

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2020.DOI

Autonomous Social Distancing in Urban Environments using a Quadraped Robot

ZHIMING CHEN^{1*}, TINGXIANG FAN^{2*}, XUAN ZHAO³, JING LIANG⁴, CONG SHEN⁵,
HUA CHEN¹, DINESH MANOCHA⁴, JIA PAN², AND WEI ZHANG¹

¹Department of Mechanical & Energy Engineering, Southern University of Science and Technology, Shenzhen, Guangdong, China

²Department of Computer Science, The University of Hong Kong, Hong Kong, China

³Department of Biomedical Engineering, City University of Hong Kong, Hong Kong, China

⁴Department of Computer Science, University of Maryland, College Park, USA

⁵School of Mechanical Science & Engineering, Huazhong University of Science and Technology, Wuhan, Hubei, China

Corresponding author: Wei Zhang (e-mail: zhangw3@sustech.edu.cn), Jia Pan (e-mail: jpan@cs.hku.hk)

* The first two authors contributed equally to this work.

This work was supported in part by National Natural Science Foundation of China (Grant No. 62073159, and Grant No. 62003155), Shenzhen Science and Technology Program (Grant No. JCYJ20200109141601708), and in part by Innovation and Technology Fund (ITF) ITS/457/17FP, and General Research Fund (GRF) 11207818, 11202119, NSFC and RGC joint grant N_HKU103/16.

ABSTRACT Corona Virus Disease 2019 (COVID-19) pandemic has become a global challenge faced by people all over the world. Social distancing has been proved to be an effective practice to reduce the spread of COVID-19. Against this backdrop, we propose that the surveillance robots can not only monitor but also promote social distancing. Robots can be flexibly deployed and they can take precautionary actions to remind people of practicing social distancing. In this paper, we introduce a *fully autonomous* surveillance robot based on a quadraped platform that can promote social distancing in complex urban environments. Specifically, to achieve autonomy, we mount multiple cameras and a three dimensional light detection and ranging sensor (3D LiDAR) on the legged robot. The robot then uses an onboard real-time social distancing detection system to track nearby pedestrian groups. Next, the robot uses a crowd-aware navigation algorithm to move freely in highly dynamic scenarios. The robot finally uses a crowd-aware routing algorithm to effectively promote social distancing by using human-friendly verbal cues to send suggestions to over-crowded pedestrians. We demonstrate and validate that our robot can be operated autonomously by conducting several experiments in various urban scenarios.

INDEX TERMS Surveillance, Robotics and automation, Human robot interaction.

I. INTRODUCTION

COVID-19 pandemic has quickly become the most dramatic and disruptive event experienced by people all over the world in the year of 2020. People may need to live with the virus for a long time. Practically, one of the most effective measures to minimize the spread of the coronavirus is to promote social distancing. To achieve this goal, several related schemes have been developed that uses existing on-site closed-circuit television (CCTV) systems to detect social distancing. However, the on-site monitoring systems are not ubiquitous in some areas and sometimes may not be able to cover all public corners. Furthermore, although this sort of monitoring systems are capable of detecting social distancing violations, it fails to take any proactive actions to promote social distancing.

Compared to the on-site monitoring systems, the surveillance robots can be flexibly deployed and patrol in the desired public areas. Moreover, the robot can take precautions to promote social distancing rather than simply monitoring them. These potential benefits have been validated by tele-operated robots [1] and hybrid CCTV-robot systems [2]. The hybrid CCTV-robot system introduces external devices such as CCTV to help robots monitoring social distancing. However, there are still several challenges that prevent direct developing a fully autonomous surveillance robot in complex urban environments without any external device. First, to monitor social distances between pedestrians without any external device, an on-board robot-centric real-time perception system is necessary, introducing additional computational complexities to the computationally limited on-board

systems. Second, in many urban scenarios, the robot needs to safely navigate through unstructured and highly dynamic environments. Third, more intelligent human-robot interaction schemes need to be designed to improve the efficiency of promoting social distancing.

In this paper, we introduce a fully autonomous surveillance robot to promote social distancing in complex urban environments. To achieve this autonomy, we first build the surveillance system with multiple cameras and a 3D LiDAR on a quadruped robot, which empowers the robot with omniperceptibility and expands its traversability in complex urban terrains with uneven terrains and stairs that are challenging for normal wheeled mobile robot. Then, we develop an on-board real-time social distancing detection system with the ability to track the robot's nearby pedestrian groups. Next, the CrowdMove [3] algorithm is used to navigate the robot in highly dynamic environments. Finally, we develop a crowd-aware routing algorithm to allow the robots to approach overly-crowded pedestrian groups and to effectively promote social distancing using verbal cues. We also investigate the influence of human voices on the effectiveness and acceptability of quadruped surveillance and social distancing, because it has been reported that a robotic patrolling inspector can be terrifying for general citizen^{**}. We demonstrate that this surveillance robot can be automatically operated with satisfactory human response by conducting experiments in various urban scenarios.

The rest of this paper is organized as follows. Section II reviews the related works. Section III describes the hardware platform that the surveillance system builds upon. Section IV presents the robot's tracking algorithm used for social distancing detection. Section V illustrates the robot's navigation in urban scenarios. Section VI discusses the robot's interactions with humans through verbal communication. Section VII presents the experiments conducted to validate the proposed algorithms. Section VIII concludes this paper.

II. RELATED WORK

In this section, we will give a brief overview of algorithms related to our system, including the perception, navigation, and interaction for surveillance robots.

A. PERCEPTION FOR SURVEILLANCE SYSTEMS

Pedestrian tracking has been widely applied in surveillance video analysis and is well developed based on research on multi-object tracking problems [4]–[7].

Discrete velocities are used to model pedestrians' motion [8], [9]. Although discretization improves the efficiency of prediction, this approach cannot fully satisfy real-life continuous situations. Chung et al. developed cognitive models to improve the performance of their model [10], while the facing direction information was omitted due to the circular modeling of the pedestrians.

Helbing et al. proposed the method of using social force to model and predict people's move according to energy potential which is caused by people and obstacles [11]. Then the tracking performance is improved by detecting abnormal events among pedestrians [12]. Pellegrini et al. developed Linear Trajectory Avoidance (LTA) to improve the accuracy of motion prediction [13]. [14]–[16] developed social interactions among pedestrians to improve the accuracy of behavior models. Sheng et al. proposed the Robust Local Effective Matching Model (RLEMM) to address the issue of partial detection of objects [17]. However, these approaches cannot describe pedestrians' dynamics in dense situations because they only use linear models. In our system, a nonlinear model, Frontal Reciprocal Velocity Obstacles (F-RVO) [4] is used to simulate motions in crowds and also to model the dynamic behaviors considering pedestrians' facing directions.

With the blossom of deep learning, convolutional neural network (CNN) is well developed to extract the trajectory of a single object [18]–[20]. Chu et al. developed spatial temporal attention mechanism (STAM) to detect more objects [21]. Fang et al. improved the performance of tracking by using recurrent neural network (RNN) [6]. The authors in [7] developed the simple online and realtime tracking (SORT) model to track pedestrians. However, by tightly coupling detection and tracking, these approaches cannot always provide satisfactory performance in pedestrian detection. Mask Region Convolution Neural Network (Mask R-CNN) [22] and You Only Look Once Algorithm (YOLO) [23] are two state-of-the-art detection networks with sufficient performance for detecting purposes, where YOLO is much faster than Mask R-CNN, and thus is more suitable for real-time tracking tasks.

