ulm university universität uulm

Ulm University | 89069 Ulm | Germany

Faculty of Engineering,
Computer Science and Psychology
Institute of Artificial Intelligence

Navigating without Intruding Humans' Social Space: A Cost-based Approach

Project Report at Ulm University

Presented by:
Carolin Schindler
carolin.schindler@uni-ulm.de

Examiner:
Jun.-Prof. Dr. Felix Lindner

Advisor:
Jakob Karalus

2021/22

Last updated July 5, 2022

© 2021/22 Carolin Schindler

This work is licensed under the Creative Commons
Attribution-NonCommercial-NoDerivatives 4.0 International License.
To view a copy of this license, visit
https://creativecommons.org/licenses/by-nc-nd/4.0/.

Typesetting: PDF-LaTeX 2_ε

Abstract

When robots are employed in human environments, they should conform to social values and norms. This way, humans can easily understand the behaviour of the robot and are not getting discomforted by it. Navigation is one of the basic tasks an autonomous robot has to accomplish. Social behaviour is involved, even when the robot just has to navigate to a given destination without interacting explicitly with any person. More precisely, the robot should always respect people's social space. This space does not only entail the area around a person but considers their interactions with objects and other people as well. To this end, we create a modular model of humans' social space by means of a cost function. We employ this model as a generic costmap in a navigation layer within the ROS framework. In a simulated environment, we qualitatively verify the applicability of our approach to human-aware robot navigation in terms of navigation performance (successfully reaching the goal and path efficiency), behavioural naturalness (smoothness of the trajectory) and human comfort (no collisions and no intrusions of social spaces). The results show that our holistic view on the social space of humans has the potential to improve the social behaviour of the robot compared to an existing implementation in ROS that only considers the space around people themselves.

Contents

1 Introduction **1**

2 Related Work **3**
2.1 A Person's Social Space 3
2.1.1 Person-related Spaces 5
2.1.2 Activity and Affordance Spaces 6
2.2 Human-aware Robot Navigation 9
2.2.1 Approaching Humans 9
2.2.2 Companioning Humans 11
2.2.3 Navigating to a given Destination 12
2.3 Summary 15

3 Methods **17**
3.1 Modular Cost Function for Modelling Humans' Social Space 17
3.1.1 Person-related Spaces 18
3.1.2 Activity and Affordance Spaces 21
3.1.3 Combining the individual Sub-Spaces 24
3.2 Social Space Navigation Layer 25
3.3 Implementation Details 27

4 Evaluation **29**
4.1 Scenarios 30
4.1.1 Scenario1: Interaction with an Object 30
4.1.2 Scenario2: Activity vs Affordance Space 31
4.1.3 Scenario3: Interaction between two Humans 33
4.2 Measures 34

5 Results **35**
5.1 Scenario1: Interaction with an Object 35
5.2 Scenario2: Activity vs Affordance Space 37
5.3 Scenario3: Interaction between two Humans 38

6 Discussion **39**
6.1 Social Space Navigation Layer 39

6.2 Social Space Cost Function 40
6.3 Other Aspects 41

7 Conclusion **43**

List of Figures

2.1 F-Formation as described by Kendon [8, 17]. 8

3.1 Depiction of the modular structure of our cost function for modelling humans' social space. 18
3.2 Exemplary visualisation of the two person-related spaces. 19
3.3 Exemplary visualisation of the two kinds of activity spaces. 21

4.1 Visualisation of Scenario1 31
4.2 Visualisation of Scenario2 32
4.3 Visualisation of Scenario3 33

5.1 Resulting navigation trajectories for Scenario1. 36
5.2 Resulting navigation trajectories for Scenario2. 37
5.3 Resulting navigation trajectories for Scenario3. 38

List of Tables

2.1 Interpersonal distance zones as defined by Hall [12]. 6

1 Introduction

Service robots [24] are gaining more and more prominence in our everyday lives, whether they are employed in professional or domestic/personal areas. When these robots get in contact with humans, it is desirable that they behave in a socially acceptable manner, i. e., the service robot should be a social robot [26] conforming to social values and norms.

This likewise holds true for the basic task of navigation [23, 31, 38]. Even when there is no explicit interaction with humans, the robot is nevertheless sharing its environment with them. Thus, the robot can cause misunderstandings and make people feel uncomfortable or even unsafe due to its chosen trajectory, for instance when it is intruding people's social space.

Hence, the navigation algorithm should not simply treat people like regular obstacles but take their social space into consideration. There exist different approaches for incorporating social norms into the navigation, e. g., in terms of manually defined costs and rewards, by machine learning, or even by direct human feedback (see Section 2.2 for a more detailed overview).

However, many works considering a specific interaction with humans are skipping the preceding navigation until their navigation scenario starts (e. g., [2, 43, 46]). Other works that are concerned with the navigation itself most often only take parts of the social space into account (e. g., [6, 15, 18, 29, 30, 37, 39, 42]). They, for example, observe the personal space while ignoring the possibility of interactions with either objects, other people, or even both.

