Mapless Navigation among Dynamics with Social-safety-awareness: a reinforcement learning approach from 2D laser scans

Jun Jin*,†, Nhat M. Nguyen†, Nazmus Sakib*,†, Daniel Graves†, Hengshuai Yao† and Martin Jagersand*

Abstract—We propose a method to tackle the problem of mapless collision-avoidance navigation where humans are present using 2D laser scans. Our proposed method uses ego-safety to measure collision from the robot’s perspective while social-safety to measure the impact of our robot’s actions on surrounding pedestrians. Specifically, the social-safety part predicts the intrusion impact of our robot’s action into the interaction area with surrounding humans. We train the policy using reinforcement learning on a simple simulator and directly evaluate the learned policy in Gazebo and real robot tests. Experiments show the learned policy can be smoothly transferred without any fine tuning. We observe that our method demonstrates time-efficient path planning behavior with high success rate in mapless navigation tasks. Furthermore, we test our method in a navigation among dynamic crowds task considering both low and high volume traffic. Our learned policy demonstrates cooperative behavior that actively drives our robot into traffic flows while showing respect to nearby pedestrians. Evaluation videos are at https://sites.google.com/view/ssw-batman

I. INTRODUCTION

The problem of autonomously navigating a robot in a map-unknown (mapless) environment while avoiding both static and dynamic obstacles, is very important but challenging in applications like delivery robots, indoor service robots, etc. The path planning and static obstacle avoidance part in this problem are often formatted as mapless navigation [1], [2] where a robot is driven by observed sensory data from the unknown environment, assuming the relative pose from robot to target is given by a third party localization module [1]. The dynamic obstacle avoidance part in this problem is more complicated since it requires future prediction on unknown surrounding dynamics like moving pedestrians, vehicles or other robots [3], [4]. As complexity of surrounding dynamics increases, the prediction may result in occupying a large portion of free space, successively causing no solution in path planning, namely the freezing robot problem [5]. Moreover when the moving obstacles are human, not only collision avoidance, but also human-awareness [6], [7] should be considered. While recent approaches using multi-modal sensing [8] or high-end object/pedestrian detection pipelines [9], [10], [4], have been proposed to tackle part of the problem, we are still curious to ask: is it possible to solve for all parts purely based on 2D laser scans?

Recent works on deep reinforcement learning have proven the capability of using 2D laser scans in mapless navigation[2], [11] and multi-agent / dense crowd[12], [13] collision avoidance. Such methods share a similar reward structure by adding a reaching target reward and a collision penalty, but differ in the training process. For example, in a mapless navigation task, policy training is done by using a fixed number of floor plans[2], [11], while in multi-agent / dense crowd[12], [13] collision avoidance by adding autonomous agents following a predefined policy. By adding the collision penalty and training with exploration in randomized multi-agent behaviors, the learned policy shows cooperative behavior that solves the freezing robot problem. This collision penalty only measures near impact of our robot with other agents and stationary obstacles. However, when human are present nearby, the inverse directional impact on human safety should also be considered: namely human-awareness [6], [7].

Inspired by the concept of egocentric and allocentric spatial cognition [14], we propose a framework with ego-safety to measure collision from the robot’s perspective while social-safety to measure the impact of our robot’s actions on surrounding pedestrians. Specifically, social-safety is defined as the intrusion into surrounding human’s interaction area considering a look-ahead distance (fig. 2A).

The policy is trained using Deep Deterministic Policy Gradient (DDPG) [15] under our specially designed toy simulator. To enhance our robot’s ability in mapless navigation, compared to other methods [12], [13], [2], [11], we add more complexity in the simulator with obstacles and moving agents of randomized numbers, sizes, shapes, and locations.
Our learned policy is then directly evaluated in Gazebo and real robot test under randomized complex indoor environments with both moving obstacles and walking pedestrians. Experimental results show our learned policy demonstrates time-efficient path planning behavior with high success rate in mapless navigation tasks, as well as, cooperative behavior that actively drives the robot into the crowd flows while showing respect to nearby pedestrians in navigation among dynamic crowds tasks.

II. RELATED WORKS

This work is inspired by researches in the following topics.