B. NAVIGATION IN URBAN ENVIRONMENTS

Compared to the fixed video surveillance system, the surveillance robot not only has the above perception capabilities but also endows the surveillance camera with mobility. However, navigating a robot in urban environments is non-trivial.

First, the robot would inevitably interacts with dynamic obstacles like pedestrians and bicycles. Some methods have been proposed to deal with the collision avoidance problems in such dynamic scenarios. [24], [25] proposed that each agent in dynamics scenarios should take half of the responsibility of collision avoidance. Based on that, they developed the multi-agent collision avoidance algorithm with zero-communication. [26], [27] presented the interacting Gaussian processes to capture the cooperative collision avoidance behavior, and introduced the cooperative planner for robot navigation. However, these algorithms fail to track a moving pedestrian without the assistance of external devices. [28], [29] deployed a light detection and ranging sensor (LiDAR) with multiple cameras on robot to track surrounding pedestrians. To navigate the robot in the crowds, they utilized the reinforcement learning algorithm to train the socially aware collision avoidance policy. Different from the above algo-

^{**}<https://www.fastcompany.com/90539438/this-covid-swabbing-robot-is-terrifying-but-it-doesnt-need-to-be>

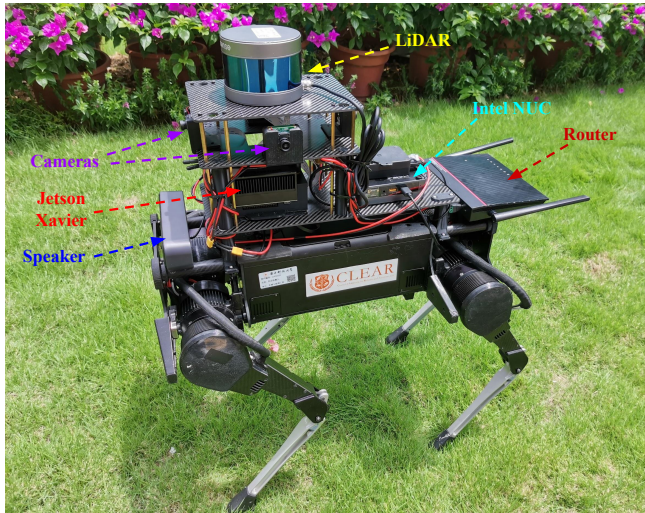


FIGURE 1: Overview of our hardware and software systems.

gorithms, [30]–[32] proposed a sensor-level collision avoidance policy learned via reinforcement learning, which can directly process the raw LiDAR data to generate collision-free actions.

C. ROBOT INTERACTION

Human-like characteristics of social robots would influence users' responses. Among various social traits, gender is important for interpersonal relationships and evokes social stereotypes [34]. Previous research has pointed out that the participants were more accepting of the robots if their perceived gender of a robot conformed to their occupation's gender role stereotypes (e.g., male security robots or female healthcare robots). However, perceived trust of the social robots was not influenced by gender-occupational role conformity [35]. In contrast, Kuchenbrandt et al. [36] found that participants, regardless of gender, evaluated the male and female robots as equally competent while performing a stereotypically female task but, in the context of a stereotypically male task, the female robot was rated as more competent compared to the male robot. Another study examining the effects of robot gender on human behavior found that participants were more likely to rate the robot of the opposite gender as more credible, trustworthy, and engaging [37]. Thus, the effects of users and robot attributes, as well as

gender-role stereotypes, are still open questions.

III. HARDWARE PLATFORM

First, we will introduce the hardware setup of our surveillance robot, which includes three components as shown in Figure 1: the mobile platform, the perception sensor-kit, and the computational platform.

- **Mobile Platform:** We deploy the Unitree Laikago (a quadruped robot) as our mobile platform for navigating in complex urban environments. Comparing to wheeled robots, quadruped robots have superiority in traversability and thus are more suitable for uneven and unstructured urban scenarios with stairs and bumps.
- **Perception Sensor-Kit:** To effectively detect and track pedestrians, we mount four color cameras (RGB cameras) evenly in the horizontal plane of the robot. Each camera is equipped with a short focal lens with the horizontal field of view (FOV) of 80° . Thus, a combination of four cameras can almost cover all directions around the robot. Moreover, for better spatial perception, we use a RoboSense 3D LiDAR with 16 channels to measure the social distance between pedestrians. The 3D LiDAR also serves the navigation applications in mapping, localization, and collision avoidance.
- **Computational Platform:** Two on-board computers are mounted to process the aforementioned sensor data for different tasks. We use NVIDIA Jetson AGX Xavier as the vision computational module that supports a maximum of six lanes CSI cameras as the input and uses 512 CUDA cores to GPU-accelerate the processing of images captured by the cameras. Since other tasks like mapping and localization would mostly consume CPU resources, we also deploy an Intel NUC with Intel i5 8259U CPU. These two computers are connected by wired network, and the processed data is shared by Robotic Operating System (ROS).

IV. SOCIAL DISTANCING DETECTION

The tracking algorithm used in our system consists of object detection, bounding box prediction, feature extraction, and sparse feature matching. We use YOLO to detect pedestrians, and update the traces of pedestrians via matching sparse features with the help of motion modeling algorithm F-RVO.

Remark 1: Due to the privacy concern, our system will not collect any private information but only give a verbal reminder. Moreover, facial recognition technology is not embedded in the application, which means the pedestrians' ID information is unknown to the robot.

A. F-RVO

Modeling pedestrians' behaviors in crowds from the front view is challenging, not only because of the non-linearly varied motions (turning shoulder, side walking, back stepping, etc. [38]), but also due to the occlusions that front view may

