Article

Evolution of Socially-Aware Robot Navigation

Silvia Guillén-Ruiz [ID], Juan Pedro Bandera [ID], Alejandro Hidalgo-Paniagua [ID] and Antonio Bandera *[ID]

Departamento de Tecnología Electrónica, ETSI Telecomunicación, University of Málaga, 29010 Malaga, Spain
* Correspondence: ajbandera@uma.es

Abstract: In recent years, commercial and research interest in service robots working in everyday environments has grown. These devices are expected to move autonomously in crowded environments, maximizing not only movement efficiency and safety parameters, but also social acceptability. Extending traditional path planning modules with socially aware criteria, while maintaining fast algorithms capable of reacting to human behavior without causing discomfort, can be a complex challenge. Solving this challenge has involved the development of proactive systems that take into account cooperation (and not only interaction) with the people around them, the determined incorporation of approaches based on Deep Learning, or the recent fusion with skills coming from the field of human–robot interaction (speech, touch). This review analyzes approaches to socially aware navigation and classifies them according to the strategies followed by the robot to manage interaction (or cooperation) with humans.

Keywords: socially aware robotics; human motion prediction; deep learning; multi-behaviour navigation; social navigation

1. Introduction

Society 5.0 represents a new paradigm in which people and artificial beings cooperate in routines, environments, and the interactions of everyday life [1]. This cooperation is intended to be natural and intuitive, and the new artificial actors are expected to behave appropriately. Service robots are one of the technologies with the greatest number of potential applications in this new social paradigm [2]. They are also one of the most profoundly affected by the new technical and contextual complexities arising from these new requirements [3]. Service robots, being inevitably social when used in everyday life scenarios, face the most challenging technical, ethical, social, and legislative demands. In particular, Socially Aware Robotics (SAWR) is an emerging area of research that seeks to understand how cognitive robots can be aware of their social context and use this capability to behave as more accessible, accepted, and useful devices, being able to establish more appropriate and effective interactions to assist humans [4]. A robot aiming to exploit socially enhanced autonomous capabilities needs to perceive its environment and reach a certain level of understanding of its context. However, to be truly socially aware, a robot must not only react intentionally to perceived changes, but it must also be able to predict or learn what the behavior of the humans surrounding it will be, anticipating consequences and selecting the best possible and most comprehensible action, while respecting social conventions [5,6].

One of the essential capabilities of most robotic solutions, more deeply affected by these new social requirements, is navigation. Service robots working in daily life environments cannot just search for the shortest collision-free path. They should also solve a multivariable optimization problem that considers, for example, human comfort and social rules. As a result, traditional navigation approaches are no longer adequate due to their limited flexibility, and new proposals arise. The growing importance of the topic has given rise to several review articles analyzing these proposals in different ways. Concepts regarding the human factor were highlighted in the survey papers on proxemics ([7,8]),

and semantic and social mapping ([9]). The review paper by Gao and Huang [10] focused on the evaluation methods, scenarios, datasets, and metrics commonly used In Previous socially-aware navigation research. Zhu and Zhang [11] reviewed Deep Reinforcement Learning (DRL) methods and DRL-based navigation frameworks. They differentiated between several typical application scenarios: local obstacle avoidance, indoor navigation, multi-robot navigation, and social navigation. Chik et al. [12] focused on robot navigation as a hierarchical task, involving a collection of sub-problems that can range from the high-level decision to reactive avoidance of low-level obstacles. The review highlighted how the whole navigation stack should evolve to address the problem of dealing with dynamic human environments, including human detection, tracking, and predictive modeling, at a more local level, and considering social costs at a higher level.

The paper by Kruse et al. [13] discusses human-aware navigation. In this paper, the authors state that '*the robotics and human–robot interaction (HRI) communities have not yet produced a holistic approach to human-aware navigation*' [13], even though they identified 77 citations between 1995 and 2012 closely related to this topic. They also defined two axes on which to classify these papers. First, they established four categories: comfort, naturalness, sociability, and others. In the second axis, the technologies on which the articles focus are considered. The categories in this second axis are pose selection, global planning, behavior selection, and local planning.

The present survey extends the previous work of Kruse et al. [13] in two dimensions. On the one hand, it updates the analysis considering new approaches and research conducted in the last ten years in the field of social navigation. On the other hand, this survey focuses on those methods in which the robot modifies its behavior in the presence of another mobile agent, or traces a path (e.g., avoiding disturbing a group of people talking to each other). Hence, it differs from the work of Kruse et al. [13] as the basis of our categorization will not be functionality, but the degree to which the method employed for reacting in the presence of humans is able to learn from previous observations or predict the near future. Prediction and Learning will be the terms that will guide the literature search, establishing subgroups in the more global group of social navigation. Moreover, this survey would like to draw attention to papers that focus on social comfort and explore how social navigation methods are evolving to equip robots with more human-like skills. As robots become increasingly similar to humans, understanding the complexities of social norms and adapting to them becomes more important. These papers shed light on how robots can be designed to better interact with humans in social situations.

The rest of the paper is organized as follows: Section 2 explains the literature review process. Results are presented in Section 3. Discussion is provided in Section 4. Finally, conclusions and future work are drawn in Section 5.

2. Methodology

This section describes the criteria used both to select the set of articles considered for this review article (Section 2.1) and to organize them into different groups (Section 2.2).

2.1. Article Selection Criteria

We carefully curated a collection of papers on the topics of social navigation, with a special interest in two keywords, prediction and learning. Our selection process involved a thorough review of the literature, drawing from the works of Kruse et al. [13], Chik et al. [12], Gao and Huang [10] and Zhu and Zhang [11], and extended with recent citations. We selected the most relevant papers that explored the areas of social navigation, comfort, prediction, and learning. These papers were further narrowed down by evaluating the quality and relevance of the references cited in each article. Finally, we prioritized the papers that were most frequently cited in previous works.

To gain insight into the number of papers published on our focus topics, we can refer to the graphs in Figures 1–3. These graphics were generated using data from the Web of Science (https://clarivate.com/webofsciencegroup/solutions/web-of-science/, accessed

on 13 February 2023). For the topic (TS = (social AND navigation)) AND TS = (robot), we obtain 723 results in the range from 1994 to 2022 (data from the current year, 2023, were not included as the year has not finished and these data could disturb the statistics). Adding TS = (learning), the number of citations reduces to 207 results (Figure 3). With TS = (prediction), the number is reduced to 80 results (Figure 2). From this large dataset, we have covered in this survey 100 papers. Figure 4 compares the papers covered by the survey by Kruse et al. [13] and the current review.
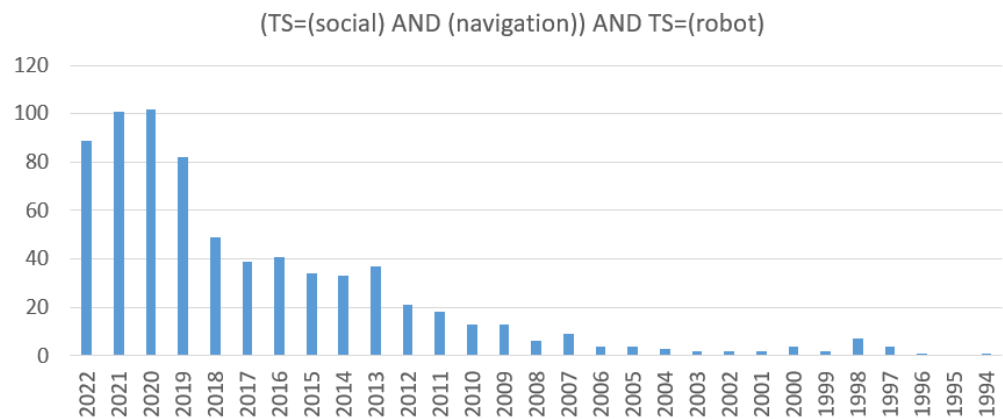

Figure 1. Web of Science Analyze filter: (TS = (social AND navigation)) AND TS = (robot).
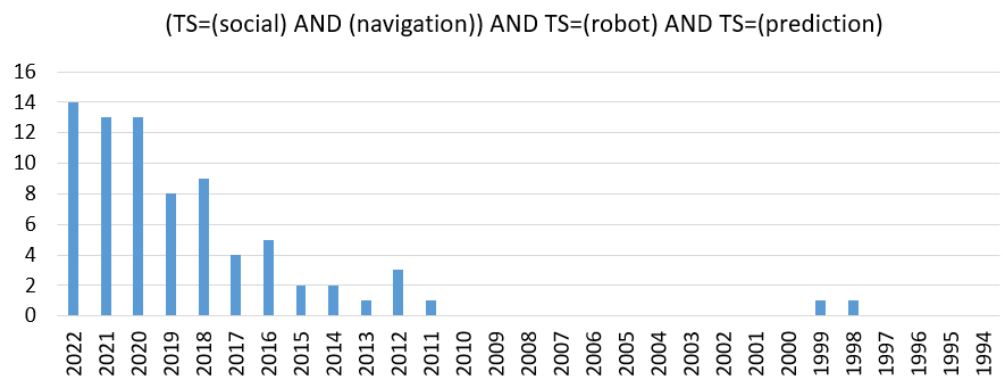

Figure 2. Web of Science Analyze filter: (TS = (social AND navigation)) AND TS = (robot) AND TS = (prediction).
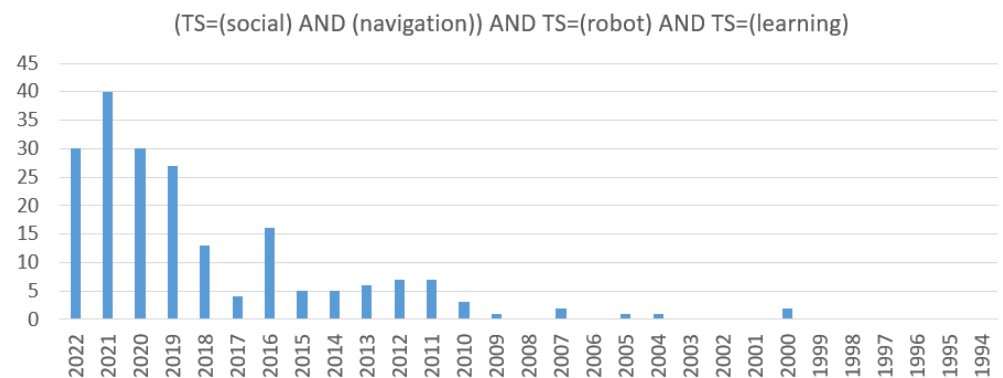

Figure 3. Web of Science Analyze filter: (TS = (social AND navigation)) AND TS = (robot) AND TS = (learning).
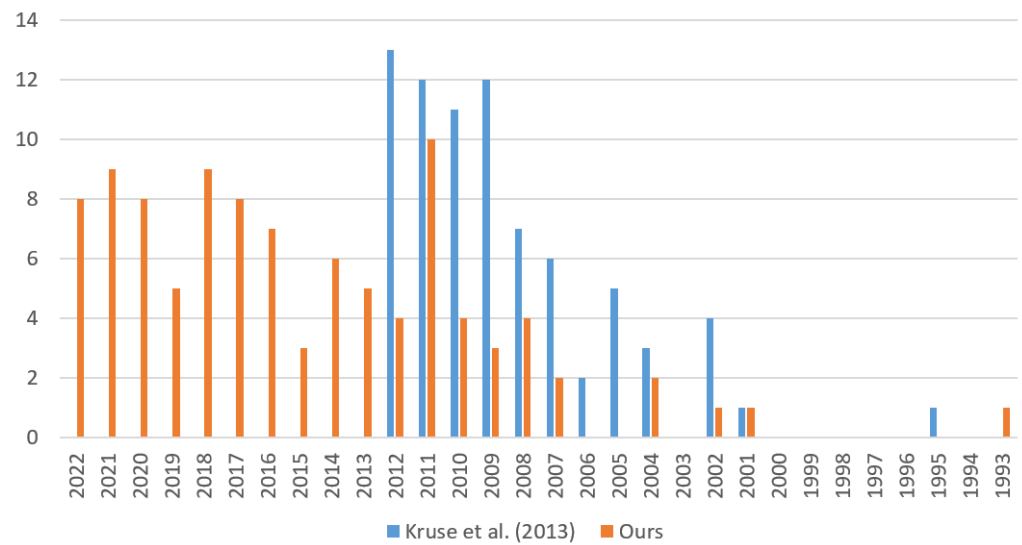
Figure 4. Distribution of publications over years considered by the review paper by Kruse et al. [13] and the current survey.

2.2. Article Classification Criteria

As mentioned above, the goal of socially aware navigation is not only to find a collision-free path from a starting point to a destination. This navigation process also requires carefully taking into account the movement of people around the robot, and the interactions processes between the robot and these people. When the number of people is small, few interactions are required to avoid possible collisions. People tend to move along rectilinear trajectories over long periods of time. However, when the density of people increases, problems arise [14]. In these scenarios, people have to frequently change their motion states (direction of travel, but also velocity or acceleration) in real time, to avoid collisions while trying to reach their destinations. Linear models are no longer correct for modeling interactions (human–robot, but also human–human). The perception of the robot's movement by others becomes particularly relevant. Thus, in addition to being safe, it is important that the motion of the robot is legible, allowing people in the vicinity of the robot to easily understand its movement intention [15].