We create a modular cost function of the social space of humans for navigating a robot in a socially accepted manner to a given destination. It is important to note that we construct the cost function in such a way that the resulting costs can be entered into the costmap utilized for navigation, i. e., we do not manipulate the planning algorithm itself but the costs on the map consumed by it. Thereby, we consider person-related spaces as well as activity and affordance spaces arising due to interactions with objects and other people. Furthermore, we implement the cost function as a navigation layer [29] within the robot operating system (ROS) [36] assuming that required detections of humans and their interactions are available. This way, the herein presented avoidance of humans' social space can be easily added into the navigation stack of any existing robot utilizing the ROS framework.

In summary, the contributions made in this work are:

- A modular cost function for modelling humans' social space. To this end, we performed an extensive literature review on modelling a person's social space including existing cost-based navigation approaches.
- Employment of this model as a generic costmap layer for social navigation.
- A qualitative evaluation of the model and its applicability to human-aware robot navigation.

The remainder of this report is structured as follows: In the next Chapter, we introduce concepts relevant to modelling a person's social space and give an overview of applications of human-aware robot navigation. Our methodology in Chapter 3 is split into two parts: First, we create the modular model of the social space of humans by means of cost functions. Second, this model is implemented as a navigation layer within the ROS [36] framework. In Chapter 4, we set up different scenarios in the simulation framework Gazebo [20] and instruct the robot TIAGo [33, 34] to navigate to a given goal. Therewith we aim at verifying our approach and the implementation of the social space model itself. The results of the aforementioned experiments are presented in Chapter 5 and are further discussed in Chapter 6. We conclude in Chapter 7 with a summary.

2 Related Work

We describe the individual parts a person's social space is composed of in Section 2.1. Afterwards, Section 2.2 gives an overview of applications and approaches to human-aware robot navigation. We review navigation tasks including human-robot interaction as well as simply navigating to a predefined goal. The latter is our scenario, nevertheless we can likewise borrow concepts from cost-based variants of the former. In the end, we give a short summary of the related work.

2.1 A Person's Social Space

In order to be able to create a modular cost function of the social space of humans, we first need to investigate what belongs to the social space of a person, which of these parts are relevant to our navigation scenario, and how to represent these. Therefore, we review existing research in the field of proxemics and social sciences. Since people seem to treat a robot similar to a human when sharing their space [18, 38] (at least when the robot has human-like dimensions [43]), we can make use of these findings even when no robots but only humans are considered in the investigation. As a consequence, the robot itself constitutes a social space [18, 27]. This means, not only the social space of the humans around the robot but the robot's own social space require consideration. In general, the cost function created for humans can be applied to the robot as well [18].

In the remainder of this section, we separate the social space of a person on a meta-level into person-related space and activity and affordance space. For the former, we review literature regarding the personal space of humans and their information processing space. The latter distinguishes between interactions with objects and humans.

Lindner and Eschenbach [27] formalized the notion of social space for human-aware robots by separating it into the following five types: personal space, activity space, affordance space, territory, and penetrated space. Compared to them, we do not consider territory and penetrated space. In our scenario, we assume that there is no territorial behaviour [27]. This means, there is no owning, claiming or defending of room. The robot as well as all humans have equal rights to be in any area. Penetrated spaces [27] are spaces with negative side effects including for instance noise, odor, or dust. Often, these side effects arise from executed activities. Considering penetrated

spaces is important when the robot is meant to interact with humans. Furthermore, the robot should be aware of social consequences when creating a penetrated space by itself. As our robot is not interacting with objects or humans, but simply navigating in the environment, we can ignore the notion of penetrated space. Navigating through a penetrated space is generally not an issue since there are usually only people responsible for this space but no one claiming it.

In a survey paper on proxemics-based socially-aware navigation [38], four kinds of spaces are distinguished. Each space is regarding a different social situation, namely: individuals themselves with personal space and information processing space, groups of people interacting (i. e., human-human interaction), human-object interaction, and human-robot interaction. As mentioned earlier, explicit interactions between humans and the robot are not part of our scenario. Hence, we are not interested in robot proxemics [38] including factors such as speed, appearance, direction of approach, or gaze.

Yet, another but more general survey [23] on human-aware robot navigation, presents a list of utilized cost functions. It entails the following social costs: hidden zones, basic comfort distance (i. e., personal space), visibility, interaction regions with objects (i. e., activity space), pass on the left, space ahead for moving, discard compatible, and crowd density along with velocity similarity. Cost for hidden zones aim at preventing that the robot appears surprisingly from behind objects in relation to nearby humans. Visibility costs force the robot to move where the humans can see it. This is mostly important when approaching humans. Since we do not consider human-robot interaction and visibility is contradictory to avoiding interaction regions, we do not consider these costs. To incooperate the social norm of passing people either on the right or the left (depending on cultural traffic rules), one can add costs on the respective other side of the humans. Additionally, Kirby [18] found that such passing costs are only relevant for moving people. Also only relevant for moving people is adding costs ahead of them. This way, a robot planning in a non-temporal manner is informed that this space will be occupied in near future. We include this concept in our review on personal space and information processing space. Discard compatibility addresses issues with generally adding space ahead costs as they might result in unnatural motions by the robot. Finally, we are not considering crowd density and velocity similarity since our navigation scenario is not aimed at crowd motion.