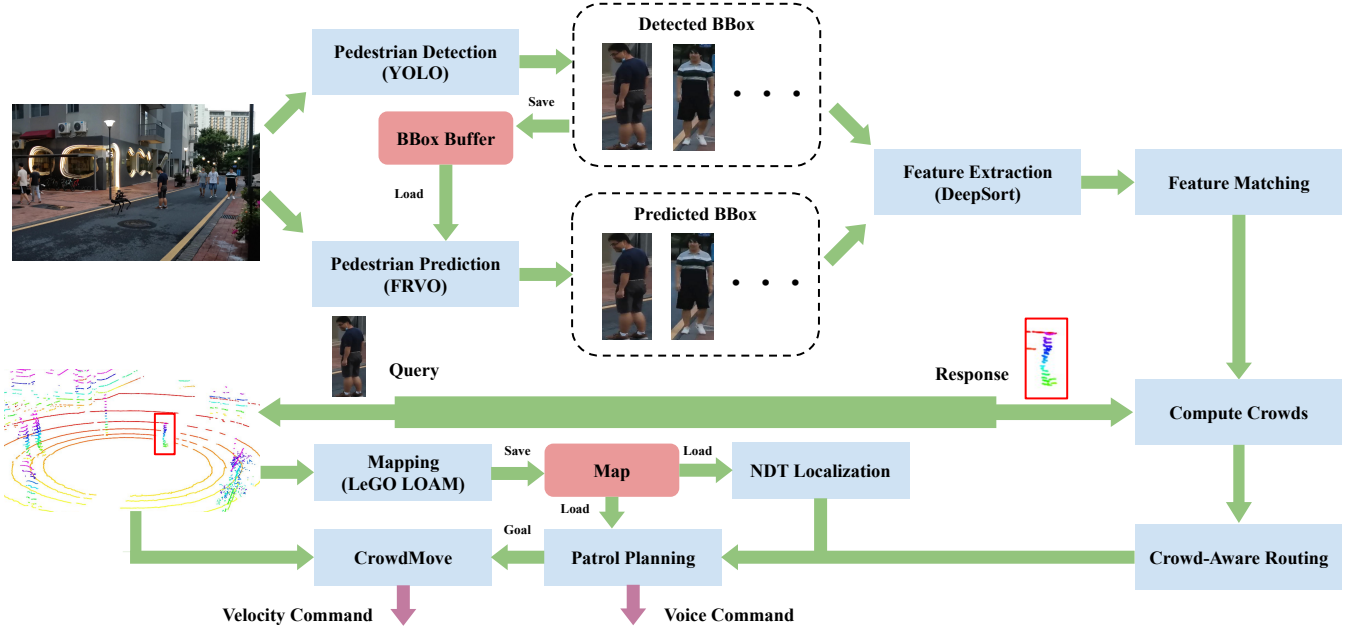


FIGURE 2: Our system contains functional modules of tracking, mapping, localization, patrol planning, routing, and motion planning. Tracking module uses YOLO [23] and F-RVO [4] to extract similar detected objects of consecutive frames and to keep track of people. Mapping is achieved by using the Lightweight and Ground-Optimized Lidar Odometry and Mapping (LeGO-LOAM) algorithm [33] which is based on 3D Lidar sensor. For localization, we used the Normal Distributions Transform (NDT) localization algorithm to match lidar data and localize robot in the generated map. According to the detected crowds and map, crowd-aware routing algorithm and patrol planning algorithm would help robot to determine current target to approach. With all information needed for motion planning, an end-to-end algorithm, CrowdMove, is used to drive robot toward the goal position. During the approaching, if the robot detects its distance to the crowds is lower than 5 meters, it starts to play a recorded vocal command to remind people to keep a proper social distance.

encounter. In this work, we use a velocity-obstacle based algorithm, F-RVO [4], to model the pedestrians motion.

In F-RVO, each pedestrian, p_i , is represented by an 8-dimensional vector: $\Psi_t = [x, v, v_{pref}, l, w]$, where x is the current 2D position, v is the 2D velocity, v_{pref} is the preferred 2D velocity that we assume people would prefer to walk along the front direction. l and w are the length and width of human's shoulder, respectively. For each frame τ , a half-plane constraint is used to determine the range parameter in F-RVO. Within the range, each pedestrian p_i has an area of velocity obstacle $VO_{p_i|p_j}^\tau$ with respect to another neighboring pedestrian p_j . The convex region of velocity obstacles considering all neighbors can then be computed as:

$$FRVO_{p_i}^\tau = \bigcup_{p_j \in H_i} VO_{p_i|p_j}^\tau, \quad (1)$$

where H_i is the set of all neighbors of pedestrian p_i . Out of the velocity obstacle area, the best velocity is chosen with the nearest distance to preferred velocity v_{pref} :

$$v_{best} = \arg \min_v \|v - v_{pref}\|, \quad (2)$$

where $v \notin FRVO_{p_i}^\tau$.

B. DENSEPEDS

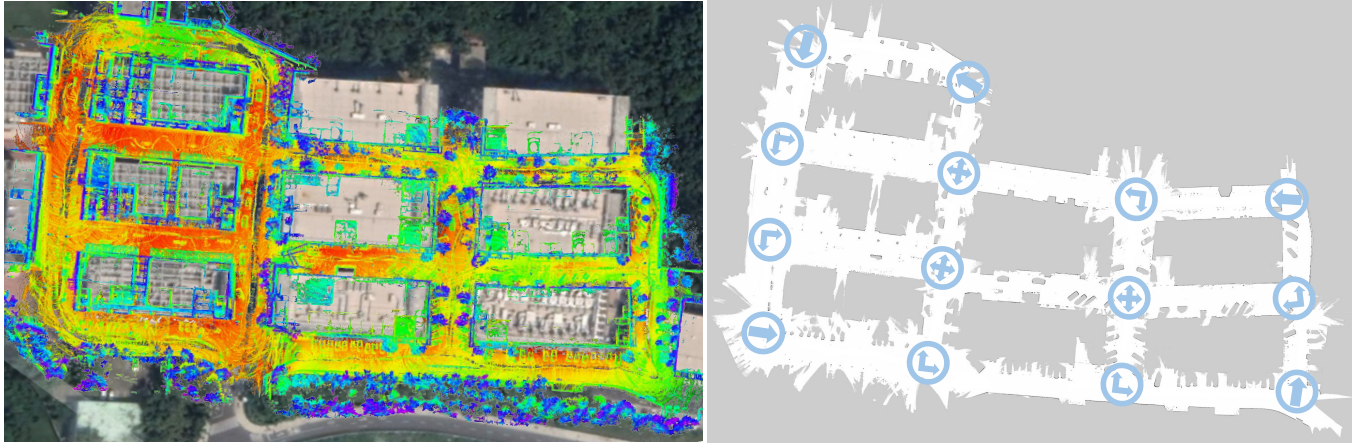
The tracking algorithm, DensePeds, involves three components to track pedestrians: object detection, feature extraction, and feature matching, as shown in upside of Figure 2. At

each time step, we first use YOLO to detect pedestrians and generate the associated bounding boxes. Denote by P the set of all detected pedestrians, we use F-RVO to predict another set of bounding boxes around each pedestrian $p_i \in P$. Given the bounding boxes computed between two adjacent time steps, we use DeepSort CNN [7] to extract binary feature vectors from the sub-images as determined by the bounding boxes. Then we perform matching over these sparse features to find in frame $t + 1$ the best matched pedestrians of frame t and assigned IDs to the pedestrians accordingly. In particular, the sparse features are matched in two steps. First, we find the most similar detected pedestrian of a predicted pedestrian using the cosine metric (cosine value of angle between two vectors), i.e.,

$$h_j^* = \arg \min_{h_j} \{d(f_{p_i}, f_{h_j}) \mid p_i \in P, h_j \in H_i\} \quad (3)$$

where $d(\cdot, \cdot)$ is the cosine metric, $f(\cdot)$ is the feature extraction function, $p_i \in P$ is one of the pedestrians in a frame, h_j is one detected pedestrian in the set H_i , which is the set of detected neighbors around the pedestrian p_i . In the second step, we maximize the IoU (Intersection over Union) overlap, i.e., the overlapped area between predicted boxes and original YOLO-detected boxes,

$$\epsilon(i, j) = \frac{B_{p_i} \cap B_{h_j}}{B_{p_i} \cup B_{h_j}}, \quad (4)$$



(a) Mapping: LeGO-LOAM

(b) Potential directions in each cross and corner

FIGURE 3: (a) We used LeGO-LOAM for mapping. The blue arrows in (b) shows the potential directions in each cross and corner that robot can go. Based on the information of the map and current position, the patrol algorithm chooses for the robot a navigation direction with the maximum probability of crowd appearance.

where B_{p_i} and B_{h_j} are the bounding boxes around p_i and h_j respectively. Matching a set of detected pedestrians to a set of predicted pedestrians with maximum overlap eventually becomes a max weight matching problem over the matrix $\epsilon(i, j)$, which can be accelerated using the Hungarian algorithm [39].

According to the computed bounding boxes, we can roughly estimate the range and bearing information between the robot and its surrounding pedestrians. To estimate the crowds more accurately, we re-project the bounding boxes to the LiDAR coordinate to query the depth of each pedestrian. The random sample consensus (RANSAC) algorithm [40] is applied to filter out the possible outlier points. If there is a large inconsistency between LiDAR and visual estimates due to the occupation by pedestrians, the visual estimates would be trusted. Finally, we obtain the social distance between pedestrians.