Analyzing the evolution of socially aware robot navigation addresses two major topics: (i) evaluating how robot social skills have made human interactions more comfortable and natural; (ii) evaluating how algorithms solve navigation in crowded scenarios. Following on previous work from several researchers on robot navigation in dense crowds, we will group in this survey works on robot navigation into three categories [14,16]: (1) reactive approaches; (2) proactive approaches; and (3) learning-based approaches. Moreover, we added a fourth category: (4) multi-behaviour navigation approaches. In *reactive* approaches, the robot reacts to other mobile agents through *one-step look ahead* strategies [14]. These approaches are typically very efficient (e.g., the social force model [17]). *Proactive* approaches predict the behavior of the human and then plan a suitable path. Predictions can be based on reasoning (assumptions of how agents behave in general), or learning (justified by observations of how agents behave) [13]. Strategies for prediction can deal with human motion [18] or intentions [14]. *Learning-based* approaches aim towards the robot learning the navigation policy, and adapting it to the target scenario. Deep reinforcement learning (DRL) has been extensively employed for solving this problem [11,19]. Finally, we added *multi-behaviour* navigation approaches as a separate subsection. Significantly, these methods address the question of whether it can be useful to consider interaction actions, such as touching, gesturing, or speaking, for the sake of allowing robots to navigate in dynamic, crowded environments. The importance of being able to coordinate different robotic functionalities (navigation, dialogue) to solve a navigation goal will grow in the coming years as service robots actually share the environment with people.

3. Socially Aware Navigation Approaches

This section develops the proposed taxonomy for classifying socially aware robotic navigation methods. A table illustrating the methods used as examples of each category in a concise manner is included at the end of each subsection.

3.1. Reactive Approaches

Robots that navigate in a dynamic or unknown environment can use path planners based on heuristic grid search, or a sampling-based approach, to obtain high-level routes. However, while these path planners can provide low-risk routes [20], they must always be combined with a reactive collision avoidance system. This reactive navigation layer must operate at a high frequency, to guarantee fast enough obstacle detection and avoidance. Moreover, the presence of different mobile agents in the environment (people or other autonomous vehicles) makes this reactive, local navigation scheme a basic element for maintaining a safe environment. These two factors (speed requirements and safety issues) set the onboard processor of the robot as the adequate device to be executing this reactive layer.

This section collects articles that deal with reactive, local navigation when there are other mobile agents in the environment. Being reactive approaches, they may not take into consideration either prediction or learning. Table 1 assesses the approaches covered in this section. Most of these methods are based on the concept of Velocity Obstacle (VO), Artificial Potential Fields (APF), Vector Field Histogram (VFH), or Social Force Model (SFM). The VO concept was introduced by Fiorini and Shiller [17], and has been widely used to ensure safe navigation (e.g., Shiller et al. [21], Kluge and Prassler [22], and Fulgenzi et al. [23]). The VO concept of a moving obstacle A_j to an agent R_i is the set of all those velocities for R_i that will result in a collision, at some moment in time, with the obstacle. Hence, in each planning cycle, the agent should choose a velocity different from any of the VOs induced by the moving obstacles (see Figure 5 (Right)). The APF is a virtual force field approach initially proposed by Khatib [24]. Briefly, the robot's motion is controlled by a gravity force, generated by the target to reach, and repulsive forces, generated by the obstacles. Yao et al. [25] proposed improving the traditional APF with reinforcement learning in order to deal with dynamic scenarios. The VFH approach uses a two-dimensional Cartesian histogram grid as a world model, which is updated continuously with data sampled by onboard sensors [26]. Babinec et al. [27] modify the VFH* to deal with both static and moving obstacles. As a relevant disadvantage, it cannot handle non-linear motion.

Table 1. Approaches considered in Section 3.1. The table covers the methods they use for solving the local navigation problem.

| Reference	Methods
Fiorini and Schiller [17]	Velocity Obstacle (VO)
Shiller et al. [21]	Non-Linear Velocity Obstacle (NLVO)
Kluge and Prassler [22]	Recursive Probabilistic Velocity Obstacles
Fulgenzi et al. [23]	Probabilistic Velocity Obstacle (PVO)
	Dynamic occupancy grid provided by a general sensor system.
Palm and Driankov [28]	Velocity potential of an incompressible fluid
Babinec et al. [27]	Vector Field Histogram (VFH*)
Yao et al. [25]	BHPF. Calibrated using the Black-hole potential field deep Q-learning (BHDQN)
Zheng et al. [29]	Improved Social Force Model Based on Emotional Contagion and Evacuation Assistant
Reddy et al. [30]	Extend Social Force Model to incorporate the social cues by adding new social forces
	Extends the Geometric approach to incorporate the social cues by selecting the geometric gap as per the social reference
	Hybrid approach combining the social potential field and geometric method
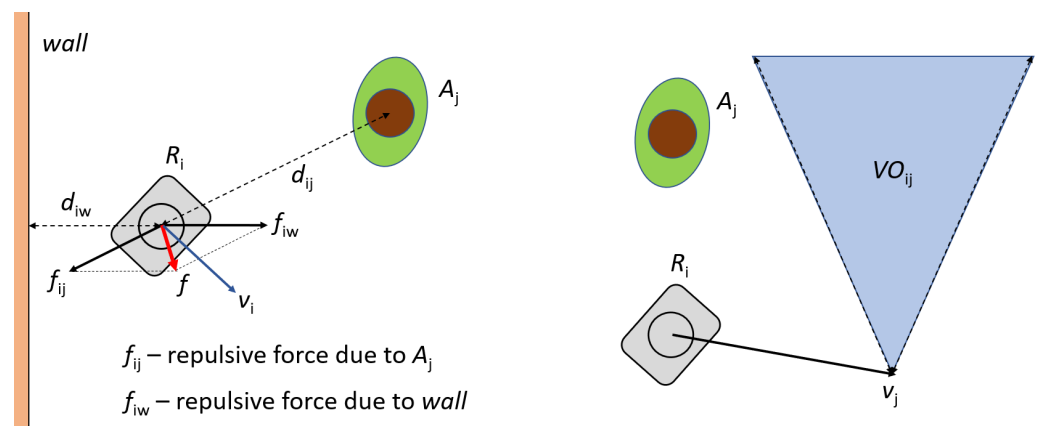
Figure 5. (**Left**) Forces of SFM. The resulting force f in the robot R_i is provoked by a static obstacle (the wall, f_{iw}) and a mobile one A_j (f_{ij}). (**Right**) The velocity obstacle VO_{ij} for a robot R_i induced by a mobile agent A_j, with velocity v_j.

The SFM simulates agent dynamics using interaction forces (Figure 5 (Left)). It allows expressing the collision-avoiding behavior through a function. The inputs for this function are the relative and absolute positions and velocities of the mobile agents [31]. The SFM is a powerful approach for crowd navigation, as discussed in the next section, which presents several types of modified SFMs proposed for robot navigation and human modeling. For instance, Zheng et al. [29] propose an SFM based on emotional contagion for evacuation assistant (ecaSFM). Reddy et al. [30] propose a novel hybrid approach extending the SFM with geometric constraints. This proposal maintains the proactive nature of the geometric approach and retains the reactive nature of the SFM.

Finally, it is interesting to include in this list the proposal by Palm and Driankov [28], as a representative example of reactive proposals not based on the previous approaches. Palm and Driankov [28] use the behavior of a fluid in the presence of obstacles as a simile. Then, they propose a local navigation method in which a set of streamlines is continuously updated. The method is applicable in an environment where there are other mobile elements such as robots or people.

3.2. Proactive Approaches

Purely reactive methods are not very common in social navigation. When navigating between people, it is interesting for the robot to consider these people as cooperative partners. Hence, the robot can interact with people to jointly avoid collisions [32]. This cooperative approach requires the robot (i) to predict the behavior (i.e., trajectories) of the people, and then (ii) proactively plan the path to follow. There is a multitude of methods to predict the behavior of other mobile agents that share the environment with the robot: from those that assume certain assumptions about how these agents behave in general (reasoning-based prediction), to those that are justified on the basis of observations of how these agents behave, usually in a specific environment or under specific conditions (learning-based techniques) [13]. Subsequent path planning methods include, among others, modified versions of the SFM [33] or the VO concept [34]. The next subsections describe the state-of-art of predictive models and path-planning algorithms for proactive approaches.

3.2.1. Predictive Models

Table 2 lists the approaches described in this section. Regarding predictive models, initial efforts to model the behavior of other mobile agents, and thus avoid routes that could lead to collisions, worked with highly deterministic models of motion. No weight was given to uncertainty, and they typically considered a multi-robot framework. In two proposals (van den Berg et al. [35], Snape et al. [34]) the concept of Velocity Obstacle was modified to deal with a scenario where several robots coexist, but where each of them

works independently. Assuming that all robots employ the same technique for navigation, the aim of the Reciprocal Velocity Obstacle (RVO) [35] is to avoid the oscillatory behavior they may adopt when approaching or crossing another robot. The Hybrid Reciprocal Velocity Obstacle [34] extends the previous approach, setting priorities in the interaction between robots.

Many methods suggest modifications to the SFM [36–38]. The aim of all these approaches is to map out a collision-free, smooth route, but one that takes into account the presence of mobile agents—people or other autonomous vehicles. To achieve this, they include some interesting techniques. For instance, Zanlungo et al. [36] assume that deviations of the robot from the straight path leading to its goal are only due to avoidance of collisions with other moving agents, but they predict the time in the future at which the relative distances to each approaching agent will be at a minimum. Then, they assume that forces at that time depend on the distance between the mobile agents (circular specification, CS). Figure 6 schematizes the situation where two approaching agents interact. Forces will be circular symmetric forces as those used in CS, but based on this future situation, which is assumed to be the most interesting for the agent since it is when a collision can happen. Shiomi et al. [38] use a specific version of the SFM, called CP-SFM, to simulate human-like collision-avoidance behavior in robots for low-density environments, such as shopping mall corridors. Paths generated by the planner maintain a social distance from people and respect their personal space.
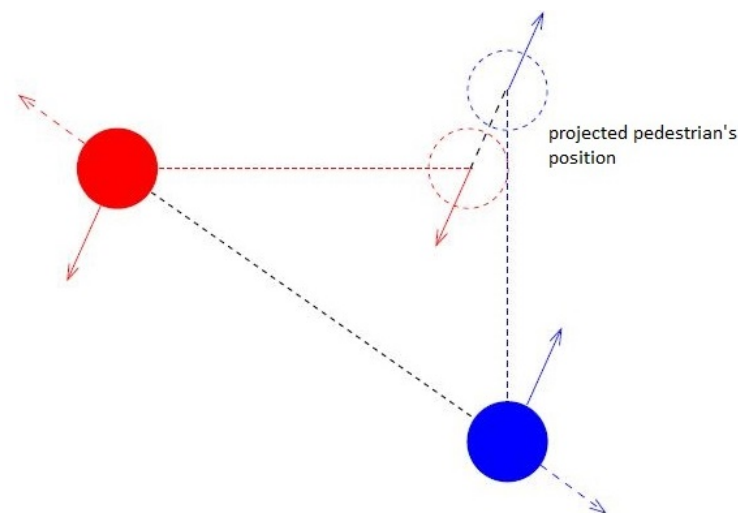

Figure 6. SFM with explicit collision prediction [36]: the position of the red and blue colored agents is projected at the time where the minimum distance between them is expected (dashed circles). For avoiding that situation, the agents will accelerate with the acceleration vectors denoted by the continuous arrows (drawn both at the present time positions and at the predicted positions). The dashed arrows (only drawn at the present time positions) are the acceleration vectors if the CS model is applied without prediction. Permission granted by Editorial EPL (Europhysics Letters).

Engines, such as the Kalman filter, could be used to predict the position of mobile agents surrounding the robot by incorporating the uncertainty dimension. The problem is that, when there are several agents, the use of such engines leads to an *uncertainty explosion* [39], which can make it impossible for the robot to find a safe path to a target. To try to control this growth in uncertainty, different models of human motion have been proposed [40,41]. For instance, Du Toit and Burdick [42] propose directly limiting the predictive uncertainty of each individual agent. Joseph et al. [43] describe a more complex motion model: a Gaussian process mixture with a Dirichlet Process prior over mixture weights. This non-parametric model presents a high computational cost. The problem is solved in the RR-GP algorithm [44], a clustering-based trajectory prediction solution that uses Bayesian non-parametric reachability trees to improve the Gaussian prediction.

Although for dealing with the position uncertainty, more sophisticated individual models are proposed, this line of research does not consider agent interaction in their models [39].