Mapless Navigation: The problem of mapless navigation [2], [11] is interesting at first sight, since autonomously navigating a robot in unknown environments is exciting, however problematic in the definition of target since it assumes a third party localization solution is given. From this aspect, visual mapless navigation [16], [17], [18], [19] seems more meaningful because the target can be practically given by reference images, however, it’s challenging to learn local motion control behaviors with respect to dynamic collision avoidance simply from visual inputs. One alternative way is a hierarchical approach [18] which has both a high-level global path planner and low-level motion controller. The high-level global path planner can be obtained from visual navigation or simply GPS waypoints, that enables long-term navigation. The low-level motion controller learns both static and dynamic collision avoidance behavior while reaching a local target defined by the global path planner. The ability of local navigation among dynamics is exactly our consideration in this paper.

Collision avoidance among dynamics: Since originally formulated as a multi-agent collision avoidance problem, prior works in centralized approaches [20] assume knowing the full state and intention of all agents via a centralized communication system, which is obviously not applicable considering our obstacles are pedestrians or non-communicable vehicles. So decentralized approaches are proposed. A traditional method is the Optimal Reciprocal Collision Avoidance (ORCA) [21], which is a one-step response method that computes a local jointly collision-free velocity by accounting for other agent’s velocity reflected on the robot itself. ORCA relies on an accurate estimation of all agents’ state which limits its real world application, however, proven to be a computationally efficient algorithm as pedestrian simulators [22]. Since OCRA does not consider future evolution of surrounding agents’ motion, the generated trajectory is often short-sighted and non-efficient. Chen et al. [3] propose CADRL to estimate a value function on the two-agent collision avoidance problem which demonstrate time efficient behavior. CADRL can be extended to handle a fixed number of surrounding agents depending on how we train the policy. Their later work [23] uses a LSTM module to handle arbitrary number of pedestrians and evaluate on a real robot where the pedestrian detection part is based on a high-end multi-sensor fusion module. As we can see, all the above mentioned approaches need explicit real-time detection and estimation of surrounding agents or pedestrians, which may not be robust in practice.

Social-aware/ Human-aware robot navigation: Following the same approach as CADRL, social-aware navigation problem is further considered which views more human-robot interactions in the navigation behavior. Chen et al. [9] propose SACADRL which considers human-like social norm behaviorootnote{e.g., left/right hand traffic rule in crossing, passing and overtaking.} by adding a complex social norm reward. Similarly, Tai et al. [24] train a social-compliant policy from RGB-D raw data inputs. To push our discussion further away, the above mentioned concepts actually align with prior works in human-aware robot navigation [6], [7] where social rules (norm), as well as, human comfort are considered. Apart from these complicated considerations, a simplified version in social-aware navigation is quite practical, that is to equip the robot with cooperative ability to solve the freezing robot problem [5]. Pfeiffer et al. [10] propose a framework to learn cooperative motion planning behavior by modeling human-robot interaction using maximum entropy methods. Chen et al. [4] jointly model human-robot and human-human interaction followed by a similar value function estimation in CADRL [3]. Also, such methods depend on explicit pedestrian detection. In this work, we propose social-safety as another simplified consideration of human-awareness.

Navigation learning from observations: As discussed in Section 1, recently end-to-end approaches are proposed to learn the navigation strategy from 2D laser scan readings. Long et al. [12] propose a framework to train a multi-robot collision avoidance policy using PPO [25] which maps 3 frames of historical laser scans to continuous robot actions. To overcome the training challenge arising from large state dimensions (compared to [3]), they utilize curriculum learning strategy under different scenarios. Interestingly, they also run successful tests when directly applying the learned policy in dense crowd navigation tasks [13]. Apart from using 2D laser scans, navigation learning from visual inputs, a.k.a. visual navigation problem [17], [18] has also been intensively studied.

III. APPROACH

A. Problem Formation

We formulate the problem as a Partially-Observable Markov Decision Process (POMDP) defined by a tuple $(S, A, P, R, \Omega, O)$, where $S$ is the state space, $A$ is the action space, $P$ is the state transition model, $R$ is the reward function, $\Omega$ is the observation space, $O$ is the probability function defining how observations are obtained from the underlying environment state $(o^t \sim O(s^t))$. The robot can only observe laser scan $o^t_g$ from surroundings and relative location $o^t_m$ to the target. We consider a 270$^\circ$ laser scanner with a resolution of 0.25$^\circ$ and a distance range between 0.1 meter and 10 meter.