V. NAVIGATION IN URBAN

In this section, we will introduce the autonomous navigation algorithm for urban scenarios. We implement the mapping and localization function by the state-of-the-art LiDAR-based approaches. The navigation framework adopts a hierarchical structure. In particular, we develop a learning-based collision avoidance algorithm for local planning, and use a global planner to plan trajectories for the robot to patrol. In addition, we will describe the routing algorithm enabling a robot to effectively select a crowded region to approach in order to accomplish the patrol.

A. MAPPING AND LOCALIZATION

Since LiDAR-based SLAM approaches have been well developed in recent years, we are not going to develop a new SLAM approach in this paper. To achieve the best performance given the limited computational capability, we choose the Lightweight and Ground-Optimized Lidar Odom-

etry and Mapping (LeGO-LOAM) algorithm, which is a light-weighted system and is optimized for the ground platforms [33]. The generated map is shown in Figure 3a.

Once the 3D point cloud map about the environment is obtained, the Normal Distributions Transform (NDT) scan matching algorithm is used for localization [41], which have been demonstrated in [42] to be able to provide more reliable result than other matching methods such as Iterative Closest Points [43].

Although we can compute the 3D point cloud map and the robot's localization, it is not easy for the robot to determine the traversable region in the 2D plane. Therefore, we transform the 3D point cloud to the 2D laser scan, by taking the closest point within the certain height as a 2D laser point. Note that, during the navigation the robot may encounter uneven terrains like stairs or steps. Thus, the transform ignores the point cloud on the ground plane by filtering out the cloud points lower than 30 cm. After the transform, we obtain a 2D occupancy map for the following navigation algorithm as shown in Figure 3b.

B. PATROL AND ROUTING

Based on the generated map and current position, we propose a patrol planning algorithm to navigate the robot around the mapped area. As shown in Figure 3, in different crosses and corners, the robot would choose different navigation directions optimized for social distancing. In particular, the robot would prefer the direction where there is a high probability that a crowd would appear.

When the robot detects gathered crowds, it would suspend the patrol algorithm and switch to the routing algorithm to find an optimal way to approach the crowds. Considering the time constraints and sizes of crowds, we propose a crowd-aware routing algorithm based on the depth-first search method to find a sequence of intermediate waypoints for the robot to follow.

TABLE 1: Dependent measures in the user questionnaire

Factors	# Items	Cronbach Alpha	Scale	Items
Attitude toward robots	2	0.836	Likert scale	It's a good idea to use the surveillance robot. It's good to make use of the robot
Perceived trust	3	0.835	Likert scale	I would trust the surveillance robot, if he/she gave me advice. I trust that the surveillance robot can keep me away from health risks. I would follow the advice that the surveillance robot gives me.
Acceptance	3	0.868	Likert scale	If given a chance, I'll use this robot in a college campus in the near future. If given a chance, I'll use this robot in a park in the near future. If given a chance, I'll use this robot in a shopping mall in the near future.
Perceived ability	3	0.864	Likert scale	The robot is very capable of performing its job. I feel very confident about the robot's skills. The robot has much knowledge about the work that needs to do.

TABLE 2: Means and standard deviations of all the measures

	Male voice			Female voice			Neutral voice			Child voice		
	overall (n=54)	male (n=23)	female (n=31)	overall (n=53)	male (n=31)	female (n=22)	overall (n=56)	male (n=31)	female (n=25)	overall (n=55)	male (n=34)	female (n=21)
Perceived ability	3.90 (0.78)	3.62 (0.96)	4.10 (0.56)	4.05 (0.81)	3.98 (0.82)	4.15 (0.80)	4.06 (0.77)	3.91 (0.79)	4.24 (0.72)	4.15 (0.66)	4.12 (0.69)	4.19 (0.64)
Acceptance	3.67 (0.98)	3.59 (1.23)	3.72 (0.76)	4.06 (0.80)	4.05 (0.83)	4.08 (0.78)	4.08 (0.73)	3.89 (0.74)	4.32 (0.65)	4.07 (0.72)	4.02 (0.73)	4.14 (0.70)
Perceived trust	3.91 (0.60)	3.86 (0.78)	3.95 (0.43)	4.09 (0.73)	4.02 (0.72)	4.18 (0.73)	4.20 (0.67)	4.08 (0.69)	4.36 (0.62)	4.13 (0.72)	4.10 (0.77)	4.19 (0.66)
Attitude toward robots	3.96 (0.98)	3.78 (1.10)	4.10 (0.88)	4.25 (0.79)	4.21 (0.74)	4.32 (0.87)	4.31 (0.66)	4.16 (0.68)	4.50 (0.61)	4.15 (0.73)	4.16 (0.71)	4.14 (0.76)

TABLE 3: F value and significance of robot voice effect and user gender effect

	Robot voice effect						User gender effect	
	overall F	p	male F	p	female F	p	F	p
Perceived ability	1.027	0.382	1.761	0.158	0.221	0.882	5.160	0.024**
Acceptance	3.362	0.020**	1.455	0.231	3.404	0.021**	1.285	0.258
Perceived trust	1.866	0.136	0.563	0.640	2.214	0.092*	1.939	0.165
Attitude toward robots	2.017	0.113	1.534	0.209	1.397	0.248	2.071	0.152

* $p < 0.1$, ** $p < 0.05$

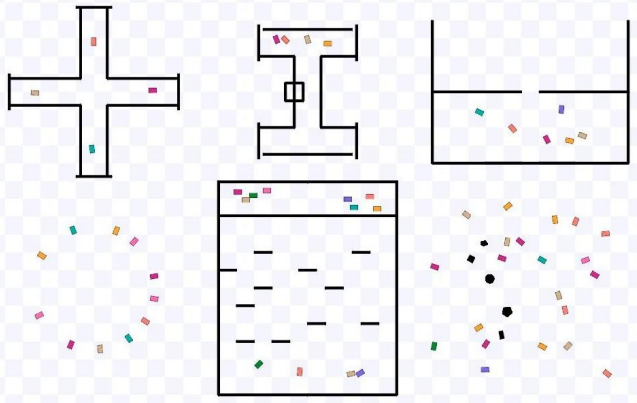


FIGURE 4: Multi-robot multi-scenario training environments in the Stage simulator

We formulate the routing problem as follows. Assume that there are N groups of people within the robot's perception range. Each crowd is denoted as a node n_i , with its specific time-window constraint t_i , and its relative location to the robot. Each crowd is assigned a weight w_i according to the number of persons in the group. The routing algorithm aims at finding an optimal path for the robot to approach as many crowds as possible with the least energy consumption. The

optimization objective is:

$$cost = \sum_{p_j \in p_c} e_j - \sum_{i \in N} w_i - n_c, \quad (5)$$

where p_c is the current trajectory which contains a set of points and edges denoting the positions of crowds and the paths connecting them. Each edge $p_j \in p_c$ between two positions has the energy cost e_j . The number of crowds explored in p_c is denoted as n_c .

Given the directions and positions of the crowds after routing algorithm, we implement the SBPL lattice planner [44] to generate a smooth patrol route passing through these waypoints.

C. LEARNING-BASED COLLISION AVOIDANCE

During patrol, the robot will not only encounter the static obstacles, but also interact with moving pedestrians. For this case, we deploy the learning-based collision avoidance approach, CrowdMove [3], for robotic navigation in crowds.