As previously mentioned, the uncertainty about the person's future behavior makes it difficult for a robot to determine the appropriate actions for avoiding collisions. In addition, non-collision can only be a partial solution, and the robot can be asked to avoid situations where its behavior disturbs people, for example, due to excessive proximity. In a scenario where the robot shares the environment with several people, instead of formulating the behavior of each person, the robot must be able to predict how each person's behavior will vary over time. Common approaches for predicting trajectories, such as the Kalman filter or Particle filters, exhibit different disadvantages for our scenario [45], being more adequate to those methods that consider in the model a goal-based policy. Thus, they assume that human behavior is captured in the previously observed trajectories, and the problem is then to determine to which group of trajectories the current one belongs. For instance, Bennewitz et al. [46] use models of human motion patterns, which can be learned using an Expectation-Maximization (EM) algorithm. With this information, the robot can predict where the person is or where s/he will be. Ziebart et al. [45] proposed using maximum entropy inverse optimal control to model the goal-directed trajectories of people. Maximum entropy is also used by Kuderer et al. [47]. Representative features of the people's trajectories are analyzed to find the probability distributions that drive their navigation behaviors. Several prediction functions using the minimum curvature variation concept were described in Ferrer and Sanfeliu [48]. Built over the Curvature Length Predictor (CLP), the wCLP weights an average of previous predictions in a limited time window. Experiments showed that it can quickly recognize the new intention of motion when the destination changes or unexpected behaviors happen. Kabtoul et al. [49] proposed using a quantitative time-varying function to model the human–robot cooperation in an interaction scenario. Using this cooperation estimation, the human motion trajectory is predicted by a cooperation-based trajectory planning model. Figure 7 provides an overview of the approach proposed by Ikeda et al. [50]. The approach performs an offline analysis for estimating sub-goals (and a probabilistic transition model) in the environment (Figure 8). Then, at run-time, the approach allows estimating the future positions of people based on the sequence of previously traversed sub-goals and the current velocity.
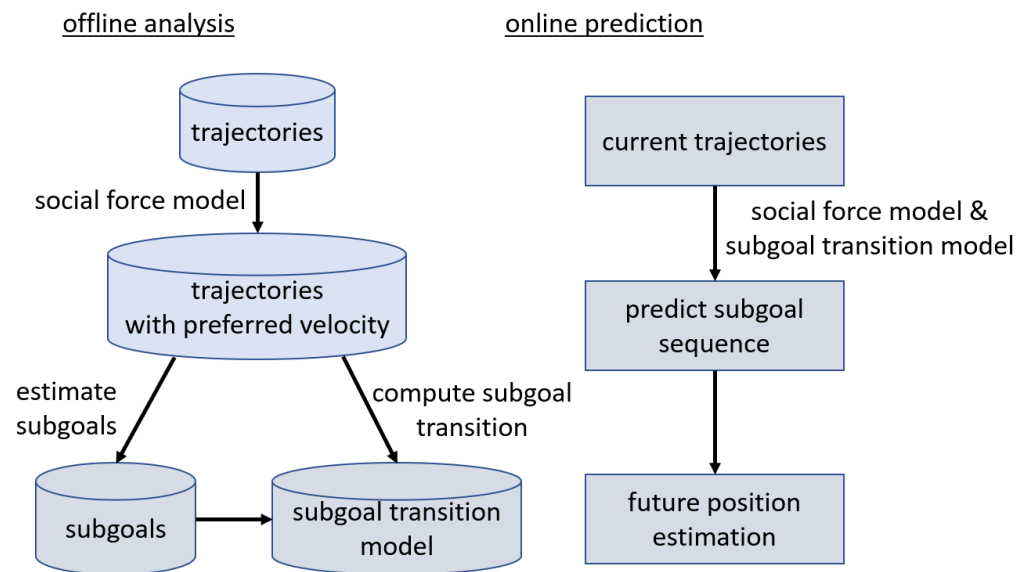

Figure 7. Overview of the approach by Ikeda et al. [50]. Image granted by author.
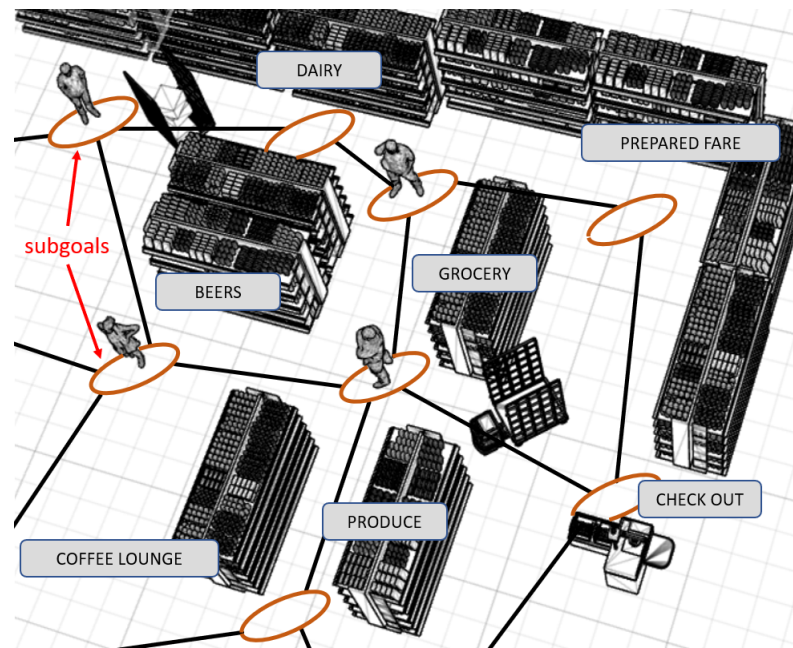
Figure 8. Examples of a distribution of sub-goals in an environment [50].

Instead of predicting trajectories, Luber et al. [51] take advantage of the place-dependency of human behaviour for building a spatial affordance map. The problem of learning this spatial model of human behavior is posed as a parameter estimation problem of a non-homogeneous spatial Poisson process. The spatial affordance map is learned using Bayesian inference from observations of track creation, matching, and false alarm events, gained by introspection of a laser-based multi-hypothesis people tracker. The framework is described for people detection and tracking, but it was for instance integrated into the navigation scheme proposed by Ferrer et al. [52].

With the aim of taking into consideration human–robot interaction, several authors have demonstrated that the same proxemic zones that exist in human–human interaction can be useful to explain human–robot interaction scenarios [53–55]. Sisbot et al. [15] proposed including in the model safety- and visibility-related criteria to control the distance between the robot and human and keep the robot mostly in the human's field of view. The safety, visibility, and hidden zone grids are combined into a single grid to find the most cost-efficient path. In the work by Svenstrup et al. [56], the behavior of the robot is based on adaptive potential functions that are adjusted accordingly such that social spaces are respected. Castro-Gonzalez et al. [57] proposed a method for predicting people's positions in crossing behaviors using proxemics. The modified social force model (MSFM) [58] integrates social components (body pose, face orientation, and personal space during motion) into the classical SFM based on human position. In the MSFM, the short-term intended direction is described by body pose, and a supplementary force-related face orientation is added for intention estimation. Face orientation is employed as the best indication of the direction of personal space during motion. To endow robots with the ability for navigating dynamic human environments in a socially acceptable manner that would guarantee human comfort and safety, Truong et al. [33] proposed extending SFM with the Hybrid Reciprocal Velocity Obstacle technique. The result is the so-called proactive Social Motion Model (PSMM), which considers not only human states (position, orientation, motion, field of view, and hand poses) relative to the robot but also social interactive information about human-object and human group interactions. A survey paper describing the social concepts of proxemics theory applied in the context of human-aware autonomous navigation was provided by Rios-Martinez et al. [7], and more recently by Samarakoon et al. [8]. In general, these navigation algorithms model human–robot interaction, emphasizing on maintaining the proper safety distances, but they do not consider human–robot cooperation.

In dense crowds, a common problem of those proactive approaches that take into consideration the uncertainty in the position of humans or robots is the so-called Freezing Robot Problem (FRP) [14,39] (Figure 9). Briefly, the problem is that the robot may not be able to find even one feasible route due to the difference between the predicted and the real motion of mobile agents. To deal with this problem, the interaction of all agents with the remaining static and dynamic obstacles must be considered. This strategy is addressed by SFM-based approaches [59,60] or the so-called optimal reciprocal collision avoidance (ORCA) approaches [14,61–63]. Farina et al. [60] proposed merging the SFM with Laumond's human locomotion models. The resulting Headed SFM is able to reliably reproduce human motions both in free space and in highly crowded environments. In the proactive kinodynamic planner,Ferrer and Sanfeliu [59] propose using the Extended Social Force Model (ESFM) to simplify both the prediction model and the planning system under differential constraints. The main problem with SFM-based approaches is that parameter tuning depends on the specific scenario [64]. For computing free routes, ORCA conducts an optimized search in the feasible geometric space. In the reciprocal n-body collision avoidance [61], the problem of avoiding collisions between multiple robots is reduced to solve a low-dimensional linear program. The approach is based on the VO concept. In the GLMP (Global and Local Movement Patterns) approach [62], the characteristics of agents' motion and movement patterns are learned from 2D trajectories using Bayesian inference. Motion patterns consider local movement ones, corresponding to the current and preferred velocities, and global characteristics such as entry points and movement features. The Pedestrian Optimal Reciprocal Collision Avoidance (PORCA) proposed by Luo et al. [63] is a human motion prediction model that takes into account the human's global navigation intention and the local interactions with the robot and other people. Chen et al. [14] proposed the intention-enhanced ORCA (iORCA). The iORCA employs a naive Bayesian classifier for estimating the most possible pedestrian destination, and then it can compute the human velocity. Moreover, to deal with possible changes of intention, iORCA updates these destinations at each time step. Inspired by the RVO model, the eRVO model integrates the emotional effect into velocity decision [65]. Generally, ORCA approaches have shown to be more stable than SFM in low sampling rates and in dense crowd scenarios [65,66].
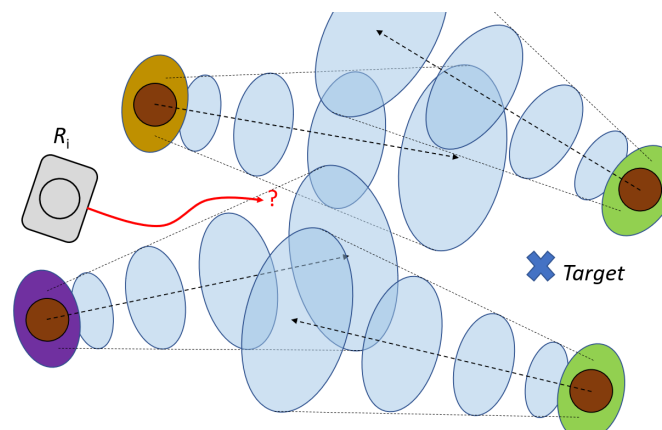
Figure 9. The Freezing Robot Problem (FRP). The robot is R_i and the ellipses represent the predictive covariance. It is not easy for the robot to find a route to the Target.

Data-driven approaches have been also proposed for capturing agent motion considering interactions with static and dynamic obstacles. Thus, inspired by the success of Long-Short Term Memory networks (LSTM) in other research tasks, Alahi et al. [67] proposed the Social-LSTM, a data-driven architecture for predicting human trajectories in future instants. For capturing the dependencies between multiple correlated sequences, a social pooling layer is introduced. This allows the associated LSTMs to spatially close sequences to share their hidden states with each other. Thus, the model can automati-

cally learn typical interactions that take place among trajectories that coincide in time. Considering that every person within a crowd implicitly cooperates with each other to avoid collisions, the Social Attention by Vemula et al. [68] captures the relative importance of each person when navigating, without emphasizing the proximity. The problem with these approaches is that they are computationally expensive, and therefore difficult to incorporate into mobile robots.

Table 2. Approaches considered in Section 3.2.1. The table covers the methods they use for modeling human motion.

Reference	Methods
van den Berg et al. [35]	Reciprocal Velocity Obstacle
Snape et al. [34]	Hybrid Reciprocal Velocity Obstacles
Zanlungo et al. [36]	SFM extended to the near future
Ferrer et al. [37]	Social Force Model (SFM)
Shiomi et al. [38]	Collision Prediction Social Force Model (CP-SFM)
Trautman et al. [39]	Multiple Goal Interacting Gaussian processes algorithm
Large et al. [40]	Velocity Obstacle (VO) & Obstacles motion prediction
Thompson et al. [41]	Probabilistic Model of Human Motion
Du Toit and Burdick [42]	Thresholding the uncertainty
Joseph et al. [43]	Gaussian processes (GP) & Dirichlet process (DP) prior over mixture weights
Aoude et al. [44]	RR-GP—Learned motion pattern model by combining the flexibility of Gaussian processes (GP) with the efficiency of RRT-Reach
Ziebart et al. [45]	Maximum entropy inverse optimal control
Bennewitz et al. [46]	Learned human motion patterns
Kuderer et al. [47]	Maximum entropy
Ferrer and Sanfeliu [48]	CLP-Time Window Predictor
Kabtoul et al. [49]	Quantitative time-varying function to model HR cooperation
Ikeda et al. [50]	Sub-goals to retrieve useful information not only for prediction but also for robot navigation, environment modeling and pedestrian simulation. Sub-goals used as the nodes of the robot global path planner, and as the nodes of the planner in the pedestrian simulator
Luber et al. [51]	Non-homogeneous spatial Poisson process Bayesian inference from observations of track creation Matching and false alarm events Gained by introspection of a laser-based multi-hypothesis people tracker
Vega et al. [53]	Adaptive Spatial Density Function Asymmetric Gaussian representation for personal space Inclusion of the space affordances Probabilistic Road Mapping (PRM) Rapidly-exploring Random Tree (RRT) Elastic band algorithm
Mead and Mataric [54]	Probabilistic framework for proxemic behavior production in HRI
Mead et al. [55]	Heuristic-Based vs. Learned Approaches
Sisbot et al. [15]	Safety and visibility related criteria
Svenstrup et al. [56]	Adaptive potential functions respecting social spaces
Castro-González et al. [57]	Hidden Markov Models for predictions
Ratsamee et al. [58]	Modified SFM (MSFM) considering body pose, face orientation and personal space
Truong and Ngo [33]	Proactive Social Motion Model (PSMM)
Ferrer and Sanfeliu [59]	Extended SFM (ESFM)
Farina et al. [60]	SFM with Laumond's human locomotion models
van den Berg et al. [61]	Reciprocal n-body collision avoidance
Bera et al. [62]	GLMP approach-Global and Local Movement Patterns Pedestrian trajectory data using Bayesian Inference Ensemble Kalman Filters (EnKF) and Expectation Maximization (EM)
Luo et al. [63]	Pedestrian Optimal Reciprocal Collision Avoidance (PORCA) combines with a Partially Observable Markov Decision Process algorithm (POMDP)
Chen et al. [14]	Interactive MPC (iMPC) framework Intention Enhanced ORCA (iORCA)
Xu et al. [65]	Emotional Reciprocal Velocity Obstacles (eRVO)
Alahi et al. [67]	Social Long-Short Term Memory networks (Social-LTSM)
Vemula et al. [68]	Social Attention S-RNN architecture

3.2.2. Navigation Strategies Using Agent Motion Models

Once a model of human motion is available, the next step in a proactive approach is to define a planner able to find an optimal navigation policy. Table 3 lists methods that aim towards this objective. Foka and Trahania [69] suggested a unified model for considering global and local planning (as well as localization). The model is a specific instantiation of the hierarchical Partially Observable Markov Decision Process (POMDP), called Robot Navigation-HPOMDP (RN-HPOMDP). The framework estimates the final destination of all mobile agents, and this information is employed for effective obstacle avoidance. Svenstrup et al. [70] proposed an algorithm for robot trajectory planning in dynamic human environments, using a potential field generated from the perceived positions and motions of people. The problem is solved using a Rapidly Exploring Random Tree (RRT) algorithm enhanced with a robot motion model and controller, and using a Model Predictive Control (MPC) approach to execute only a short segment of the planned trajectory. The method minimizes the cost of traversing the potential field, resulting in comfortable and natural robot trajectories. Dutoit and Burdick [42] describe an MPC framework for planning in a dynamic scene. As described in Section 3.2.1, the predicted motion uncertainties of both the robot and people are set as a hard threshold. Park and Kuipers [71] and Park et al. [72] proposed combining MPC and the equilibrium point control to provide a model predictive equilibrium point control (MPEPC) for a wheelchair robot navigating crowds. Taking into consideration human intention and human–robot interactions, the interactive MPC (iMPC) framework was proposed by Chen et al. [14]. The iMPC framework applies the iORCA model in the state transition function to predict human states, and extends this interactive model with robot constraints.