We define $o^t = [o^t_m, o^t_g]$. $o^t_m$ is a motion feature which encodes the laser scan response changes along time axis due to surrounding motions. Robot action $a^t$ is sampled from a
stochastic policy $\pi$ given observation $o^t$: $a^t \sim \pi_\theta(a^t|o^t)$, where $a^t = [a^t_x, a^t_y]$ is within range $[-1.5,1.5]$. Considering a nonholonomic kinematics model, $a^t$ is further converted to linear and angular velocity $[v^t_x, v^t_\theta]$ with $v^t_x = \sqrt{a^t_x^2 + a^t_y^2}$ and $v^t_\theta = \arctan(a^t_y/a^t_x) \in [-\pi, \pi]$.

1) Motion features: $o^t_n$ plays as a prediction feature for learning a cooperative path planning behavior among dynamics. We consider two issues in constructing $o^t_n$ from historical laser scans $\{o^i_{-k}, ..., o^i_j\}$: firstly disentangling robot ego motion in the laser scans. We observe laser scan response can be changed by either robot ego motion or surrounding motions. By disentangling robot ego motion, our motion feature $o^t_n$ will only encode surrounding motions, which helps our policy training. While there are many advanced methods to detect dynamic objects from laser scans [26], [27], we further observe that during a small k+1 step duration, robot rotation affects the most compared to transnational motions. For the efficiency of simulator training, we simply calibrate previous laser scan $o^t_{-i}$ based on the difference between robot heading angles at time t-i and t by a shift operation in the scan array. Fig. 2A shows the comparison between calibrated and uncalibrated results. Secondly normalizing scan readings to have more attention on nearby motion dynamics than far away ones. This is done by a generalized logistic function [28] with parameters $A=0$, $K=1$, $C=0.25$, $Q=1$, $B=0.4$, $V=1$, $M=0.8$.

2) Reward function: The reward at each step is obtained from system state $s_t$ which is fully accessible in simulator training. Our reward function is a sum of ego-safety $R_e$, social-safety $R_s$ and reaching target $R_y$ rewards:

$$R(s_t) = R_e(s_t) + R_s(s_t) + R_y(s_t)$$

We define the ego-safety zone (fig. 2B) of the robot as a circle around with radius $(r_i = 0.4)$, where $r_i$ is the physical dimension of the robot. Given a set of nearby pedestrians $\{p^j\}$ and obstacles $\{B_k\}$, their closest distance to robot is $d^i$. $R_e(s_t)$ is defined as:

$$R_e(s_t) = \begin{cases} -10 & \text{if collision happens} \\ -0.25(1 - \frac{d^i}{r_i + 0.4}) & \text{else if } d^i < r_i + 0.4 \\ 0 & \text{otherwise} \end{cases}$$

We define the social-safety zone (fig. 2C) of each pedestrian as their interaction region stretching along current moving direction with a minimum safe headway distance in $\Delta t$. For computational efficiency, only pedestrians within 5m are considered. This zone is represented as a rectangle defined in pedestrian’s reference frame with $width = -r_{object}$ and $height = r_{object}/2 + d_{min} + \Delta t \cdot v_{object}$, where $r_{object}$ is the radius of pedestrian bounding circle, $d_{min} = 0.5m$ is the minimum safe distance, $v_{object}$ is the current speed. We set $\Delta t = 0.77s$ in training. The safety zones are then projected back into the world frame. Then we also construct the ego robot’s social-safety zone likewise, and check if it intersects with that of other moving pedestrians. If intersects, we count it as one violation. Now we define $R_s(s_t)$ as:

$$R_s(s_t) = -0.1 \frac{\text{number of violations}}{\text{total number of pedestrians}}$$

Let $p^k$ be position of robot and $p^\pi$ be the target. The last term $R_y(s_t)$ is defined as:

$$R_y(s_t) = \begin{cases} +10 & \text{if reach the target} \\ -0.01 + \frac{\|p^\pi - p^t\|}{\|p^k - p^\pi\|} & \text{otherwise} \end{cases}$$

