Towards Socially Acceptable, Human-Aware Robot Navigation

Noelia Fernández Coleto(✉), Eduardo Ruiz Ramírez, Frederik Haarslev, and Leon Bodenhagen

SDU Robotics, Maersk Mc-Kinney Moller Insitute,
University of Southern Denmark, Odense, Denmark
fernandezcoletonoelia@gmail.com, {err,fh,lebo}@mmmi.sdu.dk

Abstract. The introduction of service robots to our daily life requires adaptation of the current navigation strategies. In the presence of humans, robots must be designed to ensure their safety and comfort. This paper proposes a layered costmap architecture that incorporates social norms to generate trajectories compatible with human preferences. The implemented framework creates a social abstraction of the environment – in the form of an occupancy grid – to plan human-friendly paths. It employs information about individuals in the scene to model their personal spaces. In addition, it uses predicted human trajectories to improve the efficiency and legibility of the robot trajectory. Different simulation scenarios resembling everyday situations have been used to evaluate the proposed framework. The results of the experiments have demonstrated its ability to behave according to social conventions. Furthermore, the navigation system was assessed in real life experiments where it was proved capable of following similar paths to those performed by humans.

Keywords: Socially acceptable navigation · HRI · Personal space

1 Introduction

The introduction of service robots in our daily life is growing day by day; hence, the need for intelligent mobile robots to be able to navigate in dynamic environments where the presence of humans is expected. Furthermore, a robot should not only carry out its tasks efficiently but also in a human-friendly way, adapting its behavior according to social conventions. During the last years researchers have developed different approaches to human-aware navigation; a commonly used strategy is based on the generation of a social map that models human space according to social conventions and interaction among individuals, using as a reference the studies presented by Hall [4] and Kendon et al. [5].

Papadakis et al. [8] proposed a method based on the use of non-stationary, skew-normal probability density functions for describing the social zones related to single individuals as well as associations among humans. The experiments carried out have proven the correct interpersonal spaces modeling of their proposed

M. A. Salichs et al. (Eds.): ICSR 2019, LNAI 11876, pp. 578–587, 2019.
https://doi.org/10.1007/978-3-030-35888-4_54

method; however, it entails the human interacting with the robot which is inadequate in some situations. A similar method was suggested by Charalampous et al. [3]. In addition, the approach presented by Vega et al. [11] proposed a human aware navigation strategy which uses an adaptive spatial density function to cluster groups of people according to their spatial arrangement and the environment. It uses also affordance spaces for defining potential activity spaces. The major drawback of this approach is that it assumes a static environment which can lead to an erratic robot navigation. Truong and Ngo [9] presented a solution that models group formations and human-object interactions, using social force models and the reciprocal velocity obstacle model. Experiments conducted in simulation and in an office environment demonstrate smooth operation in the presence of static or moving obstacles.

Kruse et al. [7] studied human behaviors in a path crossing situation in order to generate similar trajectories for robots. They use a linear projection of the human path based on his velocity to describe the context according to the human movement. Although the experiments demonstrated that the robot could perform distinct trajectories by velocity adaptation and waiting behavior, it has been only evaluated for the crossing scenario in simulation. On the other hand, the work in [6] uses predicted human trajectories and a social cost function to plan collision-free paths that take social constraints into account. It has been proven that it can generate consistent paths which respect to social conventions. However, the proposed method is only concerned with following a given trajectory, it does not perform any collision avoidance for unexpected obstacles, furthermore, the use of several descriptive costmaps involve a high computational cost if it would be utilized constantly during the navigation.

In order to equip the robot with a description of the situation, including information about individual personal spaces, possible groups arrangements, and future conflictive areas, this work proposes the development of three costmap layers as plugins for the current ROS Navigation Stack. This allows usage independence between layers and an easy implementation on systems using ROS as architecture. Each developed layer was individually tested under different circumstances to ensure it was capable of creating appropriate costmaps. Simulation and real life experiments were performed and their results were evaluated.