The main training framework refers to our previous work [30], which takes a 2D laser scan as the input and outputs the velocity command. The multiple training scenarios are designed with multiple robots in the Stage simulator as shown in Figure 4. We introduce the *centralized learning, decentralized execution* training paradigm, which

shares the same navigation policy during the training. Then, we obtain a multi-robot collision avoidance policy with zero communication. Furthermore, we validate that the trained policy can be transferred from the simulation to the real world without any re-tuning, and it is also suitable for the single robot navigation in crowds [31], [32]. To adapt the training framework to our quadrupedal hardware platform, we take the transformed laser scan which represents the local traversable area as the input.

VI. VOICE INTERACTION

In our surveillance scenario, we use verbal cues to send suggestions from robot to human. As we mentioned before, the user's gender and the robot's gender may influence the user's acceptance and trust in the robot. Thus, to reach an effective surveillance result, we gave our robot four types of gendered voice and designed a user study to select the best one. In this section, we introduce the study for investigating (1) the user's gender-based effects of the autonomous robot and, (2) user's attitude, acceptance, trust, and perceived trust through robot with different voices.

A. METHOD

1) Gender of the robot

We manipulated the gender of the robot through non-verbal cues by changing the vocal characteristics. Because we aim to find the robot voice with best performance, the voice selection is not strictly limited to robot gender effects. In this experiment, we prepared four types of voices: three gendered voices and a child voice. The gendered voices include a computer-generated neutral voice, a male and a female recorded by real adult human, a child voice by a girl.

2) Procedures

Among the various issues in human-robot interaction, trust was nominated as one of the primary factors to be considered. In this particular task, trust is performed as how much a human would follow the advice sent by the surveillance robot. This factor would crucially influence the performance of the robot. To better measure the users' experience of the robot, we suggest four dependent measures which include the users' attitude towards the robot, perceived trust, and acceptance of the robot. The details of the measures are shown in Table 1.

As part of a larger study investigating the users' perceptions in an autonomous surveillance robot, the participants filled out a survey measuring the factors shown in Tabel 1. Each measure was assessed on a 5-point Likert scale ('1' = strongly disagree, '5' = strongly agree).

This experiment was done in between-subject mode to minimize the learning and transfer across conditions. Each participant in this study viewed two videos and then responded to survey items related to the videos. Both of the videos demonstrate the same scenario with the same robot voice. One video was from a third-person perspective, where the robot is walking towards the crowds while asking them



FIGURE 5: The screenshots of two videos in the questionnaire. Top: third-person perspective; bottom: first-person perspective.

to keep the social distance. The other video was recorded from a first-person perspective where the robot is walking towards the camera while asking the human to keep the social distance. For both scenarios, the robot starts to play the voice at about 5 meters away from the crowd. The screenshots of the two videos are shown in figure 5. In this way, a total of 8 videos were recorded, which are 2 perspectives times 4 types of robot voices. For each scenario, we add a description "The robot shown in the videos is a surveillance robot working on keeping a low density of humans during COVID-19. When the robot finds a crowd, he/she/it will walk toward the crowd while asking them to keep a proper social distance. Please watch these two videos, and imagine you were one of the humans in the video, then answer the following questions."

3) Participants

A total of 218 adults (119 males; 99 females) between 20-55 years old ($M=29.49$, $SD=12.02$) participated in the between-subject experiment. Participants were mostly students and staff from the Southern University of Science and Technology. The participants were recruited through the posters and links shared in a social media app. Each participant needs to read and sign a consent form before they start the questionnaire.

B. DATA ANALYSES

A manipulation check was performed to ensure that the robots could manifest gender and age successfully. The perceived gender was measured through a sliding bar with 0 the most femininity and 100 the most masculinity. The perceived age was measured through a sliding bar between 5 to 70. The one-way ANOVA showed that participants perceived male voice more masculine ($M = 76.14$), female voice more feminine ($M = 49.25$) and a neutral voice in the middle ($M = 64.28$). The F and p value is $F = 9.902$ and $p < 0.0001$. The participants also significantly perceived robot with child's voice ($M = 20.93$) younger than others ($M = 28.02$, $p = 0.008$).

We calculated Cronbach's alpha values to assess the internal consistency of each psychometric measure. The reported alpha values were between 0.8-0.9, indicating that the items have relatively high internal consistency. To calculate the significance of user gender and robot voice type effect, a one-way ANOVA was conducted. The robot voice and user gender were treated as independent variables. For factors reached significant differences according to conditions, we

used the least significant difference (LSD) to make a pairwise comparison.

VII. EXPERIMENTS

In this section, we first validate the effectiveness of the proposed approach individually. Then, we integrate all the modules to realize the autonomous surveillance robot. To further investigate the performance of surveillance robot on promoting social distancing, we conduct some real-world experiments in the end.

A. CROWD GATHERING DETECTION

We first record vision and LiDAR data to better analyze and tune the social distancing detection system. The recorded dataset includes a wide variety of pedestrian group behaviors, such as walking, standing, gathering, and scattering.

Crowd gathering is not easy to be well quantified, especially the occlusion between pedestrians makes the robot difficult or even impossible to accurately acquire the location of each pedestrian. To detect each possible crowd gathering, we establish a graph-based pedestrian network called *social graph*, with one example illustrated in Figure 6a. In the social graph, each node represents the pedestrian's position. The green, yellow and red edges represent the safe, warning, and dangerous social distance, respectively. We consider the distance less than 2 meters to be dangerous, the distance between 2 and 4 meters to be warning, and the distance greater than 4 meters to be completely safe. We connect the nodes between red and yellow edges into a subgraph called the *crowd graph*, which is considered as possible crowd gathering. In this way, we can reduce the dependence of the crowd gathering detection on the accuracy of estimating pedestrian positions.

B. NAVIGATION IN URBAN

The urban navigation would mainly encounter two challenges, the unstructured environments and the dynamic obstacles. Thanks to the superior mobility of the quadruped platform, our robot can navigate over uneven terrains such as steps without extra visual estimation effort as shown in Figure 7 and thus can handle unstructured environments easily.

To validate the dynamic collision avoidance performance among pedestrians, we create a crowded and narrow indoor scenario in the lab, as shown in Figure 8. In this experiment, the robot is required to perform tasks of tracking a specified target (a bone in this work) while avoiding all the dynamical obstacles (pedestrians). We install in the lab several ultra-wide band (UWB) tags accounting for indoor localization. During the experiments each lasting about 30 minutes, the robot dog mounted with a 3D LiDAR can achieve nearly zero collision in this scenario. This experiment indicates that our learning-based collision avoidance policy can be successfully transferred and deployed to the real-world robotic dog.

C. VOICE PREFERENCE

Table 2 shows the means and standard deviations of all measures according to different robot voice types and user genders. The score of each factor was calculated by averaging both/all the related items. It can be seen that the male voice type got the lowest average score on all the measures. Also, female users marked the highest on all the measures for neutral voice type.

Table 3 demonstrates the F and p-value. The result shows that for male users, there is no significant result among all different robot voice types while for female users, the robot voice type significantly influences the user's acceptance ($p = 0.021$) and perceived trust ($p = 0.092$). To find which condition differs for female users, We used the least significant difference (LSD) to make a pairwise comparison between different robot voice types. Surprisingly, For female users, the acceptance, perceived trust, and attitude toward robot in neutral robot voice condition are higher than other robot voice condition, especially for male voice condition ($p = 0.003, 0.013, 0.061$ respectively).

We also compare the effects of users' genders in different conditions. It is found that female user has higher perceived ability than male users ($p = 0.024$), especially in male voice condition ($p = 0.027$). In the neutral voice condition, female's acceptance and attitude toward robots are significantly higher than male's ($p = 0.028$ and $p = 0.057$ respectively).