B. Parameterization

Our special consideration in parameterization is to make our network less sensitive to object shapes, which are mostly encoded in data along column axis in $o^t_n$. So we design a large kernel size on the column direction followed by a max-pooling layer. We use DDPG [15] for training. Our actor and critic network structure are illustrated in fig. 2C.
C. Training the Policy

1) Toy simulator: The policy is trained on our hand designed toy simulator (fig. 2A) that runs a laser scanner at 40Hz and an inner differential drive controller at 20Hz. Historical 40 frames of laser scan are used to construct $o^t_s$. The number, size and shape of static obstacles are randomized in simulation. Pedestrian behavior is also randomized with the following considerations: number of pedestrians in the scene, each pedestrian’s current pose / velocity, each pedestrian’s desired target and direction, each pedestrian’s geometric shape and size, the behavior of stop-and-go, new walk-in pedestrians from random directions. Each pedestrian’s behavior is controlled using ORCA [29] but ignores the robot since otherwise pedestrians will always avoid the robot.

2) Training strategy: We train an ego policy by only including the ego-safety reward and a social policy by adding both ego-safety and social-safety rewards for comparison. In practice, the training process is very tedious using DDPG. We format our training in two stages. The social policy is later trained based on a learned ego policy parameters.

IV. EVALUATION

We evaluate the learned policy on three tasks that all require the robot reaching a target: (1) Mapless path planning: The robot is required to find a path while avoiding obstacles in an unknown environment. There are no pedestrians in this task. (2) Navigation among dynamic crowds: Given a dynamic crowd with random behaviors, the robot needs to avoid moving pedestrians while showing awareness of social-safety without the ‘freezing robot’ behavior. Different sizes of crowds will be tested. (3) Mapless path planning + navigation among dynamic crowds: A combination of the above two.

Baselines: To the authors’ best knowledge, there are no current methods that both tackling mapless navigation and collision avoidance among dynamic crowds using 2D laser scans. However we still hand designed two baselines for comparison. (1) The greedy baseline takes input of laser scan and mimics human intuition on reaching the target while avoiding collisions locally. In the planner we take robot’s direction and the difference of angle between the robot and the destination. We apply 1D convolution window with laser scan and find the index having the higher convoluted value with a p controller to turn the robot to the direction. (2) The CADRL [3] baseline takes the full state of surrounding pedestrians as input. It’s not designed for avoiding static obstacles in path planning, so it’s only used in navigation among the dynamics crowds. Though it can’t handle laser scan observations, we still feed in full states of all humans obtained from simulator, while testing our methods using only laser scans.

Metrics: We designed four metrics for quantitative evaluation: (1) Success Rate (%) after 10 runs in random settings. (2) Arriving Time (s). (3) Ego Score (0-100) to measure ego-safety awareness of robot. Let $k$ be the number of ego-safety violation steps, and $N$ be total steps to reach the target, \[ Ego\text{-}\text{Score} = (1 - k/N) \times 100. \] (4) Social Score (0-100) to evaluate social-safety awareness of nearby human. Let $m$ be the number of social-safety violation steps, \[ Social\text{-}\text{Score} = (1 - m/N) \times 100. \]

We compare both our ego policy and social policy to the above two baselines. Experiments are conducted in both our toy simulator and Gazebo for quantitative evaluation, while in real world tests for qualitative evaluation (fig. 3).

A. In the toy simulator

1) On mapless path planning task: In this task, we fix the target while generating 10 random maps that only have static obstacles. Since CADRL can’t handle this task, we only compare with greedy baseline (Table 1). Results show both our ego and social policy reach the target more efficiently with a higher success rate.

|                           | Success Rate | Arriving Time | Ego Score |
|---------------------------|--------------|---------------|-----------|
| Baseline                  | 80 %         | 7.0 ± 1.6 s   | 100       |
| (Ours) Ego Policy         | 100 %        | 3.1 ± 0.3 s   | 100       |
| (Ours) Social Policy      | 100 %        | 3.3 ± 1.3 s   | 100       |

Fig. 4: Comparison between our method (ego policy, social policy) and the baseline in toy simulator. Left: results on the navigation among dynamic crowds task. right: results of the combined task. We observe the CADRL baseline typically has a passive ‘wait and go’ behavior which is not efficient and easily get trapped by crowds. In comparison, both our policies show more cooperative behaviors. This cooperative behavior helps the robot reach target more efficiently with higher success rate. While all methods achieves high ego score, the social policy shows better performance in social-safety awareness. By observation, the ego policy is more aggressive compared to our social policy. Detailed videos are on our website.