2 Methodology

The costmap layers developed in this work are explained in the following subsections. These layers work in combination with the default static, obstacle and inflation layers provided by ROS [1].

2.1 Human Layer

The *Human Layer* takes into account static humans in the environment. To ensure that the robot keeps a comfortable distance, a mixture of Gaussian distributions is used to calculate the cost of each cell around the human position.

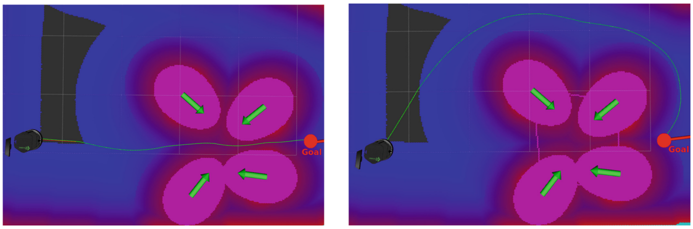


Fig. 1. Left: Path generated using only *Human Layer*. Right: Path generated using both *Human Layer* and *Proxemics Layer*. (Color figure online)

In order to achieve a distribution capable of adapting different shapes, five parameters that can be dynamically reconfigured were defined. Three of those parameters define the variance to model the distribution at the front, back and sides. Additionally, two thresholds were used to define the limits of the distribution. The first threshold was in charge of modeling the inner part of the distribution. It establishes the highest cost to the area surrounding the human. Likewise, the second threshold, models the outer part of the distribution, the cost of each cell will correspond to the cost of the Gaussian distribution until it reaches this second threshold, then the cost will be considered as free space. This approach allows to efficiently modify the used shape depending on the situation.

2.2 Proxemics Layer

The *Proxemics Layer* considers arrangements formed by static persons interacting in groups. This paper proposes a two step strategy to model groups formations. The first step groups the people in clusters according to the Euclidean distance between each of them. This assumes that two or more people that are in a close distance to each other, will be interacting. The next step is to model the space surrounding the group. The method proposed considers a polygon, which vertices are each person in the group. The edges of the polygon obtained are represented in the costmap with the highest cost possible. In the case that the group only have two members, a line connecting them is used. The fact that only the edges of the polygon are represented in the costmap significantly reduces the memory and computational costs required.

The combination of both *Human Layer* and *Proxemics Layer* has an outstanding performance. In Fig. 1, the green arrows represent the pose of four persons that seem to be having a conversation. The green line represents the global path plan between the robot and its goal. In the first case (Fig. 1a) only the *Human Layer* was used, therefore the robot moved through the group. On Fig. 1b, both layers were used, hence the path generated goes around the group, avoiding interfering with interpersonal interactions.

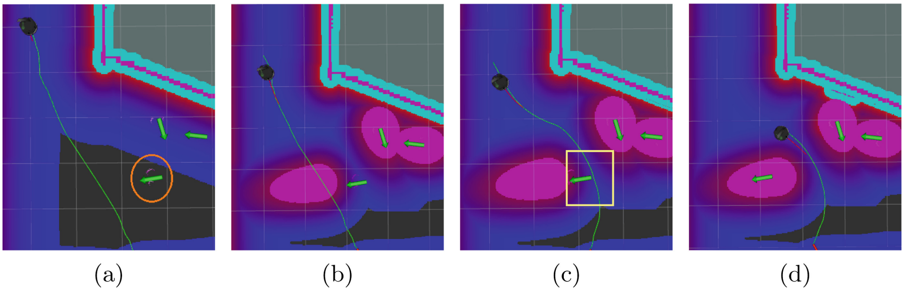


Fig. 2. (a) First trajectory generated. Orange circle represents the dynamic human (b) Master grid Costmap (c) New trajectory (d) Following new trajectory (Color figure online)