Based on the SFM, Ferrer et al. [37,52] put the emphasis on the design of socially aware navigation frameworks, where topics, such as human comfort and safety are of vital relevance. Ferrer et al. [52] integrated pedestrian intention and interaction into a scheme for a robot's human-awareness navigation based on the social forces concept. Their experiments show that socially aware navigation is well suited for a robot companion task in open spaces.

To respect people's personal space, while also avoiding collisions, Rios-Martinez et al. [73] propose an extension of the RRT algorithm for navigation. The Risk-RRT approach uses a Gaussian procedure learning for estimating the area occupied by the person (o-space).

Although human intentionality was predicted, in some sense, by Ikeda et al. [50], the topic is the core of the proposals by Ferrer and Sanfeliu [74] and Palm et al. [75]. The Bayesian Human Motion Intentionality Prediction (BHMIP) employs the Expectation-Maximization method for estimating destination points [74]. The method has a simple formulation, low computational complexity and outperforms existing methods such as the ones proposed by Foka and Trahania [69]. Figure 10 provides a snapshot of the robot Dabo, used for testing the BHMIP. The proposal by Palm et al. [75] focuses on the recognition of human intentions in a human–robot interaction scenario. The framework includes a method for predicting and avoiding collisions by extrapolating human intentions.

Taking into consideration the human–robot interaction factor, Ferrer et al. [76] propose a socially-aware navigation framework for allowing a robot to navigate, accompanying the person in a safe and natural way. In this scenario, the robot companion has to deal with two goals at the same time: to navigate toward the person's predicted destination, and to approach the person who accompanies them. The prediction model is based on Extended SFM (ESFM).

The cooperative navigation planner by Khambhaita and Alami [77] is a tool designed to plan cooperative trajectories for robots and humans while respecting the robot's kinematic constraints and avoiding other non-human dynamic obstacles. The planner could adapt the robot's trajectory and propose co-navigation solutions even in confined spaces. It predicts a plausible trajectory for humans, and plans a corresponding robot trajectory that satisfies social constraints. The planner generates both human and robot trajectories using a graph-based optimal solver and balances the efforts between both agents to solve the

co-navigation task. The approach includes proxemics, time-to-collision, and directional constraints during optimization. The trajectory optimization uses an elastic band approach and a least-squares problem is mapped into a hyper-graph representation to adjust the position and orientation of nodes and minimize the imposed constraints.

Figure 10. The robot Dabo accompanying a person to the desired goal while navigating in a crowded scenario [74]: (**left**) image captured from the robot camera; (**center**) the robot Dabo; (**right**) the robot GUI. The green cylinders correspond to people and the orange cylinder corresponds to the target. Permission by ELSEVIER 5483750554512.

Table 3. Approaches considered in Section 3.2.2. The table covers the methods they use for solving the navigation problem.

Reference	Methods
Svenstrup et al. [70]	Rapidly-exploring Random Tree (RRT)
	Trajectory Generation Problem
	Model Predictive Control (MPC)
	Dynamic Potential Field
Foka and Trahania [69]	Predictive navigation performed in a global manner with the use of a POMDP
	Polynomial Neural Network (PNN)
	Future motion prediction
	Robot Navigation-HPOMDP (RN-HPOMDP)
Rios-Martinez et al. [73]	RISK-RRT algorithm navigation
	Learned Gaussian Processes
	Personal Space
	Model of o-space in F-formations
Park and Kuipers [71]	The formulation of the kinematic control law
	The pose-following algorithm for smooth and comfortable motion of unicycle -type robots
Park et al. [72]	Model Predictive Equilibrium Point Control (MPEPC) framework
Du Toit and Burdick [42]	Motion Planning
Ferrer et al. [52]	SFM and prediction information
Ferrer and Sanfeliu [74]	Bayesian Human Motion Intentionality Prediction
	Sliding Window BHMIP (BHMIP)
	Two variants: the Sliding Window BHMIP and the Time Decay BHMIP
	Expectation-Maximization method
Palm et al. [75]	Recognize the human intention with relative speeds
	Collision avoidance by extrapolation of human intentions and heading angle
	Compass dial
	Fuzzy rules for Human-Robot interactions
Ferrer et al. [76]	Socially-aware navigation framework for allowing a robot to navigate accompanying the person
Khambhaita and Alami [77]	Cooperative navigation planner
	Trajectory Optimization: Elastic band
	Expectation-Maximization method
	Optimization framework
	Graph-based optimal solver
	Time-to-collision and directional constraints during optimization
Kabtoul et al. [78]	Cooperative navigation planner

In the same way that we describe in the previous subsection, ignoring the cooperation between the mobile agents in the path planning step can lead to the *freezing* of the robot. In the proposal by Kabtoul et al. [78], proactive and natural maneuvering is suggested for navigation around people. The approach consists of two main steps. First, the space is explored and dynamically divided into a set of channels using a local segment of the global

path. The optimal channel is found using a fuzzy cost model, and its center line provides the goal path to the local navigation module. To convey a human-like steering behavior, a smooth lane change maneuver is adapted to travel between channels using a Quintic transition path. In the final stage, the exact tracking control commands are derived using a sliding mode control method.

3.3. Learning-Based Approaches

In recent years, many researchers studied methods by which a robot learns the navigation policy adapted to the target environment. Deep reinforcement learning (DRL) was often used to learn this interaction policy. Approaches such as Collision Avoidance with DRL (CADRL) [79], Socially-Attentive RL (SARL) [80], and Socially-Attentive Object-Aware DRL (SOADRL) [81] have been proposed to address this problem. However, the exact positions of pedestrians can be difficult to estimate in real-life situations. To avoid this computation, end-to-end (E2E) learning approaches directly map raw sensory inputs to the desired steering commands. On the other hand, the problem of automating the computation of the reward function in DRL-based approaches has been addressed using inverse reinforcement learning (IRL). By learning the reward function directly from the data, IRL can improve the learning rate and performance of the system. Several IRL approaches were used to learn social navigation behaviors for robots, including optimizing reward function parameters using maximum likelihood estimation [82], and modeling IRL from a Bayesian perspective [83].

3.3.1. Deep Reinforcement Learning and End-to-End Approaches

In recent years, Deep reinforcement learning (DRL) has emerged as a successful tool for solving those tasks where it is not easy to engineer a direct solution. Briefly, DRL introduces deep neural networks to solve reinforcement learning problems. A multitude of approaches for training a collision avoidance policy based on DRL has been proposed in the last decade, some of which take social awareness into account. Table 4 enumerates some of these approaches, that are described in this section.

Chen et al. [79] use DRL to train a navigation strategy in a multi-agent scenario (Collision-Avoidance with DRL, CADRL). The hand-crafted reward function positively benefits reaching the desired goal and penalizes collisions. This proposal was extended for considering humans and social norms in Chen et al. [84] (SA-CADRL). Specifically, in this work, a socially compliant behavior (e.g., passing by the right side) was learned using a reward function that depends on situational dynamics. The scheme was improved by Everett et al. [85], who leveraged GPU processing to train multiple agents in parallel, and Long Short-Term Memory (LSTM) to convert the variable size state of the crowd into the fixed-size vector. Although these approaches were successful for dealing with multi-agent navigation, they failed to account for complex interactions among humans [86]. To improve the comfort of people sharing the crowd with a robot, Hu et al. [87] use social stress indexes in the reward function and value network of the DRL framework. A multi-layer perceptron is employed to extract local features and social-attention scores. As in the rest of the approaches cited in this paragraph, they make use of exact pedestrian positions in the input. This will be not an easy task to address for a real robot deployed in a real scenario [88]. Gil et al. [89] proposed computing robot actions by a combination of robot velocities learned by a RL model (AutoRL [90]) and robot velocities computed using an SFM [36].

Previous approaches typically address robot navigation in a crowded environment as a one-way human–robot interaction problem, experiencing problems when the crowd grows. To avoid this situation, Chen et al. [80] proposed explicitly modeling the crowd-robot Interaction (Socially Attentive RL approach, SARL). Self-attention is used to discover the collective importance of the crowd by considering human–human and human–robot interactions. Obstacles are however left out of the policies learned by the SARL approach. This issue is addressed by the SOADRL [81]. This approach extends the SARL by separately processing information related to static and dynamic objects. Moreover, the SOADRL

addresses the problem of navigation in crowds with sensors that only offer a limited field of view. As with the SA-CADRL, the SARL and SOADRL make use of the exact mobile agents' position in the input. To deal with the problem of a large-sized crowd, Chen et al. [91] suggested enabling the system to identify those humans in the crowd that are most critical for navigation. The proposal uses a graph representation to learn this policy, which encodes information about the crowd and predicts human attention scores in the navigation task. A graph convolutional network is trained based on human gaze data, which accurately predicts human attention to different agents in the crowd as they perform a navigation task. The learned attention is integrated into the graph-based reinforcement learning architecture. The problem of partial observability (due for example to sensor limitations, occlusions, or perception uncertainty) is considered by Gao et al. [92]. To achieve this, the proposal makes use of a recurrent neural network (RNN) (the so-called gated recurrent unit (GRU)) to infer the unobservable states. To respond in real-time to human behaviors, Samsani and Muhammad [86] proposed modeling Danger Zones for the robot. These zones are formulated by taking into account real-time human behavior, and then they encode all possible actions that people can take at a given time. The robot is trained to avoid these danger zones for safe and secure navigation.

With the help of deep neural networks, DRL can apply end-to-end (E2E) learning, i.e., to learn a black box model, extract features from captured high-dimensional data and learn complex policies. Thus, they avoid the need for detection and efficient tracking of people. From a general point-of-view, where the goal is to directly map raw sensory inputs to desired steering commands, the learning process can be summarized as follows [93]:

- the robot moves according to a given action and obtains information from the environment (observations) and a reward;
- following a policy, and given the captured observation, an action is generated;
- the policy is updated by an RL-based algorithm.

The robot finally gets the optimal policy to achieve the goal by repeating the learning process. In order to use DRL in an E2E robot navigation context, the whole problem setting must be stated and translated into an RL framework [94]. For instance, to avoid collisions, Long et al. [93] proposed directly mapping raw 2D laser measurements to desired motion commands using a 4-hidden-layer neural network. Shi et al. [95] proposed an E2E navigation framework that translates sparse laser-ranging results into movement actions. The goal of using this highly abstract data as input is that robots trained by simulation can be also deployed in real environments. The proposal shows robust navigation but in relatively simple environments. The Role Playing Learning (RPL) [96] endows a robot with appropriate group behaviors when it is traveling with a human companion. This E2E proposal uses neural networks to map sensory data to velocity outputs while adhering to social norms. The RPL process is formulated under an RL framework and optimized using Trust Region Policy Optimization (TRPO). To directly learn control strategies from visual input, Mnih et al. [97] combined a convolutional neural Network (CNN) with a Q learning algorithm (DQN network model). Lee and Yusuf [98] mapped the data from an RGB-D camera to steering commands. A Deep neural network is employed for detecting the target, and the collision avoidance and navigation are addressed by a DQN model (or a double DQN (DDQN)). The approach is able to deal with static and dynamic obstacles, but it does not consider social factors.