There is no significant difference for male users markings according to different robot voice types. However, it is quite surprising that the female users marked very high for the neutral robot voice. In addition, surveillance should be a masculine job but both male and female users marked all four factors the lowest in male voice condition. Therefore, we do not suggest the usage of the male robot voice. Considering the means of the four factors among all robot voice conditions, we selected the neutral voice for our surveillance robot.

D. REAL-WORLD EXPERIMENT ON PROMOTING SOCIAL DISTANCING

Finally, we integrate all the above modules together, and investigate whether the robot can navigate in the complex urban environments with satisfactory social distancing performance without terrifying general citizens. The real-world experiment was conducted in two public areas including a university campus and a park. Figure 9 shows some examples from the real world experiment.

The result shows that our robot successfully fulfills the task of social distancing. For people who have been interacted with our robot, about half of them followed the robot suggestions. For the other people, most of them glanced at the robot and then just walked away, some of them stopped and looked at the robot. It's worth to notice that at the time of our experiment, there were no existing COVID-19 patients in the testing city, which tends to reduce the pedestrian's compliance with the verbal social distancing commands. During the experiment, we selected some people randomly,

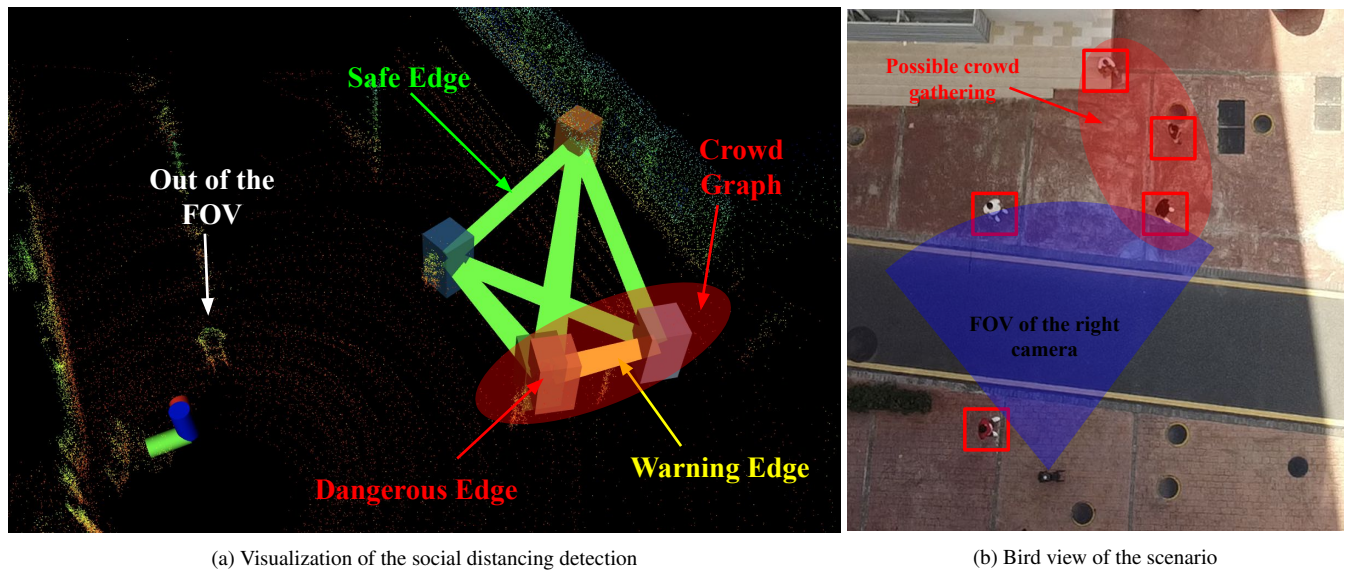


FIGURE 6: Illustration of the crowd gathering detection within the right camera's field of view (FOV). Although the estimated position of pedestrians is not very accurate, we can still detect possible crowd gatherings by establishing the crowd subgraph.

then asked them about their attitude towards the robot and why they followed/didn't follow the robot's advice. Some people reported that they felt it is a great idea to use the surveillance robot and they thought the robot's advice is reasonable. Besides, many people reported the robot looks like it came from the world of science fiction so they were very curious about the robot. However, some people felt the robot is not friendly enough so they just wanted to walk

away. For the people who ignored the robot's advice, most of them said that the pandemic is not severe so they felt it's unnecessary to keep the distance.

VIII. CONCLUSION

In the context of the COVID-19 pandemic, we develop an autonomous surveillance robot system to promote social distancing. The robot system is mainly composed of social distance detection, urban navigation, and intelligent voice interaction. The legged robot shows good adaptation to different terrain so that they can work well in human life scenarios. The real-world experiment also demonstrates that our robot successfully keeps human's social distance. In this end, we successfully deploy the system in a real environment to prevent the spread of COVID-19.

Regarding the usage of the developed platform in response to the COVID-19 pandemics, there is still room for improvement with respect to social relationship analysis as social distancing may be affected by many social factors. For example, the robot should not remind a group of members from the same family who are intentionally violating social distancing. Our future work will focus on identifying the social relationship of pedestrians from their historical behaviors. Furthermore, equipping the developed platform with temperature sensors and sanitation tools would grant the robot the ability to monitor pedestrians' body temperature and help in regular cleaning. Such a highly integrated legged platform would be very efficient in accomplishing various tasks.

Apart from the applications in response to the COVID-19 pandemic, the developed platform has the potential to be applied in a wide range of social interaction activities in human's daily life. For example, the platform can server as a tour guide in resort parks, can replace human operators

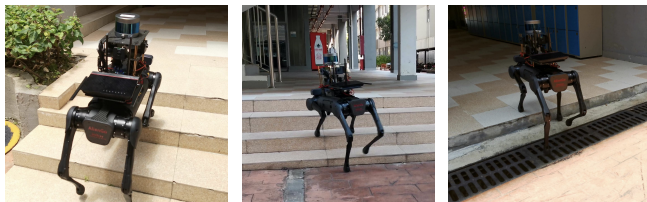


FIGURE 7: The legged robot can traverse in uneven terrains.



FIGURE 8: The demonstration of the dynamic collision avoidance experiments. We arranged 6 moving pedestrians in this scenarios about $4\text{ m} \times 4\text{ m}$ in size.