2.3 Prediction Layer

The *Prediction Layer* was created to determine possible regions that the robot and dynamic humans might share in the future. It estimates the robot and humans poses through a time period. The robot pose estimation is based on the point in the path generated by the global planner that the robot would reach every second, following its current velocity. On the other hand, the human estimation is based on a linear assumption, considering the actual and the previous pose and the time interval between them. For all the estimated poses it is found the shortest distance (collision point) between the robot and the human pose. The collision point is compared with a previous one, if any, and then if it is under a threshold it is considered into the costmap, modeling the region around the point using a mixture of Gaussian distributions, similarly to the one in the *Human Layer*. The estimation is performed every second, which is the global path update rate. As an example, a simulation experiment in which two humans are having a conversation, and a third person is walking, was considered (Fig. 2). The arrows and the green line represent the poses and the robot global path respectively. First, a global path was generated. The orange circle in Fig. 2a highlights the moving person. The collision point is determined using this path and the information from the moving person. The area enclosing the collision point is modeled using an Eggshape (Fig. 2b). Here, as the current path crossed a forbidden zone, it needed to be recalculated. Afterwards, a new path which considers the future estimation is calculated (Fig. 2c), even though it overlaps with the current human position (yellow square). Finally, the robot follows the planned trajectory avoiding the collision with the human (Fig. 2d).

2.4 ROS Implementation

As mentioned in the Introduction, there were created costmap plugins (one for each layer) following the shapes and procedures previously explained. Those plugins were loaded in the move_base node which calculates the velocity commands needed to move the robot to the designated goal taking into account

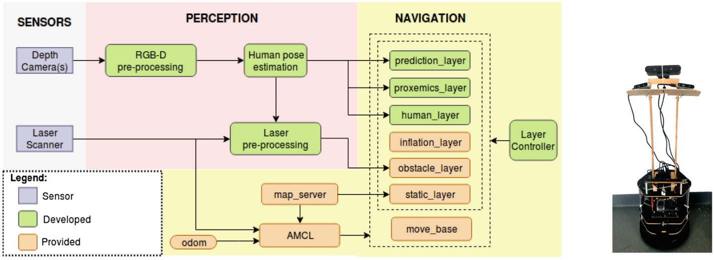


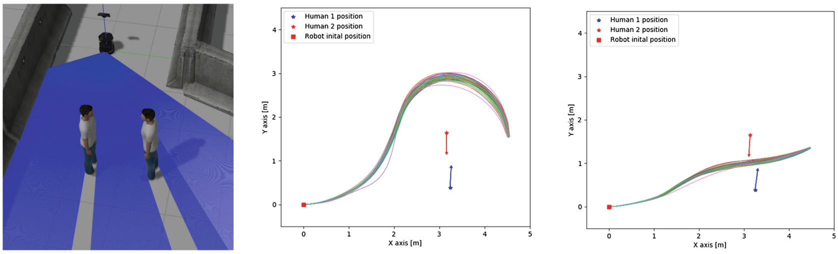
Fig. 3. Left: The software framework architecture. Right: The robot setup

the information of the Layered Costmaps. The implemented software framework architecture (Fig. 3) allows the replacement of modules with other similar ones, providing high versatility and convenience. The perception system comprised all the modules needed to extract relevant information from the cameras and the laser scanner. This includes a RGB-D pre-processing in charge of synchronizing the data captured by the cameras in order to provide the people detector a full FOV image. The perception system involves also the laser pre-processing module and filters out the scanner data corresponding to detected moving humans; therefore, they are not represented in the costmap created by the obstacle layer. Additionally, the navigation system was in charge of moving the robot to the desired position, it used the data obtained by the perception system. As part of this system, a layer controller was developed to determine which layers should be used. In crowded and narrow spaces, the proposed layered architecture could generate a particularly restrictive costmap, leading the robot to consider the desired goal as inaccessible. Using the layer controller, when the navigation system fails to find a path, it will disable the *Proxemics Layer* and the path planner will try to reach the goal again using the new generated costmap.

3 Experimental Results

The set of metrics given in [10] served as a basis to define the evaluation strategy used. Two kinds of metrics were used: *Objective metrics*, which measures the efficiency of the robot navigation (generated path length, navigation time and mission completion, number of times the robot successfully reaches the goal without colliding) and *subjective metrics*, which evaluates the robot trajectory according to social norms (minimum distance to human and number of times the robot performs a social space invasion).