Fig. 3: The policy is trained using our toy simulator (A) and tested under various settings on our toy simulator, Gazebo(B) and a Jackal robot(C) in real world environment respectively.
Fig. 5: We evaluate the mapless navigation ability of our learned policy in three maps from simple to complex. Complexity increases from map 1 to map 3 by blocking key passages and adding more random obstacles. All tasks require the robot reaching the target (orange square) without knowing the map. We test in two scenarios: fixed target and random targets. Each scenario for each map runs 10 times on each method. Our results show the greedy baseline always fails as the map is too complex compared to toy simulator. All methods achieve ego score 100 in successful runs. We observe both ego and social policy show the behavior of slowing down when obstacles are highly aggregated. The ego policy performs better since it positively runs via small gaps between obstacles. However, it easily gets trapped into dead loops, where it will run circle loops instead of tracing back to find a way out. As comparison, the social policy is more cautious even towards static obstacles. It always runs into wider open spaces and won’t take the risk to go through small gaps, in which case, it will switch to other directions. If the gaps in all directions are small, it will run a circle loop. This is the main reason it fails.

2) On navigation among dynamics task: We evaluate using 10 random crowds for each method. Each crowd has at least 8 pedestrians in a 5m × 5m area with a chance of walking humans from a random direction. All pedestrians have no sense of the robot so they won’t avoid it. We compare our methods to both baselines (Fig. 4).

3) The combined task settings: Now we combine the above two task settings. Since CADRL can’t work in mapless path planning, we only compare with the greedy baseline. Results are plotted in Fig. 4.

B. In Gazebo environment

1) On the mapless path planning task: We design three maps from simple to complex in Gazebo. We firstly evaluate on a fixed target. Then we pick 10 targets randomly and run all methods to compare. Results are plotted in Fig. 5.

2) On navigation among dynamics crowds: To ground our evaluation, we designed four scenarios of dynamic crowds (Fig. 6) with crowd size 4, 8 and 12. Each method on each scenario / crowd size combination needs to run 10 times to get evaluation results, as shown in Fig. 7.

C. In real world tests

We also conduct qualitative evaluations on a Jackal robot. Since mapless navigation needs a third party localization to define the target, we simply rely on robot’s built-in odometry module which computes robot pose using wheel encoders and on-board IMU. Our target location is defined relative to the robot. We do calibration after each run since the odometry error drifts along time. We observe that, even with this coarse localization input, the robot performs relatively well in the following tasks:

1) On mapless path planning task: We design two maps for this task without pedestrians. In each map, the robot needs to navigate to the target while avoiding static obstacles. We also test our method’s performance while human pushing the obstacles. Evaluation results are shown in our video. For static obstacle avoidance in the two maps, both ego and social
We designed four random behaved crowd scenarios considering crowd size 4, 8, 12 in a 5m×5m area. Each scenario under each crowd size is tested 10 times on the baselines and our methods. In all, our methods reaches the target in a shorter time with a higher success rate. We observe the CADRL baseline prefers to move slowly to wait for crowds pass, however, may collide with human approaching from other directions. In comparison, our ego and social policies typically move into the crowd flow while avoiding pedestrians dynamically. In all tests, the social policy has the highest social score. In the four scenarios, the ‘towards’ one is most difficult. Our ego policy has the best performance since it’s more agile running through small gaps. The ‘ahead’ scenario is the easiest one. Both baselines simply move slowly behind the crowds. In contrast, our ego and social policy positively try to pass the crowd ahead. So they are significantly faster than the two baselines. Detailed videos are on the website.

