *International conference on machine learning (PMLR)*, 4344–4353.
- Rios-Martinez, J., Spalanzani, A., and Laugier, C. (2015). From proxemics theory to socially-aware navigation: a survey. *Int. J. Soc. Robotics* 7, 137–153. doi:10.1007/s12369-014-0251-1
- Robicquet, A., Sadeghian, A., Alahi, A., and Savarese, S. (2016). “Learning social etiquette: human trajectory understanding in crowded scenes,” in *Computer Vision—ECCV 2016: 14th European conference, Amsterdam, the Netherlands, October 11–14, 2016, proceedings, part VIII* 14. Springer, 549–565.
- Rojiers, D. M., Vamplew, P., Whiteson, S., and Dazeley, R. (2013). A survey of multi-objective sequential decision-making. *J. Artif. Intell. Res.* 48, 67–113. doi:10.1613/jair.3987
- Rösmann, C., Hoffmann, F., and Bertram, T. (2015). “Timed-elastic-bands for time-optimal point-to-point nonlinear model predictive control,” in *2015 european control conference (ECC)*. IEEE, 3352–3357.
- Ross, S., Gordon, G., and Bagnell, D. (2011). “A reduction of imitation learning and structured prediction to no-regret online learning,” in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. Fort Lauderdale, FL, United States: JMLR Workshop and Conference Proceedings, 627–635.
- Roth, A. M., Liang, J., and Manocha, D. (2021). *Xai-n: sensor-based robot navigation using expert policies and decision trees*. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, 2053–2060.
- Roth, P., Nubert, J., Yang, F., Mittal, M., and Hutter, M. (2024). “Viplanner: visual semantic imperative learning for local navigation,” in *2024 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 5243–5249.
- Rudenko, A., Palmieri, L., Herman, M., Kitani, K. M., Gavrila, D. M., and Arras, K. O. (2020a). Human motion trajectory prediction: a survey. *Int. J. Robotics Res.* 39, 895–935. doi:10.1177/0278364920917446
- Rudenko, A., Kucner, T. P., Swaminathan, C. S., Chadavada, R. T., Arras, K. O., and Lilienthal, A. J. (2020b). THOR: human-robot navigation data collection and accurate motion trajectories dataset. *IEEE Robotics Automation Lett.* 5, 676–682. doi:10.1109/lra.2020.2965416
- Rusu, A. A., Colmenarejo, S. G., Gulcehre, C., Desjardins, G., Kirkpatrick, J., Pascanu, R., et al. (2015). Policy distillation. *arXiv preprint arXiv:1511.06295*.
- Sadeghian, A., Kosaraju, V., Sadeghian, A., Hirose, N., Rezaiofighi, H., and Savarese, S. (2019). “Sophie: an attentive gan for predicting paths compliant to social and physical constraints,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 1349–1358.
- Samsani, S. S., and Muhammad, M. S. (2021). Socially compliant robot navigation in crowded environment by human behavior resemblance using deep reinforcement learning. *IEEE Robotics Automation Lett.* 6, 5223–5230. doi:10.1109/lra.2021.3071954
- Sánchez-Ibáñez, J. R., Pérez-del Pulgar, C. J., and García-Cerezo, A. (2021). Path planning for autonomous mobile robots: a review. *Sensors* 21, 7898. doi:10.3390/s21237898
- Sathiamoorthy, A. J., Liang, J., Patel, U., Guan, T., Chandra, R., and Manocha, D. (2020a). *Densecavoid: real-time navigation in dense crowds using anticipatory behaviors*. IEEE International Conference on Robotics and Automation ICRA, 11345–11352.
- Sathiamoorthy, A. J., Patel, U., Guan, T., and Manocha, D. (2020b). Frozone: freezing-free, pedestrian-friendly navigation in human crowds. *IEEE Robotics Automation Lett.* 5, 4352–4359. doi:10.1109/lra.2020.2996593
- Savva, M., Kadian, A., Maksymets, O., Zhao, Y., Wijmans, E., Jain, B., et al. (2019). “Habitat: a platform for embodied ai research,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 9339–9347.
- Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.