Two approaches have been taken to evaluate the software framework according to the metrics exposed above, experiments in a simulated environment and in a real life setup. For the experiments the robot needed to navigate using the layered costmaps developed in this work. Besides, for comparison reasons, the navigation system using the default layered costmap offered by ROS was evaluated on both setups under the same circumstances.



(a) (b) Human-aware (c) Default

Fig. 4. Simulated scene (a) for experiment 1 and paths generated by the human-aware (b) or default (c) planner. Mean poses of persons represented by arrows.

3.1 Simulation Experiments

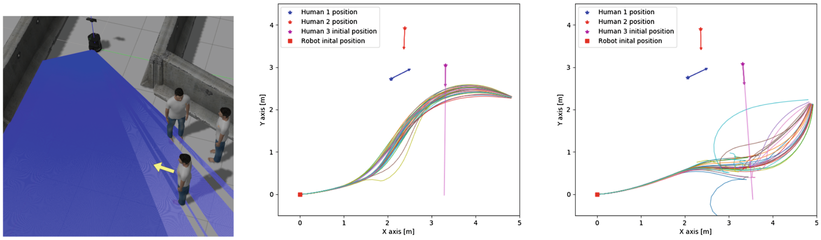
A Turtlebot 2 model was implemented in Gazebo to provide simulated sensor information to the ROS system. A generic laser scanner model was placed at the center front of the robot, and a generic RGB-D camera (represented in Gazebo as a Kinect) was placed in the robot center at a height of 1.05 m and a horizontal FOV of 180°. Human models poses were obtained from the models' information provided by Gazebo; hence, there was not applied any perception algorithm.

Two different experiments scenarios were considered. These scenarios were selected having in mind relevant situations where the robot should demonstrate a human-friendly behaviour. A total of 30 repetitions are run for each experiment, where human positions and orientations are randomly modified to resemble a realistic human scenario.

Experiment 1. Consisted of two static humans having a conversation (Fig. 4a). The paths followed in each repetition by the human-aware navigation system and the default navigation system can be seen in Fig. 4c. This illustration shows the robot's trajectory for each repetition. The humans are represented by an arrow whose location, and direction depends on the mean value of their detected positions and orientations.

Table 1 displays the mean value and standard deviations for the evaluation metrics proposed. As might be expected, the human-aware mobile robot traveled a longer distance and needed more time to reach the goal. The default navigation system provided by ROS interrupted the conversation in all the repetitions, whereas the proposed method respected the group and kept longer distances.

Experiment 2. In this experiment, there are three humans involved, two humans are having a conversation, and a third human is moving (Fig. 5a). The desired goal was selected in order to replicate a crossing situation, where the human path and the robot trajectory might concur. From the trajectories followed by the robot (Fig. 5c), it can be appreciated how the human-aware navigation system prevented the collision by navigating through the initial position



(a) (b) Human-aware (c) Default

Fig. 5. Simulated scene (a) for experiment 2 and paths generated by the human-aware (b) or default (c) planner. Mean poses of persons represented by arrows. (Color figure online)

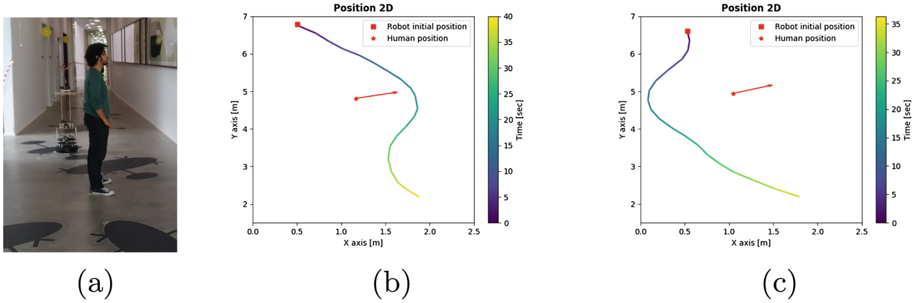
of the dynamic human (pink arrow); in contrast paths generated by the default path planner coincided with the human forcing the robot to modify the trajectory abruptly, or even colliding. The Human-Aware navigation system reached the goal without any trouble. On the other hand, the default approach collided with the human a total of eight times out of 30. Table 1 only shows evaluated data from successful repetitions.