Fig. 6: Comparison between our method and two baselines on navigation among dynamic crowds tasks in Gazebo. We designed four random behaved crowd scenarios considering crowd size 4, 8, 12 in a 5m×5m area. Each scenario under each crowd size is tested 10 times on the baselines and our methods. In all, our methods reaches the target in a shorter time with a higher success rate. We observe the CADRL baseline prefers to move slowly to wait for crowds pass, however, may collide with human approaching from other directions. In comparison, our ego and social policies typically move into the crowd flow while avoiding pedestrians dynamically. In all tests, the social policy has the highest social score. In the four scenarios, the ‘towards’ one is most difficult. Our ego policy has the best performance since it’s more agile running through small gaps. The ‘ahead’ scenario is the easiest one. Both baselines simply move slowly behind the crowds. In contrast, our ego and social policy positively try to pass the crowd ahead. So they are significantly faster than the two baselines. Detailed videos are on the website.

Fig. 7: The four crowd behavior scenarios. A crossing: The robot needs to move through the random crossing flow. B towards: Pedestrians are walking towards the robot. The robot needs to react fast since the relative velocity is large. C ahead: Pedestrians are walking ahead. The robot needs to decide how to pass. D random: All pedestrians’ intention are randomized.

policies perform well. For pushing obstacles setting, the ego policy agilely avoids human kicking boxes while the social policy behaves more cautiously.

2) On navigation among dynamics task: We also test the four crowd scenarios (Fig. 7) using the social policy. Fig. 8 shows the observed robot behavior.

V. CONCLUSION

We propose a method to tackle the problem of mapless collision-avoidance navigation where humans are present using 2D laser scans by the formation of ego safety and social safety. Extensive experiments were conducted with both quantitative and qualitative evaluation. Results show our method can demonstrate cooperative path planning behavior by predicting the future motion of humans, while considering

the impact of the robot actions on surrounding humans, namely the awareness of social-safety.

Though some success is achieved in this project, other practical considerations are worth pointing out: (1) To define the target, mapless navigation relies on a third party localization module. This is usually unrealistic considering mapless navigation is designed for unknown environments. As discussed in section 2, a high level abstract path planner with vision sensor support may solve the problem. (2) It’s still hard to train the policy. Though [12], [13] propose a multi-stage training that may help, it lacks theoretical guarantees. We expect finite horizon RL solutions since path planning problem typically needs to consider multiple steps ahead. (3) We observe the learned policy is still sensitive to object shapes, surface smoothness. More randomization in training should solve the problem.

The authors would like to give thanks to all our colleagues participating in the real world tests.
REFERENCES

[1] C. Giovannangeli, P. Gaussier, and G. Desilles, “Robust mapless outdoor navigation based on vision-based localization,” in 2006 IEEE/RSJ international conference on intelligent robots and systems. IEEE, 2006, pp. 3293–3300.

[2] L. Tai, G. Paolo, and M. Liu, “Virtual-to-real deep reinforcement learning: Continuous control of mobile robots for mapless navigation,” in 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2017, pp. 31–36.

[3] Y. F. Chen, M. Liu, M. Everett, and J. P. How, “Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning,” in 2017 IEEE international conference on robotics and automation (ICRA). IEEE, 2017, pp. 285–292.

[4] C. Chen, Y. Liu, S. Kreiss, and A. Aalai, “Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning,” in 2019 International Conference on Robotics and Automation (ICRA). IEEE, 2019, pp. 6015–6022.

[5] T. Fan, X. Cheng, J. Pan, P. Long, W. Liu, R. Yang, and D. Manocha, “Getting robots unfrozen and unlost in dense pedestrian crowds,” IEEE Robotics and Automation Letters, vol. 4, no. 2, pp. 1178–1185, 2019.

[6] E. A. Sisbot, L. F. Marin-Urias, R. Alami, and T. Simeon, “A human aware mobile robot motion planner,” IEEE Transactions on Robotics, vol. 23, no. 5, pp. 874–883, 2007.

[7] T. Kruse, A. K. Pandey, R. Alami, and A. Kirsch, “Human-aware robot navigation: A survey,” Robotics and Autonomous Systems, vol. 61, no. 12, pp. 1726–1743, 2013.

[8] G.-H. Liu, A. Siravuru, S. Prabhakar, M. Veloso, and G. Kantor, “Learning end-to-end multimodal sensor policies for autonomous navigation,” arXiv preprint arXiv:1705.10422, 2017.

[9] Y. F. Chen, M. Everett, M. Liu, and J. P. How, “Socially aware motion planning with deep reinforcement learning,” in 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2017, pp. 1343–1350.