- Seitz, M. J., and Köster, G. (2012). Natural discretization of pedestrian movement in continuous space. *Phys. Rev. E—Statistical, Nonlinear, Soft Matter Phys.* 86, 046108. doi:10.1103/PhysRevE.86.046108
- Shi, H., Shi, L., Xu, M., and Hwang, K. S. (2019a). End-to-end navigation strategy with deep reinforcement learning for mobile robots. *IEEE Trans. Industrial Inf.* 16, 2393–2402. doi:10.1109/tii.2019.2936167
- Shi, S., Wang, X., and Li, H. (2019b). “Pointcnn: 3d object proposal generation and detection from point cloud,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 770–779.
- Sighencea, B. I., Stanciu, R. I., and Căleanu, C. D. (2021). A review of deep learning-based methods for pedestrian trajectory prediction. *Sensors* 21, 7543. doi:10.3390/s21227543
- Simonyan, K., and Zisserman, A. (2014). Two-stream convolutional networks for action recognition in videos. *Adv. neural Inf. Process. Syst.* 27.
- Singamaneni, P. T., Favier, A., and Alami, R. (2022). *Watch out! there may be a human. addressing invisible humans in social navigation*. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, 11344–11351.
- Singamaneni, P. T., Bachiller-Burgos, P., Manso, L. J., Garrell, A., Sanfeliu, A., Spalanzani, A., et al. (2024). A survey on socially aware robot navigation: taxonomy and future challenges. *Int. J. Robotics Res.*, 02783649241230562.
- Smart, W. D., and Kaelbling, L. P. (2000). Practical reinforcement learning in continuous spaces. *ICML*, 903–910.
- Smart, W. D., and Kaelbling, L. P. (2002). “Effective reinforcement learning for mobile robots,” in *Proceedings 2002 IEEE international conference on robotics and automation (cat. No. 02CH37292)*, IEEE 4, 3404–3410. doi:10.1109/robot.2002.1014237
- Song, D., Liang, J., Payandeh, A., Raj, A. H., Xiao, X., and Manocha, D. (2024). Vlm-social-nav: socially aware robot navigation through scoring using vision-language models. *IEEE Robotics Automation Lett.* 10, 508–515. doi:10.1109/lra.2024.3511409
- Sprague, Z., Chandra, R., Holtz, J., and Biswas, J. (2023). Socialgym 2.0: simulator for multi-agent social robot navigation in shared human spaces. *arXiv Prepr. arXiv:2303.05584*.
- Stratton, A., Hauser, K., and Mavrogiannis, C. (2024). Characterizing the complexity of social robot navigation scenarios. *arXiv Prepr. arXiv:2405.11410*.
- Strigel, E., Meissner, D., Seeliger, F., Wilking, B., and Dietmayer, K. (2014). “The kaper intersection laserscanner and video dataset,” in *17th international IEEE conference on intelligent transportation systems (ITSC)*. IEEE, 1900–1901.
- Stüvel, S. A. (2025). Python-rvo2 library. Available online at: <https://github.com/sybreinstuvel/Python-RVO2>.
- Sun, L., Zhai, J., and Qin, W. (2019). Crowd navigation in an unknown and dynamic environment based on deep reinforcement learning. *IEEE Access* 7, 109544–109554. doi:10.1109/access.2019.2933492
- Sun, P., Wang, W., Chai, Y., Elsayed, G., Bewley, A., Zhang, X., et al. (2021). “Rsn: range sparse net for efficient, accurate lidar 3d object detection,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 5725–5734.
- Szot, A., Clegg, A., Undersander, E., Wijmans, E., Zhao, Y., Turner, J., et al. (2021). Habitat 2.0: training home assistants to rearrange their habitat. *Adv. neural Inf. Process. Syst.* 34, 251–266.
- Tai, L., Paolo, G., and Liu, M. (2017). “Virtual-to-real deep reinforcement learning: continuous control of mobile robots for mapless navigation,” in *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 31–36.
- Tai, L., Zhang, J., Liu, M., and Burgard, W. (2018). “Socially compliant navigation through raw depth inputs with generative adversarial imitation learning,” in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 1111–1117.
- Tamar, A., Wu, Y., Thomas, G., Levine, S., and Abbeel, P. (2016). Value iteration networks. *Adv. neural Inf. Process. Syst.* 29.