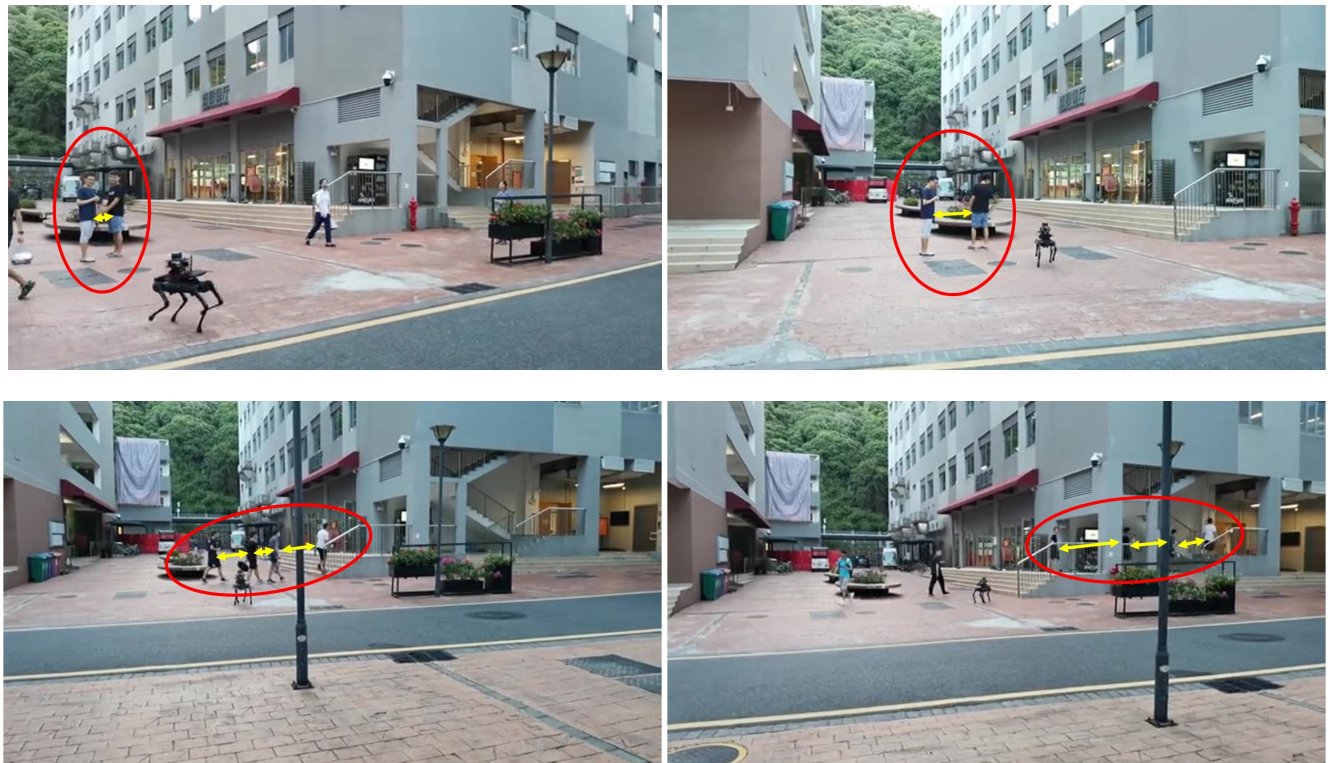


FIGURE 9: Examples from the real-world experiment. The top and bottom images describes two different scenarios. Left: The robot detected and approached the crowds, then persuaded them to keep social distance. Right: The crowds density decreased.

in routine navigation and inspection for factories, and can accompany and serve elders.

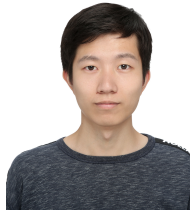
REFERENCES

- [1] J. Nalewicz. (2020) Singapore is using a robotic dog to enforce proper social distancing during covid-19. [Online]. Available: <https://www.smithsonianmag.com/smart-news/singapore-using-robotic-dog-enforce-proper-social-distancing-during-covid-19-180974912/>
- [2] A. J. Sathyamoorthy, U. Patel, Y. A. Savle, M. Paul, and D. Manocha, "Covid-robot: Monitoring social distancing constraints in crowded scenarios," *arXiv preprint arXiv:2008.06585*, 2020.
- [3] T. Fan, X. Cheng, J. Pan, D. Manocha, and R. Yang, "Crowdmov: Autonomous mapless navigation in crowded scenarios," *arXiv preprint arXiv:1807.07870*, 2018.
- [4] R. Chandra, U. Bhattacharya, A. Bera, and D. Manocha, "Densepeds: Pedestrian tracking in dense crowds using front-rvo and sparse features," *arXiv preprint arXiv:1906.10313*, 2019.
- [5] R. Chandra, U. Bhattacharya, C. Roncal, A. Bera, and D. Manocha, "Robusttp: End-to-end trajectory prediction for heterogeneous road-agents in dense traffic with noisy sensor inputs," in *ACM Computer Science in Cars Symposium*, 2019, pp. 1–9.
- [6] K. Fang, Y. Xiang, X. Li, and S. Savarese, "Recurrent autoregressive networks for online multi-object tracking," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2018, pp. 466–475.
- [7] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *2017 IEEE international conference on image processing (ICIP)*. IEEE, 2017, pp. 3645–3649.
- [8] G. Antonini, S. V. Martinez, M. Bierlaire, and J. P. Thiran, "Behavioral priors for detection and tracking of pedestrians in video sequences," *International Journal of Computer Vision*, vol. 69, no. 2, pp. 159–180, 2006.
- [9] T. Robin, G. Antonini, M. Bierlaire, and J. Cruz, "Specification, estimation and validation of a pedestrian walking behavior model," *Transportation Research Part B: Methodological*, vol. 43, no. 1, pp. 36–56, 2009.
- [10] S.-Y. Chung and H.-P. Huang, "A mobile robot that understands pedestrian spatial behaviors," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2010, pp. 5861–5866.
- [11] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Physical review E*, vol. 51, no. 5, p. 4282, 1995.
- [12] R. Mehran, A. Oyama, and M. Shah, "Abnormal crowd behavior detection using social force model," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 935–942.
- [13] S. Pellegrini, A. Ess, K. Schindler, and L. Van Gool, "You'll never walk alone: Modeling social behavior for multi-target tracking," in *2009 IEEE 12th International Conference on Computer Vision*. IEEE, 2009, pp. 261–268.
- [14] S. Pellegrini, A. Ess, and L. Van Gool, "Improving data association by joint modeling of pedestrian trajectories and groupings," in *European conference on computer vision*. Springer, 2010, pp. 452–465.
- [15] W. Choi, K. Shahid, and S. Savarese, "What are they doing?: Collective activity classification using spatio-temporal relationship among people," in *2009 IEEE 12th international conference on computer vision workshops, ICCV Workshops*. IEEE, 2009, pp. 1282–1289.
- [16] K. Yamaguchi, A. C. Berg, L. E. Ortiz, and T. L. Berg, "Who are you with and where are you going?" in *CVPR 2011*. IEEE, 2011, pp. 1345–1352.
- [17] H. Sheng, L. Hao, J. Chen, Y. Zhang, and W. Ke, "Robust local effective matching model for multi-target tracking," in *Pacific Rim Conference on Multimedia*. Springer, 2017, pp. 233–243.
- [18] S. Hong, T. You, S. Kwak, and B. Han, "Online tracking by learning discriminative saliency map with convolutional neural network," in *International conference on machine learning*, 2015, pp. 597–606.
- [19] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Hierarchical convolutional features for visual tracking," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 3074–3082.
- [20] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Stct: Sequentially training convolutional networks for visual tracking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1373–1381.
- [21] Q. Chu, W. Ouyang, H. Li, X. Wang, B. Liu, and N. Yu, "Online multi-object tracking using cnn-based single object tracker with spatial-