[10] M. Pfeiffer, U. Schwesinger, H. Sommer, E. Galceran, and R. Siegwart, “A human centered approach for social navigation: A survey,” Journal of Field Robotics, vol. 35, no. 1, pp. 1–49, 2018.

[11] O. Zhelov, J. Zhang, L. Tai, M. Liu, and W. Burgard, “Curiosity-driven exploration for mapless navigation with deep reinforcement learning,” arXiv preprint arXiv:1804.00456, 2018.

[12] P. Long, T. Fan, X. Liao, W. Liu, H. Zhang, and J. Pan, „Towards optimally decentralized multi-robot collision avoidance via deep reinforcement learning,” in 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018, pp. 6252–6259.

[13] T. Fan, X. Cheng, J. Pan, D. Monacha, and R. Yang, “Crowdmove: Autonomous mapless navigation in crowded scenarios,” arXiv preprint arXiv:1807.07870, 2018.

[14] M. J. Proulx, O. S. Todorov, A. Taylor Aiken, and A. A. de Sousa, “Where am I? who am i? the relation between spatial cognition, social cognition and individual differences in the built environment,” Frontiers in psychology, vol. 7, p. 64, 2016.

[15] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, “Continuous control with deep reinforcement learning,” arXiv preprint arXiv:1509.02971, 2015.

[16] J. J. Guerrero, R. Martinez-Cantín, and C. Sagüés, “Visual mapless navigation based on homographies,” Journal of Robotic Systems, vol. 22, no. 10, pp. 569–581, 2005.

[17] Y. Zhu, R. Mottaghi, E. Kolve, J. J. Lim, A. Gupta, L. Fei-Fei, and A. Farhadi, “Target-driven visual navigation in indoor scenes using deep reinforcement learning,” in 2017 IEEE international conference on robotics and automation (ICRA). IEEE, 2017, pp. 3357–3364.

[18] W. Gao, D. Hsu, W. S. Lee, S. Shen, and K. Subramanian, “Intentionnet: Integrating planning and deep learning for goal-directed autonomous navigation,” arXiv preprint arXiv:1710.05627, 2017.

[19] K. Chen, J. P. de Vicente, G. Sepulveda, F. Xia, A. Soto, M. Vázquez, and S. Savarese, “A behavioral approach to visual navigation with graph localization networks,” arXiv preprint arXiv:1903.00445, 2019.

[20] F. Augugliaro, A. P. Schoellig, and R. D’Andrea, “Generation of collision-free trajectories for a quadrocopter fleet: A sequential convex programming approach,” in 2012 IEEE/RSJ international conference on Intelligent Robots and Systems. IEEE, 2012, pp. 1917–1922.

[21] J. Van den Berg, M. Lin, and D. Manocha, “Reciprocal velocity obstacles for real-time multi-agent navigation,” in 2008 IEEE International Conference on Robotics and Automation. IEEE, 2008, pp. 1928–1935.

[22] S. Curtis, A. Best, and D. Manocha, “Menge: A modular framework for simulating crowd movement,” Collective Dynamics, vol. 1, pp. 1–40, 2016.

[23] M. Everett, Y. F. Chen, and J. P. How, “Motion planning among dynamic, decision-making agents with deep reinforcement learning,” in 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2018, pp. 3052–3059.

[24] L. Tai, J. Zhang, M. Liu, and W. Burgard, “Socially compliant navigation through raw depth inputs with generative adversarial imitation learning,” in 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018, pp. 1111–1117.

[25] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” arXiv preprint arXiv:1707.06347, 2017.

[26] C. Mett, L. E. Navarro-Serment, R. MacLachlan, P. Rybski, A. Steinfeld, A. Suppe, C. Urmson, N. Vandapel, M. Hebert, C. Thorpe, et al., “Moving object detection with laser scanners,” Journal of Field Robotics, vol. 30, no. 1, pp. 17–43, 2013.

[27] Z. Wang, “Laser-based detection and tracking of dynamic objects,” Ph.D. dissertation, University of Oxford, 2014.

[28] Wikipedia, “Generalised logistic function,” Wikipedia, 2019.

[29] J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha, “Reciprocal n-body collision avoidance,” in Robotics research. Springer, 2011, pp. 3–19.