- Tan, Q., Fan, T., Pan, J., and Manocha, D. (2020). *Deepmnavigate: deep reinforced multi-robot navigation unifying local and global collision avoidance*. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS IEEE, 6952–6959.
- Thalhammer, S., Park, K., Patten, T., Vincze, M., and Kropatsch, W. (2019). Sydd: synthetic depth data randomization for object detection using domain-relevant background. *Stift Vorau, Austria: Computer Vision Winter Workshop*, 14–22.
- Thomaz, A., Hoffman, G., and Cakmak, M. (2016). Computational human-robot interaction. *Found. Trends Robotics* 4 (2–3), 105–223. doi:10.1561/23000000049
- Thrun, S., Beetz, M., Bennewitz, M., Burgard, W., Cremers, A. B., Dellaert, F., et al. (2000). Probabilistic algorithms and the interactive museum tour-guide robot minerva. *Int. J. robotics Res.* 19, 972–999. doi:10.1177/02783640022067922
- Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., and Abbeel, P. (2017). *Domain randomization for transferring deep neural networks from simulation to the real world*. IEEE/RSJ international conference on intelligent robots and systems IROS, 23–30.
- Tongloy, T., Chuwongin, S., Jaksukam, K., Chousangsunton, C., and Boonsang, S. (2017). “Asynchronous deep reinforcement learning for the mobile robot navigation with supervised auxiliary tasks,” in *2017 2nd international conference on robotics and automation engineering (ICRAE)*. IEEE, 68–72.
- Truong, X. T., and Ngo, T. D. (2017). “to approach humans?”: a unified framework for approaching pose prediction and socially aware robot navigation. *IEEE Trans. Cognitive Dev. Syst.* 10, 557–572. doi:10.1109/tcds.2017.2751963
- Tsai, C. E., and Oh, J. (2020). “A generative approach for socially compliant navigation,” in *2020 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 2160–2166.
- Tsoi, N., Hussein, M., Espinoza, J., Ruiz, X., and Vázquez, M. (2020). “Sean: social environment for autonomous navigation,” in *Proceedings of the 8th international conference on human-agent interaction*, 281–283.
- Tsoi, N., Hussein, M., Fugikawa, O., Zhao, J., and Vázquez, M. (2021). *An approach to deploy interactive robotic simulators on the web for hri experiments: results in social robot navigation*. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, 7528–7535.

- Tsoi, N., Xiang, A., Yu, P., Sohn, S. S., Schwartz, G., Ramesh, S., et al. (2022). Sean 2.0: formalizing and generating social situations for robot navigation. *IEEE Robotics Automation Lett.* 7, 11047–11054. doi:10.1109/ra.2022.3196783
- Van den Berg, J., Lin, M., and Manocha, D. (2008). “Reciprocal velocity obstacles for real-time multi-agent navigation,” in *2008 IEEE international conference on robotics and automation (Icra)*, 1928–1935.
- Van Den Berg, J., Guy, S. J., Lin, M., and Manocha, D. (2011). “Reciprocal n-body collision avoidance,” in *Robotics research: the 14th international symposium ISRR*. Springer, 3–19.
- Van Den Berg, J., Guy, S. J., Snape, J., Lin, M., and Manocha, D. (2025). Rvo2 library. Available online at: <https://gamma.cs.unc.edu/RVO2>.
- van Toll, W., Grzeskowiak, F., Gandía, A. L., Amirian, J., Berton, F., Bruneau, J., et al. (2020). “Generalized microscopic crowd simulation using costs in velocity space,” in *Symposium on interactive 3D graphics and games*, 1–9.
- Vasquez, D., Okal, B., and Arras, K. O. (2014). “Inverse reinforcement learning algorithms and features for robot navigation in crowds: an experimental comparison,” in *2014 IEEE/RSJ international conference on intelligent robots and systems (IEEE)*, 1341–1346.
- Vora, S., Lang, A. H., Helou, B., and Beijbom, O. (2020). “Pointpainting: sequential fusion for 3d object detection,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 4604–4612.