Previously described E2E planning approaches use RL. Another possibility is to use a supervised learning scheme to imitate expert demonstrations (imitation learning, IL). IL is sample efficient and, given training data, a navigation model could be found quickly [99]. On the other hand, IL is conceptually less robust than RL [100], as RL allows the robot to learn from its own mistakes during training [101]. Pfeiffer et al. [99] proposed a data-driven E2E motion planner, where the robot learns to navigate as the human user likes. To achieve this, expert demonstrations of how to navigate in a given training environment are provided. Using this data, the aim is not only to replicate the provided demonstrations in one specific scenario, but rather to be able to learn collision avoidance policies and transfer

them to previously unseen environments. As Figure 11 shows, a single model based on the TensorFlow framework [102] is used for extracting information and for estimating the steering commands. IL and RL are combined by Pfeiffer et al. [100] into a single target-driven, mapless navigation scenario. The Reinforced imitation learning (R-IL) uses expert demonstrations to pre-train the navigation policy, and then applies a Constrained Policy Optimization (CPO) [103] for incorporating constraints during RL training. The authors demonstrate that this scheme can reduce the training time, to reach the same performance level that plain RL, by a factor of 5. Pfeiffer et al. [99,100] test their solutions in a static environment. The CrowdMove approach [104] is a multi-robot, multi-scenario, and multi-stage training (3M) framework. It employs a 4-hidden-layer neural network as a nonlinear function approximator to the navigation policy, and extends the Proximal Policy Optimization (PPO) [105] to the multi-robot scenario. Experiments demonstrate that the navigation policy can achieve autonomous navigation for different mobile platforms in a large variety of crowd environments.
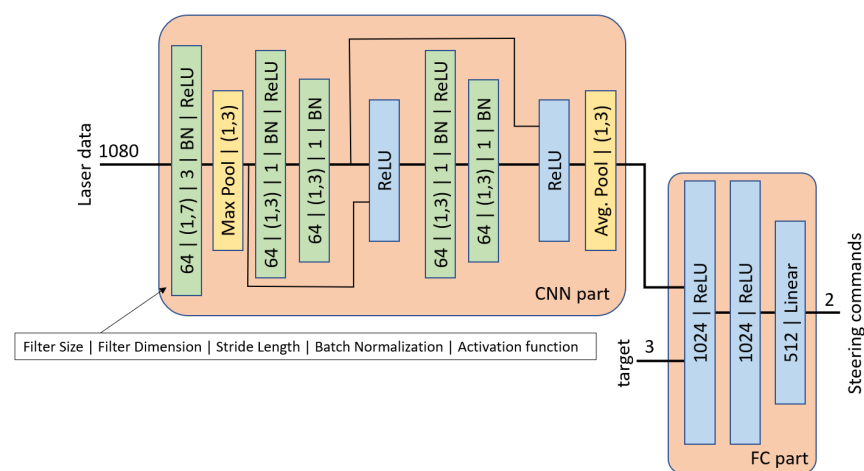

Figure 11. Structure of the CNN employed by Pfeiffer et al. [99]. Two residual building blocks provide the structure of the CNN part [106], which takes the input data (laser data) and provides the feature vector to the FC part. The FC layer of the model fuses this feature vector with the target information to obtain the translational and rotational steering commands. L1 regularization is applied to all model parameters.

Long-term planning and learning for navigating within other mobile agents have been also combined as separate (but intimately tied together) modules. Instead of the E2E previous solutions, the aim here is to allow the motion planner to estimate the path and decompose it into a set of subgoals, being the learned low-level controller in charge of adapting the route to the dynamics of the current situation. Gao et al. [107] proposed combining a path planner with a neural-network motion controller (the intention-net). The intention net maps images to motion controls in an E2E scheme. The path planner uses an a priori 2D map to compute the paths. This planned path provides the intentions to the intention-net layer. The navigation system proposed by Pokle et al. [108] also combines a path planner and machine learning. Here, the global planner computes the routes toward a goal, and a deep local trajectory planner and velocity controller provides the motion commands. The low-level motion controller is responsible for avoiding obstacles and also for respecting the space of nearby pedestrians. Significantly, both approaches demonstrated that this scheme outperforms an end-to-end framework where planning guidance is not considered. Pérez-D'Arpino et al. [109] proposed a navigation stack that combines motion planning and RL. The RL component learns to handle the local interactions with other mobile agents as it pursues the globally planned trajectory. Choi et al. [110] proposed a framework where the DRL learns navigation policies that adapt to a wide range of reward weightings and other navigation parameters. Then, a Bayesian deep learning method is used for optimizing navigation parameters to human preferences.

Table 4. Approaches considered in Section 3.3.1. The table covers the methods for DRL and E2E approaches.

Reference	Methods
Chen et al. [79]	Decentralized Multi-agent Collision Avoidance algorithm based on a novel application of deep reinforcement learning
Chen et al. [84]	SA-CADRL, a multi-agent collision avoidance algorithm that considers and exhibits socially compliant behaviors
	Time-efficient navigation policy that respects common social norms
	Reinforcement learning framework
Everett et al. [85]	Collision avoidance algorithm, GA3C-CADRL, that is trained in simulation with deep RL without requiring any knowledge of other agents' dynamic
	Long Short-Term Memory (LSTM)
	LSTM that enables the algorithm to use observations of an arbitrary number of other agents
Samsani and Muhammad [86]	Human Behavior Resemblance Using Deep Reinforcement Learning
	The Danger Zones are formulated by considering the real time human behavior
Hu et al. [87]	Deep reinforcement learning framework (DRL) and the value network
	The DRL framework incorporating these social stress indexes
Dugas et al. [88]	Reinforcement Learning of Robot Navigation in Dynamic Human Environments
	NavRepSim environment is designed with RL applications in mind
Gil et al. [89]	Social Force Model (SFM) allowing human-aware
	Two Machine Learning techniques: Social navigation and Neural Network (NN)
	RL technique
Francis et al. [90]	PRM-RL:Probabilistic road-maps (PRMs) as the sampling-based planner and reinforcement learning-RL method in the indoor navigation context
Chen et el. [80]	Crowd-Robot Interaction (CRI)
	Attention-based Deep Reinforcement Learning
Liu et al. [81]	Imitation learning and deep reinforcement learning approach for motion planning in such crowded and cluttered environments
Chen et al. [91]	Graph convolutional network (GCN) for reinforcement learning to integrate information
	Attention network trained using human gaze data for assigning adjacency values.
Gao et al. [92]	Learn an efficient navigation policy that exhibits socially normative navigation behaviors
	Convolutional social pooling layer that robustly models human–robot co-operations and complex interactions between pedestrians
	Partial observability in socially normative navigation
Long et al. [93]	Decentralized sensor-level collision avoidance policy for multi-robot systems
	Policy gradient-based reinforcement learning algorithm
Gromniak and Stenzel [94]	End-to-end Deep reinforcement learning
Shi et al. [95]	Navigation strategy based on deep reinforcement learning (DRL)
	Conventional A3C algorithm, an ICM A3C model was proposed
Li et al. [96]	Role Playing Learning (RPL)
	NN policy is optimized end-to-end
[97]	A deep Q-network (DQN), combine reinforcement learning with a deep neural networks
Lee and Yusuf [98]	Deep reinforcement learning for autonomous mobile
	The trained DQN and DDQN policies are robot navigation in an unknown environment evaluated in the Gazebo testing environment
	Two types of deep Q-learning agents, such as deep Q-network and double deep Q-network agents
Pfeiffer et al. [99]	Data-driven end-to-end motion planning approach for a robotic platform
	End-to-end model is based on a CNN
	Learn navigation strategies
Pfeiffer et al. [100]	Imitation learning(IL) and reinforcement learning (RL)
Tai et al. [101]	A map-less motion planner was trained end-to-end through continuous control deep-RL from scratch
	The learned planner can generalize to a real non-holonomic differential robot platform without any fine-tuning to real-world samples
Fan et al. [104]	Multi-robot, multi-scenario, and multi-stage training framework
Gao et al. [107]	Two-level hierarchical approach:Model-free deep learning and model-based path planning
	Neural-network motion controller, called the intention-net, is trained end-to-end to provide robust local navigation
	Path planner uses a crude map

Table 4. *Cont.*

Reference	Methods
Pokle et al. [108]	Hierarchical planning and machine learning
	Traditional global planner to compute optimal paths towards a goal
	Deep local trajectory planner and velocity controller to compute motion commands
	Combines traditional planning with modern deep learning techniques
Pérez-D'Arpino et al. [109]	Reinforcement learning to learn robot policies
	The proposed model uses a motion planner that provides a globally planned trajectory whereas the reinforcement component handles the local interactions needed for on-line adaptation to pedestrians
Choi et al. [110]	Novel deep RL navigation method that can adapt its policy to a wide range of parameters and reward functions without expensive retraining
	Bayesian deep learning method

3.3.2. Inverse Reinforcement Learning

One of the major issues to solve when designing a DRL-based scheme is the choice of the reward function. Thus, a bad choice can dramatically impact learning rate and performance [111]. One alternative to the hand-crafted setting of the reward function is to learn this function directly from the data. Inverse Reinforcement Learning (IRL) is a good mechanism for addressing this learning process. Table 5 presents some recent approaches that use IRL in the context of robot social navigation. Ziebart et al. [112] proposed a probabilistic approach based on the principle of maximum entropy that allows optimizing the parameters of a reward function using maximum likelihood estimation (MLE). The scheme was used by Henry et al. [82] to learn human-like navigation behavior in crowded environments. Specifically, the approach learns from example paths by estimating values such as crowd density on the fly using Gaussian processes. The performance of the approach was evaluated within a realistic crowd simulation and resulted in natural paths that blended seamlessly with existing crowd movements. Pérez-Higueras et al. [113] also proposed IRL for social navigation. However, instead of using the costs for path plans, they employ them to learn local execution policies and provide steering commands to the robot. The global path planner is based on Dijkstra's algorithm and the local planner is an extension of the Trajectory Roll-out algorithm [114]. Vasquez et al. [115] compared IRL and manual tuning in the learning of the parameters of a reward function. They proposed evaluation metrics to benchmark these techniques. The results from simulations using two IRL approaches and several feature sets are reported and evaluated using objective and subjective performance metrics. Obtained results demonstrated that IRL-learned reward parameters are better than manually tuned ones. The maximum entropy approach was used by Kretzschmar et al. [116] to learn the parameters of a joint trajectory distribution of all navigating agents in an environment, including the robot itself. Hamiltonian Markov chain Monte Carlo sampling is used to compute the feature expectations over the resulting high-dimensional continuous distributions. While learning a joint distribution over all agents allows for high-quality inference, it does not scale to moderately populated settings (e.g., a few dozen agents).

Wang et al. [117] proposed a Neural Network Rapidly-exploring Random Trees (NN-RRT*) planner for generating robot paths in human–robot interaction environments. Based on this planner, they propose the NRTIRL framework. Figure 12 provides an overview of the framework. Briefly, the idea is to compare planned routes generated by the NN-RRT* and demonstration routes generated by human volunteers. To achieve this, features of demonstration and planned routes are provided to the neural network to obtain their corresponding cost. If their difference is higher than an allowable error, the feature vector of the planned route is used as input, and the cost of the demonstration route is used as the output of the neural network. The neural network is then optimized until the difference between demonstration and planned routes is lower than this allowable error. When the

parameters of the neural network converge, the new NN-RRT* planner is updated and able to generate routes that are more similar to the ones provided by humans.
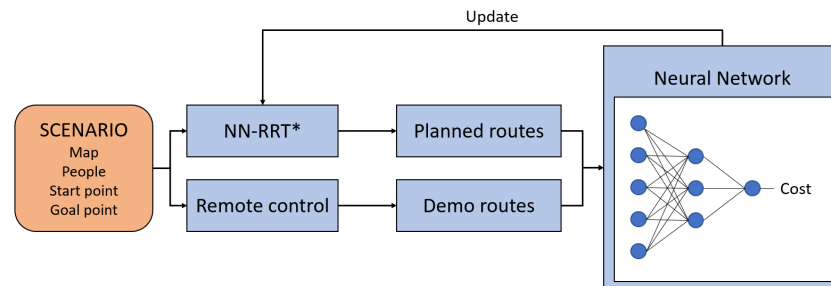

Figure 12. Overview of the NRTIRL [117].

Ramachandran and Amir [118] proposed modeling IRL from a Bayesian perspective (BIRL) and solving reward learning and apprenticeship learning using a modified Markov Chain Monte Carlo (MCMC) algorithm. The reward is modeled as a random variable vector that determines the distribution of expert states and actions. The distribution of the rewards which best explains expert trajectories is then inferred. Kim et al. [83] use BIRL to learn a linear reward function over features extracted from RGB-D cameras on a robotic wheelchair. The proposed framework consists of three modules: feature extraction, BIRL, and path planning. The feature extraction module extracts information from a RGB-Depth sensor, the BIRL module uses expert demonstrations to learn a cost function that considers social variables, and the planning module uses a three-layer architecture to optimize the global and local paths while avoiding obstacles. Okal and Arras [119] proposed a new approach to modeling socially normative behavior in robots using MDPs and a modified version of the BIRL. They focused on spatial robot motion behaviors and use a graph-based representation to integrate task-specific constraints into the MDP. The use of this graph-based representation allows the authors to instantiate global planners such as RRT or A* using the rewards learned for local navigation.

Table 5. Approaches considered in Section 3.3.2. The table covers the reviewed methods for IRL.

Reference	Methods
Ziebart et al. [112]	Maximum Entropy Inverse Reinforcement Learning
	Inverse reinforcement and imitation learning
Henry et al. [82]	Inverse Reinforcement Learning with
	Gaussian Processes for environmental
Pérez-Higueras et al. [113]	Inverse reinforcement learning
	Global path planner- Dijkstra's algorithm
Gerkey and Konolige [114]	DARPA Learning Applied to Ground
	Globally optimal paths on a cost map
Vasquez et al. [115]	Compare IRL based learning methods
	Motion Planning -grid-based GPU
Kretzschmar et al. [116]	Hamiltonian Markov Chain Monte Carlo (MCMC)
	Learn a model of the navigation behavior of cooperatively
	navigating agents such as pedestrians
	Voronoi graph of the environment
Wang et al. [117]	Neural Network Rapidly-exploring Random Trees
	A cost function based on neural network
Ramachandran and Amir [118]	Bayesian IRL (BIRL)
	Reward learning is an estimation task
	Markov Decision Process
	Apprenticeship learning task
Kim et al. [83]	Path planning module
	Inverse Reinforcement Learning
	Framework for socially adaptive path planning in dynamic
	environments
	Generating human-like path trajectory
Okal and Arras [119]	Graph-based representation of the continuous
	New extension of BIRL

3.4. Multi-Behavior Navigation Approaches

When people move in an environment where there are many other people, it is not easy to behave like a rigid object, whose aim is to reach a goal without touching these other people. We usually have to ask for permission to pass, sometimes even interrupting a conversation, informing with gestures of our intentions, or even giving a light touch so that a person moves slightly to let us pass. This scenario, which is complex to translate to a robotic agent as it involves mixing originally different functionalities, implies integrating the navigation stack with other modules present in the software architecture. This integration can be done by incorporating all these functionalities into the navigation framework (as we would do for a robot that follows a person, for example, by integrating the person detector into the navigation stack) or by allowing all these functionalities to cooperate in a single shared representation of the environment. In this section, we present a few examples of approaches (Table 6) that concatenate actions associated with different functionalities for reaching a navigation goal (multi-modal behaviors). The first of the integration options discussed above is the one used by Kamezaki et al. [16] and Dugas et al. [120]. The second is the one proposed by Vega-Magro et al. [121].