3.2 Real Life Experiments

In addition to the Turtlebot, a Hokuyo URG-04LX laser scanner was placed in the same position as the simulation. Two Intel RealSense D435 cameras were placed at the same height as in the simulated setup and separated by 18.5 cm. They were positioned with a pitch rotation of −20° in order to focus on people's upper body. Additionally a yaw rotation was applied to each of the cameras in order to have a merged horizontal FOV of approximately 135° (Fig. 3). The human pose estimation was done using OpenPose [2]. The detected keypoints were projected to 3D by looking up the coordinates in the RGB-D point cloud.

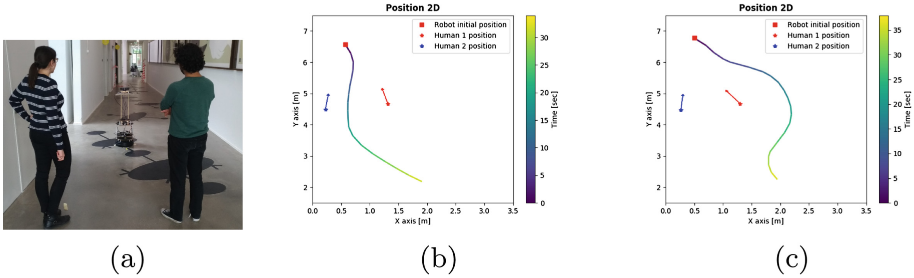
Table 1. Results for simulation experiments (standard deviations in brackets).

Planner	Objective metrics			Subjective metrics			
	Path length [m]	Time [s]	Mission completed	Min distance [m]			Social space invasions
				H1	H2	H3	
Experiment 1, n = 30							
Human-aware	6.764 (0.114)	22.055 (0.495)	100%	1.610 (0.059)	1.083 (0.033)	-	0%
Default	4.715 (0.014)	12.345 (0.182)	100%	0.636 (0.041)	0.638 (0.046)	-	100%
Experiment 2, n = 30							
Human-aware	8.518 (0.095)	16.178 (0.428)	100%	1.001 (0.049)	1.814 (0.076)	0.872 (0.066)	-
Default	5.876 (0.194)	16.958 (2.711)	73.33%	3.099 (0.196)	3.031 (0.204)	0.402 (0.166)	-



(a) (b) (c)

Fig. 6. The setup (a) for the real life experiment 1 and example trajectories for the proposed (b) and default (c) planner.



(a) (b) (c)

Fig. 7. The setup (a) for the real life experiment 2 and example trajectories for the proposed (b) and default (c) planner.

The orientation of each detected person was calculated by projecting the two shoulder keypoints to the floor plane, and calculating the vector orthogonal to the line between the two points. The detected neck keypoint was chosen as the position of the person. All system processing including neural network computations were done on a Jetson TX2.

A total of six repetitions were run per experiment. In this case, the humans taking part in the experiment were static since the perception system used is not able to extract an adequate number of pose estimations from a moving human.

Experiment 1. It involves a human standing in the middle of a corridor, looking at the paintings in the wall. The robot should transverse the hall without interrupting the human's activity. In Fig. 6, it can be observed one of the repetitions carried out for the proposed navigation systems and the default navigation system provided by ROS. Red arrows depict the mean detected human position and orientation calculated using the information from the perception system during the corresponding repetition. As might be expected, the human-aware system moved behind the person; in contrast, the default system crossed in front of him, interrupting the activity. A comparison between the proposed approach and the default navigation system can be found in Table 2.

Table 2. Results for real life experiments (standard deviations in brackets).