- Vouros, G. A. (2022). Explainable deep reinforcement learning: state of the art and challenges. *ACM Comput. Surv.* 55, 1–39. doi:10.1145/3527448
- Vuong, A. D., Nguyen, T. T., Vu, M. N., Huang, B., Nguyen, D., Binh, H. T. T., et al. (2023). “Habicrowd: a high performance simulator for crowd-aware visual navigation,” in *arXiv preprint arXiv:2306.11377*.
- Wang, M., and Deng, W. (2018). Deep visual domain adaptation: a survey. *Neurocomputing* 312, 135–153. doi:10.1016/j.neucom.2018.05.083
- Wang, Y., He, H., and Sun, C. (2018a). Learning to navigate through complex dynamic environment with modular deep reinforcement learning. *IEEE Trans. Games* 10, 400–412. doi:10.1109/tg.2018.2849942
- Wang, X., Girshick, R., Gupta, A., and He, K. (2018b). “Non-local neural networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7794–7803.
- Wang, Q., Chen, M., Nie, F., and Li, X. (2018c). Detecting coherent groups in crowd scenes by multiview clustering. *IEEE Trans. pattern analysis Mach. Intell.* 42, 46–58. doi:10.1109/tpami.2018.2875002
- Wang, S., Jiang, H., and Wang, Z. (2021). “Resilient navigation among dynamic agents with hierarchical reinforcement learning,” in *Advances in computer graphics: 38th computer graphics international conference, CGI 2021, virtual event, September 6–10, 2021, proceedings* 38. Springer, 504–516.
- Wang, R., Wang, W., and Min, B. C. (2022a). “Feedback-efficient active preference learning for socially aware robot navigation,” in *2022 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 11336–11343.
- Wang, X., Fu, C., Li, Z., Lai, Y., and He, J. (2022b). Deepfusionmot: a 3d multi-object tracking framework based on camera-lidar fusion with deep association. *IEEE Robotics Automation Lett.* 7, 8260–8267. doi:10.1109/ra.2022.3187264
- Wang, J., Chan, W. P., Carreno-Medrano, P., Cosgun, A., and Croft, E. (2022c). “Metrics for evaluating social conformity of crowd navigation algorithms,” in *2022 IEEE international conference on advanced robotics and its social impacts (ARSO)*. IEEE, 1–6.
- Wang, W., Wang, R., Mao, L., and Min, B. C. (2023a). Navistar: socially aware robot navigation with hybrid spatio-temporal graph transformer and preference learning. *IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, 11348–11355.
- Wang, L., Zhang, X., Qin, W., Li, X., Gao, J., Yang, L., et al. (2023b). Camo-mot: combined appearance-motion optimization for 3d multi-object tracking with camera-lidar fusion. *IEEE Trans. Intelligent Transp. Syst.* 24, 11981–11996. doi:10.1109/tits.2023.3285651
- Wang, W., Obi, I., Bera, A., and Min, B. C. (2024). Unifying large language model and deep reinforcement learning for human-in-loop interactive socially-aware navigation. *arXiv preprint arXiv:2403.15648*.
- Weng, X., Wang, J., Held, D., and Kitani, K. (2020). Ab3dmot: a baseline for 3d multi-object tracking and new evaluation metrics. *arXiv preprint arXiv:2008.08063*.
- Wijmans, E., Kadian, A., Morcos, A., Lee, S., Essa, I., Parikh, D., et al. (2019). Ddppo: learning near-perfect pointgoal navigators from 2.5 billion frames. *arXiv preprint arXiv:1911.00357*.
- Wijmans, E., Savva, M., Essa, I., Lee, S., Morcos, A. S., and Batra, D. (2023). Emergence of maps in the memories of blind navigation agents. *AI Matters* 9, 8–14. doi:10.1145/3609468.3609471
- Wojke, N., Bewley, A., and Paulus, D. (2017). “Simple online and realtime tracking with a deep association metric,” in *2017 IEEE international conference on image processing (ICIP)* (IEEE), 3645–3649.
- Wu, W., Chang, T., Li, X., Yin, Q., and Hu, Y. (2024). Vision-language navigation: a survey and taxonomy. *Neural Comput. Appl.* 36, 3291–3316. doi:10.1007/s00521-023-09217-1
- Xiang, F., Qin, Y., Mo, K., Xia, Y., Zhu, H., Liu, F., et al. (2020). “Sapien: a simulated part-based interactive environment,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11097–11107.