Allowing a robot to switch behaviors depending on the context entails entering the field of self-adaptation. In the proposal by Chen et al. [122], after estimating a local trajectory for obstacle avoidance based on predicted mobile agents' routes, the robot can choose a travel model for navigation according to the traffic state. Freitas et al. [123] proposed a framework that allows a robot to change its navigation configuration depending on the context (e.g., aborting a mission when the power autonomy level is low and the robot is redirected to the charging station). Some of these context changes are related to the presence of people in the environment. However, this work does not really focus on enabling a robot to navigate naturally and socially correctly between people. Using the MROS model-based framework, Bozhinoski and Wijkhuizen [124] presented a similar approach that adapts the local planner configuration at run-time to satisfy a set of quality requirements. None of these methods consider other action skills for the robot apart from moving through the environment.

The IAN (Interaction Actions for Navigation) approach [120] is defined as a *high-level, multi-behavior, interaction-aware planning for navigation in unstructured, human-populated environments*. Briefly, the approach allows the robot to choose a specific navigation behavior based on the observed state of the environment. These behaviors combine actions associated with different robotic functionalities (multi-modal behaviors). The first behavior considers a static or sparsely dynamic scenario and consists of a motion planner based on the local velocity field. RVO is used for modeling mobile agents. In the second behavior, the robot is able to verbally announce that it is moving, while indicating the direction of movement with its hand. The third behavior is employed by the robot to pass through very close people in a highly populated scenario. The Dynamic Window Approach (DWA) is employed as a local planner. When the DWA cannot find a non-zero solution, the robot moves at a very low speed, with its arm forward, reaching its own base footprint, and announcing verbally that it is passing through the people.

The SNAPE framework [121] emphasizes that a robot needs social behaviors and cooperation from humans to navigate socially. The core of the proposal is the use of the CORTEX software architecture for robotics [125,126]. This architecture deploys a set of software agents surrounding a graph-based world model. These agents are organized into five layers (perception, social, navigation, human–robot interaction, and high-level planning). Using CORTEX, the SNAPE framework manages all the information flow, from the perception of the environment to behavior planning. However, planning is considered not only at the behavior level, but also at the dialogue and task-planning levels. The decision maker coordinates the activity at all layers. For instance, when the robot detects that a person is blocking its route, it can approach this person, draw its attention with a specific dialogue, and ask permission to pass.

Based on an inducible SFM (i-SFM), Kamezaki et al. [16] propose a reactive, proactive, and inducible way-point-i-SFM fused-path planning method (the proximal crowd navigation (PCN) approach). The PCN considers proximity but also physical touching for tracing the routes, and uses i-SFM for predicting human motions. The PCN is able to predict the movement of the people in the robot's surroundings, generate multiple paths including physical-touch paths using the way-point method, and determine the route taking into consideration the movement efficiency and the degree of crowd invasion.

Table 6. Approaches considered in Section 3.4. The table covers multi-behavior navigation approaches and multi-modal behavior navigation approaches.

Reference	Methods
Chen et al. [122]	Travel model selection according to the traffic state
Freitas et al. [123]	Self-adaptation based on QoS metrics
	Planning encoded in Behaviour Trees. CORTEX software architecture
Bozhinoski and Wijkhuizen [124]	Self-adaptation based on MROS framework
	Quality models for adapting the local planner configuration at run-time
Kamezaki et al. [16]	Proximal Crowd Navigation (PCN) approach (proximity and physical-touching).
	inducible SFM (i-SFM) for predicting human motion.
Dugas et al. [120]	The IAN framework
	Interaction actions (saying, touching, and gesturing) for navigating in crowded scenarios
Vega-Magro et al. [121]	The SNAPE framework
	CORTEX cognitive software architecture

4. Discussion

Robot navigation within humans has been a goal of the robotics community for decades. Reactive approaches, such as the ones presented in this study, were designed to handle dynamic scenarios, but have severe constraints (e.g., constant velocities for moving obstacles) that do not allow them to handle complex environments. These approaches do not model humans and their not always predictable behavior. Moreover, they cannot take into account multi-agent behaviors, such as joint planning. To deal with these problems, proactive approaches focus on human modeling and reciprocal planning. They have faced several problems, related to, for example, giving relevance to social comfort (practically covered by most of the proactive and learning-based approaches), the interaction with humans but also with groups of humans, or the Freezing Robot Problem. The problem of the first approaches for modeling human motion was solved by considering goal-based policies. Considering that people are moving towards certain goals, the robot can project their motion (trajectories) on a map and trace a route for avoiding them. However, these goals are not necessarily available to the robot. Given the positions in a map of all people surrounding the robot, proxemics can be used for extending this map, taking into account social norms. In addition to space, other factors need to be taken into account. Traits such as body posture and facial expressions can help the robot to be more approachable and predictable, further improving its ability to function in social environments. As we review in this work, proactive approaches have evolved to manage human–robot interaction and, subsequently, human–robot cooperation. Cooperative planning provides navigation efficiency and also human acceptability [120]. In parallel, learning-based approaches have emerged in this last decade to learn motion planners. As data-driven approaches, the major strength of these methods is that they are able to estimate a practical human model without having to specify social norms, as they are implicitly present in the data. In recent years, the major limitation was that these approaches need large data sets, which were captured from virtual scenarios. The great difference between the simulation environment and the real-world one is the major challenge to transfer the trained model to a real

robot. Virtual-to-real approaches are currently able to generalize the learned planners and to satisfactorily deployed them in unseen real environments [101]. It is clear that real-world experiments involving actual people and uncontrolled environments are crucial to validating the effectiveness of social navigation. These experiments reveal the challenges and complexities that arise in real-life situations, providing valuable insights into how robots can improve their navigation skills.

The use of databases to calibrate or train the system is usually necessary in all methods examined. These datasets are usually obtained by recording pedestrian trajectories with cameras or sensors in specific environments, or are collected by experts. Relying solely on recorded datasets may not be good practice, as the goal is to generalize pedestrian behavior, which requires a variety of datasets from different sources. In the case of proactive methods, the use of real data is common. For example, the Edinburgh Informatics Forum Pedestrian Database was successfully used by Ferrer and Sanfeliu [48,74]. Others, such as Luber et al. [127] and Ferrer and Sanfeliu [74], used The Freiburg People Tracker. However, this is not the case, as mentioned above, for learning-based methods. These usually describe the tools used to generate virtual learning environments, which enable them to obtain the volumes of data needed to successfully complete the training. For these methods to be successful in real environments, the data used for prediction or learning must be as realistic and diverse as possible. Therefore, it is crucial to collect data sets from a variety of sources and experts.

Two relevant topics have begun to be taken into account in the most recent proposals. On the one hand, the possibility for the robot to handle different behavioral options, and that, depending on the context, the robot itself self-adapts its behavior. Self-adaptation is a topic widely addressed in robotics, with specific proposals related to navigation and including parameter reconfiguration, algorithm changes, or even reconfiguration of the architecture itself [123,124]. Although it has not been analyzed in depth in this survey, some references were added to Section 3.4. On the other hand, navigation has started to be considered as a task that is not only about finding a path free of obstacles, or that this path is as close as possible to the one a person would follow, but may require specific skills that we would not include a priori in a navigation stack. Thus, to avoid getting stuck in an environment densely crowded with people, the robot may need to talk to people, or even push them lightly. This multi-modal collaboration scenario can be useful for a robot to become more socially aware, allowing it, for example, to greet people it crosses paths with.

5. Conclusions and Future Work

This survey analyzes the problem of robot navigation in every day, crowded environments. The analysis of recent studies highlights the advancements in robot navigation, particularly in the area of social navigation. Although purely reactive proposals were presented, when a mobile robot is deployed in an environment where people move around freely, it becomes necessary for this robot to predict the movement of these other agents. The use of prediction enables the robot to quickly adapt and optimize its navigation, To generate these predictions, navigation algorithms have evolved from a human–robot interaction scenario to a human–robot cooperation one, where it is expected that people will proactively help the robot to find a free and safe path. However, the complexity and variety of human behavior in the real world can make this assumption fail. Recent approaches propose that the mobile robot can interact with people not only because their paths may cross on the map, but more actively, through gestures, vocalization and touch, to require their help in navigating [16,120,121]. As a signaling mechanism for conveying an intention to humans, incorporating features such as body posture and gestures also contributes to making the robot appear more friendly and predictable to humans, leading to better human–robot interactions and an overall improved experience.

Future work in this field should focus on thoroughly reviewing existing experiments and exploring ways to further improve robot performance. It is also important to keep abreast of the latest developments and advances in this field. In addition, as mentioned

above, it could be interesting to analyze methods using record data and methods using test databases and to analyze the behavior of the methods in real environments.

References

1. Gladden, M.E. Who Will Be the Members of Society 5.0? Towards an Anthropology of Technologically Posthumanized Future Societies. *Soc. Sci.* **2019**, *8*, 148. [CrossRef]
2. SPARC. *Robotics 2020 Multi-Annual Roadmap for Robotics in Europe*; Technical Report; SPARC: The Partnership for Robotics in Europe, euRobotics Aisbl: Brussels, Belgium, 2015.
3. Seibt, J.; Damholdt, M.F.; Vestergaard, C. Integrative social robotics, value-driven design, and transdisciplinarity. *Interact. Stud.* **2020**, *21*, 111–144. [CrossRef]
4. Rossi, S.; Rossi, A.; Dautenhahn, K. The Secret Life of Robots: Perspectives and Challenges for Robot's Behaviours during Non-interactive Tasks. *Int. J. Soc. Robot.* **2020**, *12*, 1265–1278. [CrossRef]
5. Sandini, G.; Sciutti, A.; Vernon, D. Cognitive Robotics. In *Encyclopedia of Robotics*; Springer: Berlin/Heidelberg, Germany, 2021.
6. Hellström, T.; Bensch, S. Understandable robots-What, Why, and How. *Paladyn J. Behav. Robot.* **2018**, *9*, 110–123. [CrossRef]
7. Rios-Martinez, J.; Spalanzani, A.; Laugier, C. From Proxemics Theory to Socially-Aware Navigation: A Survey. *Int. J. Soc. Robot.* **2015**, *7*, 137–153. [CrossRef]
8. Samarakoon, S.M.B.P.; Muthugala, M.A.V.J.; Jayasekara, A.G.B.P. A Review on Human–Robot Proxemics. *Electronics* **2022**, *11*, 2490. [CrossRef]
9. Charalampous, K.; Kostavelis, I.; Gasteratos, A. Recent trends in social aware robot navigation: A survey. *Robot. Auton. Syst.* **2017**, *93*, 85–104. [CrossRef]
10. Gao, Y.; Huang, C.M. Evaluation of Socially-Aware Robot Navigation. *Front. Robot. AI* **2022**, *8*, 721317. [CrossRef]
11. Zhu, K.; Zhang, T. Deep reinforcement learning based mobile robot navigation: A review. *Tsinghua Sci. Technol.* **2021**, *26*, 674–691. [CrossRef]
12. Chik, S.; Fai, Y.; Su, E.; Lim, T.; Subramaniam, Y.; Chin, P. A review of social-aware navigation frameworks for service robot in dynamic human environments. *J. Telecommun. Electron. Comput. Eng.* **2016**, *8*, 41–50.
13. Kruse, T.; Pandey, A.K.; Alami, R.; Kirsch, A. Human-aware robot navigation: A survey. *Robot. Auton. Syst.* **2013**, *61*, 1726–1743. [CrossRef]
14. Chen, Y.; Zhao, F.; Lou, Y. Interactive Model Predictive Control for Robot Navigation in Dense Crowds. *IEEE Trans. Syst. Man Cybern. Syst.* **2022**, *52*, 2289–2301. [CrossRef]
15. Sisbot, E.A.; Marin-Urias, L.F.; Alami, R.; Simeon, T. A Human Aware Mobile Robot Motion Planner. *IEEE Trans. Robot.* **2007**, *23*, 874–883. [CrossRef]
16. Kamezaki, M.; Tsuburaya, Y.; Kanada, T.; Hirayama, M.; Sugano, S. Reactive, Proactive, and Inducible Proximal Crowd Robot Navigation Method Based on Inducible Social Force Model. *IEEE Robot. Autom. Lett.* **2022**, *7*, 3922–3929. [CrossRef]
17. Fiorini, P.; Shiller, Z. Motion planning in dynamic environments using the relative velocity paradigm. In Proceedings of the 1993 IEEE International Conference on Robotics and Automation (ICRA), Atlanta, GA, USA, 2–6 May 1993; Volume 1, pp. 560–565. [CrossRef]
18. Rudenko, A.; Palmieri, L.; Arras, K.O. Joint Long-Term Prediction of Human Motion Using a Planning-Based Social Force Approach. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018; pp. 4571–4577. [CrossRef]
19. Lee, J.; Won, J.; Lee, J. Crowd Simulation by Deep Reinforcement Learning. In Proceedings of the 11th ACM SIGGRAPH Conference on Motion, Interaction and Games, MIG '18, Limassol, Cyprus, 8–10 November 2018; Association for Computing Machinery: New York, NY, USA, 2018; . [CrossRef]
20. Karlsson, S.; Koval, A.; Kanellakis, C.; Agha-mohammadi, A.; Nikolakopoulos, G. D$^*_{+S}$: A Generic Platform-Agnostic and Risk-Aware Path Planing Framework with an Expandable Grid. *arXiv* **2021**, arXiv:2112.05563.