Planner	Objective metrics			Subjective metrics		
	Path length [m]	Time [s]	Mission completed	Min distance [m]		Social space invasions
				H1	H2	
Experiment 1, n = 6						
Human-aware	5.531 (0.113)	44.388 (2.445)	100%	0.961 (0.072)	-	0%
Default	5.324 (0.147)	44.722 (7.158)	100%	0.700 (0.073)	-	83.33%
Experiment 2, n = 6						
Human-aware	5.459 (0.101)	46.111 (10.432)	100%	0.854 (0.055)	1.708 (0.142)	0%
Default	5.216 (0.092)	40.333 (2.880)	100%	0.661 (0.453)	0.661 (0.453)	83.33%

Experiment 2. It involves two persons standing in front of the robot (Fig. 7a). In this case, humans might be having a conversation, therefore the robot should avoid to cross between them. The default navigation system (Fig. 7b) interrupted the conversation, whereas the proposed method behaves in a human-friendly way (Fig. 7c). Table 2 shows further information about this experiment.

4 Conclusions

The paper in hand proposed an integrated navigation system which enables a mobile robot to behave according to social conventions and generates socially acceptable trajectories in a human populated environment. The layered costmap suggested, have been integrated into the current version of ROS navigation stack. Therefore, it can be easily added to any navigation system based on ROS while maintaining the advantages of the local and global path planners. Moreover, the layers have been developed having in mind the memory and computational costs associated with each costmap. The *Proxemic Layer* only represents the edges of a polygon since its combination with the *Human Layer* provides adequate information about the scenario, reducing the computational effort.

While other studies proposed solutions that are only applicable under certain circumstances, the suggested approach demonstrates how a detailed layered costmap is more versatile, and it can be used in a broader range of situations. Its capability of respecting personal spaces and group formations, as well as avoiding conflictive regions, has been proven not only in simulation but also in real life experiments. In particular, the results obtained in simulation verify that in order to achieve suitable trajectories, a comprehensive layered costmap is needed. Furthermore, the evaluation in real life experiments demonstrates the correct performance of the method and its feasibility.

Future improvements involve the use of a robust and reliable perception system, capable of detecting the poses of multiple of humans. This would allow the evaluation of the method in complex constellations of persons. Furthermore, the persons orientation can be taken into account in the group formation determined by the *Proxemics Layer*. Besides, the performance of the *Prediction Layer* could be assessed using real life experiments.

Acknowledgment. This work was supported by the project Health-CAT, funded by the European Fund for regional development.

References

1. ROS navigation stack (2007). http://wiki.ros.org/navigation
2. Cao, Z., Simon, T., Wei, S., Sheikh, Y.: Realtime multi-person 2D pose estimation using part affinity fields. CoRR abs/1611.08050 (2016)
3. Charalampous, K., Kostavelis, I., Gasteratos, A.: Robot navigation in large-scale social maps: an action recognition approach. Expert Syst. Appl. **66**, 261–273 (2016)
4. Hall, E.T.: The hidden dimension (1966)
5. Kendon, A.: Conducting Interaction: Patterns of Behavior in Focused Encounters. Studies in Interactional Sociolinguistics (1990)
6. Kollmitz, M., Hsiao, K., Gaa, J., Burgard, W.: Time dependent planning on a layered social cost map for human-aware robot navigation. In: European Conference on Mobile Robots (2015)
7. Kruse, T., Basili, P., Glasauer, S., Kirsch, A.: Legible robot navigation in the proximity of moving humans. In: IEEE Works on Advanced Robotics and its Social Impacts (2012)
8. Papadakis, P., Rives, P., Spalanzani, A.: Adaptive spacing in human-robot interactions. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (2014)
9. Truong, X., Ngo, T.D.: Toward socially aware robot navigation in dynamic and crowded environments: a proactive social motion model. IEEE Trans. Autom. Sci. Eng. **14**(4), 1743–1760 (2017)
10. Vasquez, D., Okal, B., Arras, K.O.: Inverse reinforcement learning algorithms and features for robot navigation in crowds: an experimental comparison. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (2014)
11. 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. **118**, 72–84 (2019)