- Xie, Z., and Dames, P. (2023). Drl-vo: learning to navigate through crowded dynamic scenes using velocity obstacles. *IEEE Trans. Robotics* 39, 2700–2719. doi:10.1109/tro.2023.3257549
- Xie, L., Wang, S., Rosa, S., Markham, A., and Trigoni, N. (2018). “Learning with training wheels: speeding up training with a simple controller for deep reinforcement learning,” in *2018 IEEE international conference on robotics and automation (ICRA)* (IEEE), 6276–6283.
- Xu, B., and Chen, Z. (2018). “Multi-level fusion based 3d object detection from monocular images,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2345–2353.
- Xu, M., Xie, X., Lv, P., Niu, J., Wang, H., Li, C., et al. (2019a). Crowd behavior simulation with emotional contagion in unexpected multihazard situations. *IEEE Trans. Syst. Man, Cybern. Syst.* 51, 1–15. doi:10.1109/tsmc.2019.2899047
- Xu, Y., Zhou, X., Chen, S., and Li, F. (2019b). Deep learning for multiple object tracking: a survey. *IET Comput. Vis.* 13, 355–368. doi:10.1049/iet-cvi.2018.5598
- Yan, Z., Duckett, T., and Bellotto, N. (2017). *Online learning for human classification in 3d lidar-based tracking*. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, 864–871.
- Yan, Y., Mao, Y., and Li, B. (2018). Second: sparsely embedded convolutional detection. *Sensors* 18, 3337. doi:10.3390/s18103337
- Yan, Z., Schreiberhuber, S., Halmetschlager, G., Duckett, T., Vincze, M., and Bellotto, N. (2020). Robot perception of static and dynamic objects with an autonomous floor scrubber. *Intell. Serv. Robot.* 13, 403–417. doi:10.1007/s11370-020-00324-9
- Yang, G. S., Chen, E. K., and An, C. W. (2004). “Mobile robot navigation using neural q-learning,” *Proc. 2004 Int. Conf. Mach. Learn. Cybern. (IEEE Cat. No. 04EX826)*, 1, 48–52.
- Yang, B., Luo, W., and Urtasun, R. (2018a). “Pixor: real-time 3d object detection from point clouds,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7652–7660.
- Yang, B., Liang, M., and Urtasun, R. (2018b). “Hdnet: exploiting hd maps for 3d object detection,” in *Conference on robot learning (PMLR)*, 146–155.
- Yang, D., Li, L., Redmill, K., and Özgüner, Ü. (2019). Top-view trajectories: a pedestrian dataset of vehicle-crowd interaction from controlled experiments and crowded campus. *IEEE Intell. Veh. Symp. (IV)* (IEEE), 899–904. doi:10.1109/ivs.2019.8814092
- Yang, F., Wang, C., Cadena, C., and Hutter, M. (2023). Iplanner: imperative path planning. *arXiv preprint arXiv:2302.11434*.
- Yao, X., Zhang, J., and Oh, J. (2019). Following social groups: socially compliant autonomous navigation in dense crowds. *arXiv preprint arXiv:1911.12063*.
- Yao, J., Zhang, X., Xia, Y., Wang, Z., Roy-Chowdhury, A. K., and Li, J. (2024). Sonic: safe social navigation with adaptive conformal inference and constrained reinforcement learning. *arXiv Prepr. arXiv:2407.17460*.
- Yen, G. G., and Hickey, T. W. (2004). Reinforcement learning algorithms for robotic navigation in dynamic environments. *ISA Trans.* 43, 217–230. doi:10.1016/s0019-0578(07)60032-9
- Yi, S., Li, H., and Wang, X. (2016). “Pedestrian behavior understanding and prediction with deep neural networks,” in *Computer Vision—ECCV 2016: 14th European conference, Amsterdam, the Netherlands, October 11–14, 2016, proceedings, part I* 14. Springer, 263–279.