21. Shiller, Z.; Large, F.; Sekhavat, S. Motion planning in dynamic environments: Obstacles moving along arbitrary trajectories. In Proceedings of the 2001 IEEE International Conference on Robotics and Automation (ICRA), Seoul, Korea, 21–26 May 2001; Volume 4, pp. 3716–3721. [CrossRef]

22. Kluge, B.; Prassler, E. Reflective navigation: Individual behaviors and group behaviors. In Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA), New Orleans, LA, USA, 26 April–1 May 2004; Volume 4, pp. 4172–4177. [CrossRef]

23. Fulgenzi, C.; Spalanzani, A.; Laugier, C. Dynamic Obstacle Avoidance in uncertain environment combining PVOs and Occupancy Grid. In Proceedings of the 2007 IEEE International Conference on Robotics and Automation (ICRA), Roma, Italy, 10–14 April 2007; pp. 1610–1616. [CrossRef]

24. Khatib, O. Real-time obstacle avoidance for manipulators and mobile robots. In Proceedings of the 1985 IEEE International Conference on Robotics and Automation (ICRA), St. Louis, MO, USA, 25–28 March 1985; Volume 2, pp. 500–505. [CrossRef]

25. Yao, Q.; Zheng, Z.; Qi, L.; Yuan, H.; Guo, X.; Zhao, M.; Liu, Z.; Yang, T. Path planning method with improved artificial potential field—A reinforcement learning perspective. *IEEE Access* **2020**, *8*, 135513–135523. [CrossRef]

26. Borenstein, J.; Koren, Y. The vector field histogram-fast obstacle avoidance for mobile robots. *IEEE Trans. Robot. Autom.* **1991**, *7*, 278–288. [CrossRef]

27. Babinec, A.; Duchoň, F.; Dekan, M.; Mikulová, Z.; Jurišica, L. Vector Field Histogram* with look-ahead tree extension dependent on time variable environment. *Trans. Inst. Meas. Control* **2018**, *40*, 1250–1264. [CrossRef]

28. Palm, R.; Driankov, D. Velocity potentials and fuzzy modeling of fluid streamlines for obstacle avoidance of mobile robots. In Proceedings of the 2015 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Istanbul, Turkey, 2–5 August 2015; pp. 1–8. [CrossRef]

29. Zheng, Z.; Zhu, G.; Sun, Z.; Wang, Z.; Li, L. Improved Social Force Model Based on Emotional Contagion and Evacuation Assistant. *IEEE Access* **2020**, *8*, 195989–196001. [CrossRef]

30. Reddy, A.; Malviya, V.; Kala, R. Social Cues in the Autonomous Navigation of Indoor Mobile Robots. *Int. J. Soc. Robot.* **2021**, *13*, 1335–1358. [CrossRef]

31. Helbing, D.; Molnar, P. Social Force Model for Pedestrian Dynamics. *Phys. Rev. E* **1995**, *51*, 4282. [CrossRef] [PubMed]

32. Trautman, P.; Krause, A. Unfreezing the Robot: Navigation in Dense, Interacting Crowds. In Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Taipei, Taiwan, 18–22 October 2010; pp. 797–803. [CrossRef]

33. Truong, X.T.; Ngo, T.D. Toward Socially Aware Robot Navigation in Dynamic and Crowded Environments: A Proactive Social Motion Model. *IEEE Trans. Autom. Sci. Eng.* **2017**, *14*, 1743–1760. [CrossRef]

34. Snape, J.; Berg, J.V.D.; Guy, S.J.; Manocha, D. The Hybrid Reciprocal Velocity Obstacle. *IEEE Trans. Robot.* **2011**, *27*, 696–706. [CrossRef]

35. van den Berg, J.; Lin, M.; Manocha, D. Reciprocal Velocity Obstacles for real-time multi-agent navigation. In Proceedings of the 2008 IEEE International Conference on Robotics and Automation (ICRA), Pasadena, CA, USA, 19–23 May 2008; pp. 1928–1935. [CrossRef]

36. Zanlungo, F.; Ikeda, T.; Kanda, T. Social force model with explicit collision prediction. *EPL Europhys. Lett.* **2011**, *93*, 68005. [CrossRef]

37. Ferrer, G.; Garrell, A.; Sanfeliu, A. Social-aware robot navigation in urban environments. In Proceedings of the 2013 European Conference on Mobile Robots (ECMR), Barcelona, Spain, 25–29 September 2013; pp. 331–336. [CrossRef]

38. Shiomi, M.; Zanlungo, F.; Hayashi, K.; Kanda, T. Towards a Socially Acceptable Collision Avoidance for a Mobile Robot Navigating Among Pedestrians Using a Pedestrian Model. *Int. J. Soc. Robot.* **2014**, *6*, 443–455. [CrossRef]

39. Trautman, P.; Ma, J.; Murray, R.M.; Krause, A. Robot navigation in dense human crowds: Statistical models and experimental studies of human—Robot cooperation. *Int. J. Robot. Res.* **2015**, *34*, 335–356. [CrossRef]

40. Large, F.; Vasquez, D.; Fraichard, T.; Laugier, C. Avoiding cars and pedestrians using velocity obstacles and motion prediction. In Proceedings of the 2004 IEEE Intelligent Vehicles Symposium, Parma, Italy, 14–17 June 2004; pp. 375–379.

41. Thompson, S.; Horiuchi, T.; Kagami, S. A probabilistic model of human motion and navigation intent for mobile robot path planning. In Proceedings of the 2009 4th International Conference on Autonomous Robots and Agents, Wellington, New Zealand, 10–12 February 2009; pp. 663–668.

42. Du Toit, N.E.; Burdick, J.W. Robot Motion Planning in Dynamic, Uncertain Environments. *IEEE Trans. Robot.* **2012**, *28*, 101–115. [CrossRef]

43. Joseph, J.M.; Doshi-Velez, F.; Huang, A.S.; Roy, N. A Bayesian nonparametric approach to modeling motion patterns. *Auton. Robot.* **2011**, *31*, 383–400. [CrossRef]

44. Aoude, G.; Luders, B.; Joseph, J.M.; Roy, N.; How, J.P. Probabilistically safe motion planning to avoid dynamic obstacles with uncertain motion patterns. *Auton. Robot.* **2013**, *35*, 51–76. [CrossRef]

45. Ziebart, B.D.; Ratliff, N.; Gallagher, G.; Mertz, C.; Peterson, K.; Bagnell, J.A.; Hebert, M.; Dey, A.K.; Srinivasa, S. Planning-based prediction for pedestrians. In Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), St. Louis, MO, USA, 11–15 October 2009; pp. 3931–3936. [CrossRef]

46. Bennewitz, M.; Burgard, W.; Thrun, S. Learning motion patterns of persons for mobile service robots. In Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA), Washington, DC, USA, 11–15 May 2002; Volume 4, pp. 3601–3606. [CrossRef]

47. Kuderer, M.; Kretzschmar, H.; Sprunk, C.; Burgard, W. Feature-Based Prediction of Trajectories for Socially Compliant Navigation. In *Robotics: Science and Systems VIII*; MIT Press: Sydney, Australia, 2013; pp. 193–200. [CrossRef]

48. Ferrer, G.; Sanfeliu, A. Comparative analysis of human motion trajectory prediction using minimum variance curvature. In Proceedings of the 2011 6th ACM/IEEE International Conference on Human-Robot Interaction (HRI), Lausanne, Switzerland, 8–11 March 2011; pp. 135–136. [CrossRef]

49. Kabtoul, M.; Spalanzani, A.; Martinet, P. Towards Proactive Navigation: A Pedestrian-Vehicle Cooperation Based Behavioral Model. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 6958–6964. [CrossRef]

50. Ikeda, T.; Chigodo, Y.; Rea, D.; Zanlungo, F.; Shiomi, M.; Kanda, T. Modeling and Prediction of Pedestrian Behavior based on the Sub-goal Concept. In *Robotics: Science and Systems VIII*; MIT Press: Sydney, Australia, 2013; . [CrossRef]

51. Luber, M.; Tipaldi, G.D.; Arras, K. Place-Dependent People Tracking. *Int. J. Robotic Res.* **2011**, *30*, 280–293. [CrossRef]

52. Ferrer, G.; Garrell, A.; Sanfeliu, A. Robot companion: A social-force based approach with human awareness-navigation in crowded environments. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Tokyo, Japan, 3–7 November 2013; pp. 1688–1694. [CrossRef]

53. Vega, A.; Manso, L.J.; Macharet, D.G.; Bustos, P.; Núñez, P. Socially aware robot navigation system in human-populated and interactive environments based on an adaptive spatial density function and space affordances. *Pattern Recognit. Lett.* **2019**, *118*, 72–84. [CrossRef]

54. Mead, R.; Mataric, M.J. A Probabilistic Framework for Autonomous Proxemic Control in Situated and Mobile Human-Robot Interaction. In Proceedings of the HRI '12, Seventh Annual ACM/IEEE International Conference on Human-Robot Interaction, Boston, MA, USA, 5–8 March 2012; Association for Computing Machinery: New York, NY, USA, 2012; pp. 193–194. [CrossRef]

55. Mead, R.; Atrash, A.; Matarić, M.J. Proxemic Feature Recognition for Interactive Robots: Automating Metrics from the Social Sciences. In Proceedings of the International Conference on Software Reuse, Pohang, Republic of Korea, 13–17 June 2011.

56. Svenstrup, M.; Tranberg, S.; Andersen, H.J.; Bak, T. Pose estimation and adaptive robot behaviour for human–robot interaction. In Proceedings of the 2009 IEEE International Conference on Robotics and Automation (ICRA), Kobe, Japan, 12–17 May 2009; pp. 3571–3576. [CrossRef]

57. Castro-González, A.; Shiomi, M.; Kanda, T.; Salichs, M.A.; Ishiguro, H.; Hagita, N. Position prediction in crossing behaviors. In Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Taipei, Taiwan, 18–22 October 2010; pp. 5430–5437. [CrossRef]

58. Ratsamee, P.; Mae, Y.; Ohara, K.; Takubo, T.; Arai, T. Human–robot collision avoidance using a modified social force model with body pose and face orientation. *Int. J. Humanoid Robot.* **2013**, *10*, 1350008. [CrossRef]

59. Ferrer, G.; Sanfeliu, A. Proactive Kinodynamic Planning using the Extended Social Force Model and Human Motion Prediction in Urban Environments. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Chicago, IL, USA, 14–18 September 2014; pp. 1730–1735. [CrossRef]

60. Farina, F.; Fontanelli, D.; Garulli, A.; Giannitrapani, A.; Prattichizzo, D. When Helbing meets Laumond: The Headed Social Force Model. In Proceedings of the 2016 IEEE 55th Conference on Decision and Control (CDC), Las Vegas, NV, USA, 12–14 December 2016; pp. 3548–3553. [CrossRef]

61. van den Berg, J.; Guy, S.J.; Lin, M.; Manocha, D. Reciprocal n-Body Collision Avoidance. In *Proceedings of the Robotics Research*; Pradalier, C., Siegwart, R., Hirzinger, G., Eds.; Springer Berlin/Heidelberg, Germany, 2011; pp. 3–19.

62. Bera, A.; Kim, S.; Randhavane, T.; Pratapa, S.; Manocha, D. GLMP- realtime pedestrian path prediction using global and local movement patterns. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 5528–5535. [CrossRef]

63. Luo, Y.; Cai, P.; Bera, A.; Hsu, D.; Lee, W.S.; Manocha, D. PORCA: Modeling and Planning for Autonomous Driving Among Many Pedestrians. *IEEE Robot. Autom. Lett.* **2018**, *3*, 3418–3425. [CrossRef]

64. Kim, S.; Guy, S.J.; Liu, W.; Wilkie, D.; Lau, R.W.; Lin, M.C.; Manocha, D. BRVO: Predicting pedestrian trajectories using velocity-space reasoning. *Int. J. Robot. Res.* **2015**, *34*, 201–217. [CrossRef]

65. Xu, M.; Xie, X.; Lv, P.; Niu, J.; Wang, H.; Li, C.; Zhu, R.; Deng, Z.; Zhou, B. Crowd Behavior Simulation with Emotional Contagion in Unexpected Multihazard Situations. *IEEE Trans. Syst. Man, Cybern. Syst.* **2021**, *51*, 1567–1581. [CrossRef]

66. Curtis, S.; Guy, S.J.; Zafar, B.; Manocha, D. Virtual Tawaf: A Velocity-Space-Based Solution for Simulating Heterogeneous Behavior in Dense Crowds. In *Modeling, Simulation and Visual Analysis of Crowds*; The International Series in Video Computing; Springer: Berlin/Heidelberg, Germany, 2013; Volume 11, pp. 181–209.