- temporal attention mechanism,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4836–4845.
- [22] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask r-cnn,” in *IEEE international conference on computer vision*, 2017, pp. 2961–2969.
- [23] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.
- [24] J. Van den Berg, M. Lin, and D. Manocha, “Reciprocal velocity obstacles for real-time multi-agent navigation,” in *IEEE International Conference on Robotics and Automation*, 2008, pp. 1928–1935.
- [25] 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.
- [26] P. Trautman and A. Krause, “Unfreezing the robot: Navigation in dense, interacting crowds,” in *IROS*, 2010, pp. 797–803.
- [27] P. Trautman, J. Ma, R. M. Murray, and A. Krause, “Robot navigation in dense human crowds: the case for cooperation,” in *ICRA*, 2013.
- [28] Y. F. Chen, M. Everett, M. Liu, and J. P. How, “Socially aware motion planning with deep reinforcement learning,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2017, pp. 1343–1350.
- [29] M. Everett, Y. F. Chen, and J. P. How, “Motion planning among dynamic, decision-making agents with deep reinforcement learning,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2018, pp. 3052–3059.
- [30] 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 *In Proceedings of the IEEE International Conference on Robotics and Automation*, 2018, pp. 6252–6259.
- [31] T. Fan, P. Long, W. Liu, and J. Pan, “Distributed multi-robot collision avoidance via deep reinforcement learning for navigation in complex scenarios,” *The International Journal of Robotics Research*, 2020.
- [32] 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.
- [33] T. Shan and B. Englot, “Lego-loam: Lightweight and ground-optimized lidar odometry and mapping on variable terrain,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 4758–4765.
- [34] N. L. Muscanell and R. E. Guadagno, “Make new friends or keep the old: Gender and personality differences in social networking use,” *Computers in Human Behavior*, vol. 28, no. 1, pp. 107–112, 2012.
- [35] B. Tay, Y. Jung, and T. Park, “When stereotypes meet robots: the double-edged sword of robot gender and personality in human–robot interaction,” *Computers in Human Behavior*, vol. 38, pp. 75–84, 2014.
- [36] D. Kuchenbrandt, M. Häring, J. Eichberg, F. Eyssel, and E. André, “Keep an eye on the task! how gender typicality of tasks influence human–robot interactions,” *International Journal of Social Robotics*, vol. 6, no. 3, pp. 417–427, 2014.
- [37] M. Siegel, C. Breazeal, and M. I. Norton, “Persuasive robotics: The influence of robot gender on human behavior,” in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2009, pp. 2563–2568.
- [38] A. Best, S. Narang, and D. Manocha, “Real-time reciprocal collision avoidance with elliptical agents,” in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 298–305.
- [39] H. W. Kuhn, “The hungarian method for the assignment problem,” *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.
- [40] M. A. Fischler and R. C. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [41] M. Magnusson, A. Lilienthal, and T. Duckett, “Scan registration for autonomous mining vehicles using 3d-ndt,” *Journal of Field Robotics*, vol. 24, no. 10, pp. 803–827, 2007.
- [42] M. Magnusson, A. Nuchter, C. Lorken, A. J. Lilienthal, and J. Hertzberg, “Evaluation of 3d registration reliability and speed—a comparison of icp and ndt,” in *2009 IEEE International Conference on Robotics and Automation*. IEEE, 2009, pp. 3907–3912.
- [43] P. J. Besl and N. D. McKay, “Method for registration of 3-d shapes,” in *Sensor fusion IV: control paradigms and data structures*, vol. 1611. International Society for Optics and Photonics, 1992, pp. 586–606.
- [44] M. Likhachev and D. Ferguson, “Planning long dynamically feasible maneuvers for autonomous vehicles,” *The International Journal of Robotics Research*, vol. 28, no. 8, pp. 933–945, 2009.



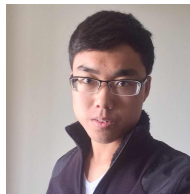
robots, computer vision and machine learning.



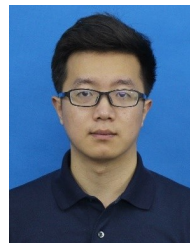
TINGXIANG FAN received the B.S. degree in aerospace engineering from Beijing Institute of Technology, Beijing, China. Now he is a Ph.D. candidate at the University of Hong Kong, Hong Kong. His research interest includes robotic navigation and artificial intelligence.



XUAN ZHAO received the B.S. degree in mechanical engineering from Xi'an Jiaotong University, Xi'an, China, and the M.S. degree in Robotics from the University of Bristol, Bristol, UK. Now she is a Ph.D. student at City University of Hong Kong, Hong Kong. Her research interest includes human-robot collaboration and user-friendly robot motion planning.



JING LIANG received B.S. degree in Autonomous Control from China University of Geosciences, Beijing, China in 2014 and the M.S. degree in Robotics from University of Maryland, College Park, USA, in 2020. He is currently pursuing the Ph.D. degree in Computer Science at University of Maryland, College Park, USA. From 2016-2017, he worked in Clever Sys Inc as system engineer, VA, USA. In 2020, he is working in WaterMirror Technology. His research interest includes Robotics, Planning, Perception and Artificial Intelligence.



CONG SHEN received the B.S. degree in mechanical engineering from Southern University of Science and Technology, Shen Zhen, China. Now he is pursuing the M.S. degree in mechanical engineering at Huazhong University of Science and Technology, Wu Han. His research interest includes modelling and motion control of wheel-bipedal robot.



HUA CHEN received the B.S. in Automation from Zhejiang University in 2012, the Ph.D. in Electrical and Computer Engineering from The Ohio State University in 2018. From January 2019 to June 2019, he was a Postdoctoral Researcher in Department of ECE at The Ohio State University. Then, he joined the Southern University of Science and Technology, Shenzhen, China, where he is currently a Research Assistant Professor with the Department of Mechanical and Energy Engineering. His research interests lie in the intersection of control, optimization and reinforcement learning, with applications in intelligent robotic and autonomous systems.



DINESH MANOCHA is Paul Chrisman-Irbe Chair in Computer Science & ECE and Distinguished University Professor at University of Maryland College Park. His research interests include virtual environments, physically-based modeling, and robotics. His group has developed a number of software packages that are standard and licensed to 60+ commercial vendors. He has published more than 600 papers & supervised 40 PhD dissertations. He is a Fellow of AAAI, AAAS, ACM, and IEEE, member of ACM SIGGRAPH Academy, and Bézier Award from Solid Modeling Association. He received the Distinguished Alumni Award from IIT Delhi the Distinguished Career in Computer Science Award from Washington Academy of Sciences. He was a co-founder of Impulsonic, a developer of physics-based audio simulation technologies, which was acquired by Valve Inc in November 2016.



JIA PAN received a B.S. in Control Theory and Engineering from Tsinghua University in 2005 and the M.S. degree from Chinese Academy of Sciences in 2008, where he worked on Computer-Aided Design (CAD), and the Ph.D. degree at the Department of Computer Science at the University of North Carolina at Chapel Hill (UNC) in 2013. In 2014, He completed a post-doc at UC Berkeley working with Pieter Abbeel. In October 2014, he started at the University of Hong Kong in the Computer Science Department, and then moved to the Department of Mechanical and Biomedical Engineering at the City University of Hong Kong as an assistant professor in 2015. After three years, he moved back to The University of Hong Kong and now is an assistant professor in the Computer Science Department. His research focuses on creating algorithms that allow robots to efficiently and intelligently interact with the world and collaborate with people. These general-purpose sensing, control, planning, and manipulation algorithms can be applied to robots that work in homes, factories, laboratories, or fields.



WEI ZHANG received the B.S. in Automatic Control from the University of Science and Technology of China in 2003, the M.S. in Electrical and Computer Engineering from the University of Kentucky in 2005, and the M.S. in Statistics and the Ph.D. in Electrical Engineering both from Purdue University in 2009. From January 2010 to August 2011, he was a Postdoctoral Researcher in the EECS Department at the University of California, Berkeley. He served as an Assistant Professor (2011 – 2017) and then an Associate Professor (with tenure) of Electrical and Computer Engineering at the Ohio State University. In May 2019, he joined the Southern University of Science and Technology, Shenzhen, China, where he is currently a Professor in the Department of Mechanical and Energy Engineering. His research interests include control and optimization theory, machine learning, motion planning, and their applications in robotics and autonomous systems. He is a recipient of the NSF CAREER award and the Lumley Research Award at the Ohio State University. He is currently a Senior Member of IEEE and an Associate Editor of the IEEE Transactions on Control System Technology.

...