- Yoon, K., Song, Y., and Jeon, M. (2018). Multiple hypothesis tracking algorithm for multi-target multi-camera tracking with disjoint views. *IET Image Process.* 12, 1175–1184. doi:10.1049/iet-ipr.2017.1244
- Yu, F., Chen, H., Wang, X., Xian, W., Chen, Y., Liu, F., et al. (2020). “Bdd100k: a diverse driving dataset for heterogeneous multitask learning,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2636–2645.
- Yuan, W., Duan, J., Blukis, V., Pumacay, W., Krishna, R., Murali, A., et al. (2024). Robopoint: a vision-language model for spatial affordance prediction for robotics. *arXiv Prepr. arXiv:2406.10721*.
- Zakharov, S., Kehl, W., and Ilic, S. (2019). “Deceptionnet: network-driven domain randomization,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 532–541.
- Zhang, J., Springenberg, J. T., Boedecker, J., and Burgard, W. (2017). “Deep reinforcement learning with successor features for navigation across similar environments,” in *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)* (IEEE), 2371–2378.

- Zhang, K., Yang, Z., and Başar, T. (2021). "Multi-agent reinforcement learning: a selective overview of theories and algorithms," in *Handbook of reinforcement learning and control*, 321–384.
- Zhang, Y., Sun, P., Jiang, Y., Yu, D., Weng, F., Yuan, Z., et al. (2022). "Bytetrack: multi-object tracking by associating every detection box," in *European conference on computer vision*. Springer, 1–21.
- Zhang, B., Holloway, C., and Carlson, T. (2023). "Reinforcement learning based user-specific shared control navigation in crowds," in *2023 IEEE international conference on systems, man, and cybernetics (SMC) (IEEE)*, 4387–4392.
- Zheng, W., Tang, W., Jiang, L., and Fu, C. W. (2021). "Se-ssd: self-ensembling single-stage object detector from point cloud," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 14494–14503.
- Zhou, Y., and Tuzel, O. (2018). "Voxelnet: end-to-end learning for point cloud based 3d object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4490–4499.
- Zhou, B., Wang, X., and Tang, X. (2012). "Understanding collective crowd behaviors: learning a mixture model of dynamic pedestrian-agents," in *2012 IEEE conference on computer vision and pattern recognition (IEEE)*, 2871–2878.
- Zhou, C., Huang, B., and Fränti, P. (2022). A review of motion planning algorithms for intelligent robots. *J. Intelligent Manuf.* 33, 387–424. doi:10.1007/s10845-021-01867-z
- Zhou, Z., Ren, J., Zeng, Z., Xiao, J., Zhang, X., Guo, X., et al. (2023). A safe reinforcement learning approach for autonomous navigation of mobile robots in dynamic environments. *CAAI Trans. Intell. Technol.*, cit2.12269. doi:10.1049/cit2.12269
- Zhu, W., and Hayashibe, M. (2022). A hierarchical deep reinforcement learning framework with high efficiency and generalization for fast and safe navigation. *IEEE Trans. industrial Electron.* 70, 4962–4971. doi:10.1109/tie.2022.3190850
- Zhu, K., and Zhang, T. (2021). Deep reinforcement learning based mobile robot navigation: a review. *Tsinghua Sci. Technol.* 26, 674–691. doi:10.26599/tst.2021.9010012
- Zhu, K., Li, B., Zhe, W., and Zhang, T. (2022). Collision avoidance among dense heterogeneous agents using deep reinforcement learning. *IEEE Robotics Automation Lett.* 8, 57–64. doi:10.1109/lra.2022.3222989
- Zhu, K., Xue, T., and Zhang, T. (2025). Confidence-aware robust dynamical distance constrained reinforcement learning for social robot navigation. *IEEE Trans. Automation Sci. Eng.* 22, 16572–16590. doi:10.1109/tase.2025.3578326
- Ziebart, B. D., Maas, A. L., Bagnell, J. A., and Dey, A. K. (2008). Maximum entropy inverse reinforcement learning. *Aaai* 8, 1433–1438.
- Zou, Z., Chen, K., Shi, Z., Guo, Y., and Ye, J. (2023). Object detection in 20 years: a survey. *Proc. IEEE* 111, 257–276. doi:10.1109/jproc.2023.3238524