67. Alahi, A.; Goel, K.; Ramanathan, V.; Robicquet, A.; Fei-Fei, L.; Savarese, S. Social LSTM: Human Trajectory Prediction in Crowded Spaces. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 961–971. [CrossRef]

68. Vemula, A.; Muelling, K.; Oh, J. Social Attention: Modeling Attention in Human Crowds. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 4601–4607. [CrossRef]

69. Foka, A.; Trahania, P. Probabilistic Autonomous Robot Navigation in Dynamic Environments with Human Motion Prediction. *I. J. Soc. Robot.* **2010**, *2*, 79–94. [CrossRef]

70. Svenstrup, M.; Bak, T.; Andersen, H.J. Trajectory planning for robots in dynamic human environments. In Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Taipei, Taiwan, 18–22 October 2010; pp. 4293–4298. [CrossRef]

71. Park, J.J.; Kuipers, B. A smooth control law for graceful motion of differential wheeled mobile robots in 2D environment. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, 9–13 May 2011; pp. 4896–4902. [CrossRef]

72. Park, J.J.; Johnson, C.; Kuipers, B. Robot navigation with model predictive equilibrium point control. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vilamoura-Algarve, Portugal, 7–12 October 2012; pp. 4945–4952. [CrossRef]

73. Rios-Martinez, J.; Spalanzani, A.; Laugier, C. Understanding human interaction for probabilistic autonomous navigation using Risk-RRT approach. In Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), San Francisco, CA, USA, 25–30 September 2011; pp. 2014–2019. [CrossRef]

74. Ferrer, G.; Sanfeliu, A. Bayesian Human Motion Intentionality Prediction in urban environments. *Pattern Recognit. Lett.* **2014**, *44*, 134–140. [CrossRef]

75. Palm, R.; Chadalavada, R.; Lilienthal, A.J. Recognition of human–robot motion intentions by trajectory observation. In Proceedings of the 2016 9th International Conference on Human System Interactions (HSI), Portsmouth, UK, 6–8 July 2016; pp. 229–235. [CrossRef]

76. Ferrer, G.; Zulueta, A.; Cotarelo, F.; Sanfeliu, A. Robot social-aware navigation framework to accompany people walking side-by-side. *Auton. Robot.* **2017**, *41*, 775–793. [CrossRef]

77. Khambhaita, H.; Alami, R. A Human-Robot Cooperative Navigation Planner. In Proceedings of the 2017 ACM/IEEE International Conference, Vienna, Austria, 6–9 March 2017, pp. 161–162. [CrossRef]

78. Kabtoul, M.; Spalanzani, A.; Martinet, P. Proactive Furthermore, Smooth Maneuvering For Navigation Around Pedestrians. In Proceedings of the 2022 International Conference on Robotics and Automation (ICRA), Philadelphia, PA, USA, 23–27 May 2022; pp. 4723–4729. [CrossRef]

79. Chen, Y.F.; Liu, M.; Everett, M.; How, J.P. Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 285–292. [CrossRef]

80. Chen, C.; Liu, Y.; Kreiss, S.; Alahi, A. Crowd-Robot Interaction: Crowd-Aware Robot Navigation with Attention-Based Deep Reinforcement Learning. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 6015–6022. [CrossRef]

81. Liu, L.; Dugas, D.; Cesari, G.; Siegwart, R.; Dubé, R. Robot Navigation in Crowded Environments Using Deep Reinforcement Learning. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 25–29 October 2020; pp. 5671–5677. [CrossRef]

82. Henry, P.; Vollmer, C.; Ferris, B.; Fox, D. Learning to navigate through crowded environments. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation (ICRA), Anchorage, AK, USA, 3–8 May 2010; pp. 981–986. [CrossRef]

83. Kim, B.; Pineau, J. Socially Adaptive Path Planning in Human Environments Using Inverse Reinforcement Learning. *Int. J. Soc. Robot.* **2016**, *8*, 51–66. [CrossRef]

84. Chen, Y.F.; Everett, M.; Liu, M.; How, J.P. Socially aware motion planning with deep reinforcement learning. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 1343–1350. [CrossRef]

85. Everett, M.; Chen, Y.F.; How, J.P. Motion Planning Among Dynamic, Decision-Making Agents with Deep Reinforcement Learning. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 3052–3059. [CrossRef]

86. Samsani, S.S.; Muhammad, M.S. Socially Compliant Robot Navigation in Crowded Environment by Human Behavior Resemblance Using Deep Reinforcement Learning. *IEEE Robot. Autom. Lett.* **2021**, *6*, 5223–5230. [CrossRef]

87. Hu, Z.; Zhao, Y.; Zhang, S.; Zhou, L.; Liu, J. Crowd-Comfort Robot Navigation Among Dynamic Environment Based on Social-Stressed Deep Reinforcement Learning. *Int. J. Soc. Robot.* **2022**, *14*, 913–929. [CrossRef]

88. Dugas, D.; Nieto, J.; Siegwart, R.; Chung, J.J. NavRep: Unsupervised Representations for Reinforcement Learning of Robot Navigation in Dynamic Human Environments. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 7829–7835. [CrossRef]

89. Gil, O.; Garrell, A.; Sanfeliu, A. Social Robot Navigation Tasks: Combining Machine Learning Techniques and Social Force Model. *Sensors* **2021**, *21*, 7087. [CrossRef] [PubMed]

90. Francis, A.; Faust, A.; Chiang, H.T.L.; Hsu, J.; Kew, J.C.; Fiser, M.; Lee, T.W.E. Long-Range Indoor Navigation with PRM-RL. *IEEE Trans. Robot.* **2020**, *36*, 1115–1134. [CrossRef]

91. Chen, Y.; Liu, C.; Shi, B.E.; Liu, M. Robot Navigation in Crowds by Graph Convolutional Networks with Attention Learned From Human Gaze. *IEEE Robot. Autom. Lett.* **2020**, *5*, 2754–2761. [CrossRef]

92. Gao, X.; Sun, S.; Zhao, X.; Tan, M. Learning to Navigate in Human Environments via Deep Reinforcement Learning. In Proceedings of the Neural Information Processing, Sydney, Australia; 8–11 December 2019; Gedeon, T., Wong, K.W., Lee, M., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 418–429.

93. Long, P.; Fan, T.; Liao, X.; Liu, W.; Zhang, H.; Pan, J. Towards Optimally Decentralized Multi-Robot Collision Avoidance via Deep Reinforcement Learning. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018; pp. 6252–6259. [CrossRef]

94. Gromniak, M.; Stenzel, J. Deep Reinforcement Learning for Mobile Robot Navigation. In Proceedings of the 2019 4th Asia-Pacific Conference on Intelligent Robot Systems (ACIRS), Nagoya, Japan, 13–15 July 2019; pp. 68–73. [CrossRef]

95. Shi, H.; Shi, L.; Xu, M.; Hwang, K.S. End-to-End Navigation Strategy with Deep Reinforcement Learning for Mobile Robots. *IEEE Trans. Ind. Inform.* **2020**, *16*, 2393–2402. [CrossRef]

96. Li, M.; Jiang, R.; Ge, S.; Lee, T. Role Playing Learning for Socially Concomitant Mobile Robot Navigation. *CAAI Trans. Intell. Technol.* **2018**, *3*, 49–58. [CrossRef]

97. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.A.; Fidjeland, A.; Ostrovski, G.; et al. Human-level control through deep reinforcement learning. *Nature* **2015**, *518*, 529–533. [CrossRef]

98. Lee, M.F.R.; Yusuf, S.H. Mobile Robot Navigation Using Deep Reinforcement Learning. *Processes* **2022**, *10*, 2748. [CrossRef]

99. Pfeiffer, M.; Schaeuble, M.; Nieto, J.; Siegwart, R.; Cadena, C. From Perception to Decision: A Data-driven Approach to End-to-end Motion Planning for Autonomous Ground Robots. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Piscataway, NJ, USA, 29 May–3 June 2017; pp. 1527–1533.

100. Pfeiffer, M.; Shukla, S.; Turchetta, M.; Cadena, C.; Krause, A.; Siegwart, R.; Nieto, J. Reinforced Imitation: Sample Efficient Deep Reinforcement Learning for Mapless Navigation by Leveraging Prior Demonstrations. *IEEE Robot. Autom. Lett.* **2018**, *3*, 4423–4430. [CrossRef]

101. Tai, L.; Paolo, G.; Liu, M. Virtual-to-real deep reinforcement learning: Continuous control of mobile robots for mapless navigation. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 31–36. [CrossRef]

102. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. TensorFlow: A system for large-scale machine learning. *arXiv* **2016**, arXiv:1605.08695.

103. Achiam, J.; Held, D.; Tamar, A.; Abbeel, P. Constrained Policy Optimization. In Proceedings of the 34th International Conference on Machine Learning (ICML), Sydney, Australia, 6–11 August 2017; Volume 70, pp. 22–31.

104. Fan, T.; Cheng, X.; Pan, J.; Manocha, D.; Yang, R. CrowdMove: Autonomous Mapless Navigation in Crowded Scenarios. *arXiv* **2018**, arXiv:1807.07870.

105. Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; Klimov, O. Proximal Policy Optimization Algorithms. *arXiv* **2017**, arXiv:1707.06347.

106. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. *arXiv* **2015**, arXiv:1512.03385.

107. Gao, W.; Hsu, D.; Lee, W.S.; Shen, S.; Subramanian, K. Intention-Net: Integrating Planning and Deep Learning for Goal-Directed Autonomous Navigation. In Proceedings of the Conference on Robot Learning, Mountain View, CA, USA, 13–15 November 2017.

108. Pokle, A.; Martín-Martín, R.; Goebel, P.; Chow, V.; Ewald, H.M.; Yang, J.; Wang, Z.; Sadeghian, A.; Sadigh, D.; Savarese, S.; et al. Deep Local Trajectory Replanning and Control for Robot Navigation. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 5815–5822. [CrossRef]

109. Pérez-D'Arpino, C.; Liu, C.; Goebel, P.; Martín-Martín, R.; Savarese, S. Robot Navigation in Constrained Pedestrian Environments using Reinforcement Learning. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 1140–1146. [CrossRef]

110. Choi, J.; Dance, C.; Kim, J.E.; Park, K.S.; Han, J.; Seo, J.; Kim, M. Fast Adaptation of Deep Reinforcement Learning-Based Navigation Skills to Human Preference. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 30 May–4 June 2020; pp. 3363–3370. [CrossRef]

111. Baghi, B.H.; Dudek, G. Sample Efficient Social Navigation Using Inverse Reinforcement Learning. *arXiv* **2021**, arXiv:2106.10318.

112. Ziebart, B.D.; Maas, A.L.; Bagnell, J.A.; Dey, A.K. Maximum Entropy Inverse Reinforcement Learning. In Proceedings of the AAAI Conference on Artificial Intelligence, Chicago, IL, USA, 13–17 July 2008.

113. Pérez-Higueras, N.; Ramón-Vigo, R.; Caballero, F.; Merino, L. Robot local navigation with learned social cost functions. In Proceedings of the 2014 11th International Conference on Informatics in Control, Automation and Robotics (ICINCO), Vienna, Austria, 1–3 September 2014; Volume 2, pp. 618–625. [CrossRef]

114. Gerkey, B.; Konolige, K. Planning and control in unstructured terrain. In Proceedings of the ICRA Workshop on Path Planning on Costmaps, Pasadena, CA, USA, 19–23 May 2008.

115. Vasquez, D.; Okal, B.; Arras, K.O. Inverse Reinforcement Learning algorithms and features for robot navigation in crowds: An experimental comparison. In Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Chicago, IL, USA, 14–18 September 2014; pp. 1341–1346. [CrossRef]

116. Kretzschmar, H.; Spies, M.; Sprunk, C.; Burgard, W. Socially compliant mobile robot navigation via inverse reinforcement learning. *Int. J. Robot. Res.* **2016**, *35*, 1289–1307. [CrossRef]

117. Wang, Y.; Kong, Y.; Ding, Z.; Chi, W.; Sun, L. NRTIRL Based NN-RRT* Path Planner in Human-Robot Interaction Environment. In Proceedings of the Social Robotics, Florence, Italy, 13–16 December 2022; Cavallo, F., Cabibihan, J.J., Fiorini, L., Sorrentino, A., He, H., Liu, X., Matsumoto, Y., Ge, S.S., Eds.; Springer: Cham, Switzerland, 2022; pp. 496–508.

118. Ramachandran, D.; Amir, E. Bayesian Inverse Reinforcement Learning. In Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI), Hyderabad, India, 6–12 January 2007; pp. 2586–2591.

119. Okal, B.; Arras, K.O. Learning socially normative robot navigation behaviors with Bayesian inverse reinforcement learning. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 6–21 May 2016; pp. 2889–2895. [CrossRef]

120. Dugas, D.; Nieto, J.; Siegwart, R.; Chung, J.J. IAN: Multi-Behavior Navigation Planning for Robots in Real, Crowded Environments. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 24 October 2020–24 January 2021; pp. 11368–11375. [CrossRef]

121. Vega-Magro, A.; Gondkar, R.; Manso, L.; Núñez, P. Towards efficient human–robot cooperation for socially-aware robot navigation in human-populated environments: The SNAPE framework. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 3169–3174. [CrossRef]

122. Chen, Z.; Song, C.; Yang, Y.; Zhao, B.; Hu, Y.; Liu, S.B.; Zhang, J. Robot Navigation Based on Human Trajectory Prediction and Multiple Travel Modes. *Appl. Sci.* **2018**, *8*, 2205. . [CrossRef]

123. Freitas, R.S.D.; Romero-Garcés, A.; Marfil, R.; Vicente-Chicote, C.; Cruz, J.M.; Inglés-Romero, J.F.; Bandera, A. QoS Metrics-in-the-Loop for Better Robot Navigation. In *Advances in Intelligent Systems and Computing, Proceedings of the WAF, Alcala de Henares, Spain, 19–20 November 2020*; Springer: Berlin/Heidelberg, Germany, 2020; Volume 1285, pp. 94–108.

124. Bozhinoski, D.; Wijkhuizen, J. Context-based navigation for ground mobile robot in semi-structured indoor environment. In Proceedings of the 2021 Fifth IEEE International Conference on Robotic Computing (IRC), Taichung, Taiwan, 15–17 November 2021; pp. 82–86. [CrossRef]

125. Bustos, P.; Manso, L.J.; Bandera, A.; Rubio, J.P.B.; García-Varea, I.; Martínez-Gómez, J. The CORTEX cognitive robotics architecture: Use cases. *Cogn. Syst. Res.* **2019**, *55*, 107–123. [CrossRef]

126. Marfil, R.; Romero-Garcés, A.; Rubio, J.P.B.; Manso, L.J.; Calderita, L.V.; Bustos, P.; Bandera, A.; García-Polo, J.; Fernández, F.; Voilmy, D. Perceptions or Actions? Grounding How Agents Interact within a Software Architecture for Cognitive Robotics. *Cogn. Comput.* **2020**, *12*, 479–497. [CrossRef]

127. Luber, M.; Tipaldi, G.D.; Arras, K. Better models for people tracking. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, 9–13 May 2011; pp. 854–859. [CrossRef]