Representing the social space in terms of geometry is a rather tricky task since the shape and size of a person's social space depends on a variety of factors [27], including context, time, and influences between several spaces. Not only the geometry of the different sub-spaces making up the social space but also the combination of them is subject to tuning [23]. In general, the boundaries of the spaces are vaguely defined [32]. This means that costs decrease in a linear, exponential, or sinusoid

manner for example [23]. In order to combine several cost functions, one can use weighted sums, the maximum operation, or context-dependent enabling of costs [23]. It is even possible to make use of machine learning techniques in order to learn a representation of people's social behaviour [23]. We, however, stick with manually defining the geometry of the sub-spaces and their combination. In order to do so, we build a theoretical grounding with the subsequent research on person-related space and activity and affordance spaces. Additionally, we borrow modelling concepts from existing human-aware navigation systems (see Section 2.2).

2.1.1 Person-related Spaces

In the following, we review the personal space and the information processing space, which are both related to individual persons.

Personal Space Hall [12] conducted an investigation on interpersonal distance resulting in a model of four distinct zones, each with a close and a far phase. Their results are summarized in Table 2.1. The given distances need to be taken with care since they generally depend on cultural, personal, and environmental factors. The intimate zone involves direct physical contact. In the personal distance zone, one can identify the relationship between people by means of their distance. To non-related people, this zone is like a protective area or bubble that should be maintained. At social distance, no physical contact takes place. It is the zone for impersonal business, working together with other people, or when being at a regular social gathering. The further the distance in the social zone, the more formal the setting. Public distance marks the boundary of involvement and is in theory infinitely large. It is used for public figures and public occasions.

The personal space of a human consists of their intimate and personal zone [38]. Intrusion of a personal space leads to discomfort [18, 38], i. e., it captures the "bubble-characteristic" of the personal distance zone.

The survey by Rios-Martinez et al. [38] points out different shapes of personal space: In Hall's [12] model, the different zones and hence the personal space are concentric circles. Other works suggest to use an egg shape since intrusions to the front of a person are regarded as more uncomfortable. The social force model by Helbing and Molnár [13] makes use of concentric ellipses oriented into the direction of pedestrian movement. Similarly, the personal space can be modelled by an asymmetric shape with less space at the person's dominant side.

Furthermore, the personal space is dependent on the situation and hence dynamic [38]. Even the activity executed by the person can have an impact on the

Table 2.1: Interpersonal distance zones as defined by Hall [12] for a sample group of Americans. Each zone has a close and a far phase.

distance zones		**distance from person [m]** from	to
intimate distance	close	0	0.15
	far	0.15	0.46
personal distance	close	0.46	0.76
	far	0.76	1.22
social distance	close	1.22	2.13
	far	2.13	3.66
public distance	close	3.66	7.62
	far	7.62	∞

size and shape of personal space [32]. For instance, the personal space may be enlarged ahead of a walking person depending on their velocity [23, 32].

Information Processing Space The information processing space (IPS) is an area in front of pedestrians in which all objects are treated as potential obstacles when planning the next movement steps [19]. Experiments by Kitazawa and Fujiyama [19] evaluate the fixations of pedestrians and show that the shape of the IPS follows a cone rather then the so-far proposed field-of-view semicircle. More precisely [19, 38], this cone reaches 4.5m into the walking direction. In this, it has an angle of 45° to each side until it reaches a total width of 1.0m. The IPS is related to visual behaviour which seems to be dependent on cultural and personal factors [38]. Kitazawa and Fujiyama [19] moreover made the finding that the attention given to ground surfaces is much higher than that given to obstacles. Furthermore, they conclude that static obstacles are more relevant than approaching pedestrians when planning future trajectories.

2.1.2 Activity and Affordance Spaces

An executed activity requires a certain amount of space. Ostermann and Timpf [32] name this space *activity footprint*. They do not only consider activities with objects and other people but also non-interactive activities. Furthermore, an activity footprint does not only entail the activity space as we and Lindner and Eschenbach [27] describe it. An activity footprint also contains spaces related to the activity such as penetrated space. To us an activity space is an area of direct and explicit interaction of a person with other humans or objects, i. e., passing such a space is socially not accepted [27, 38] as it would interrupt the respective interaction.

Since affordances are potential activities, an affordance space is a potential activity space [27, 38]. Gibson [10] introduced the term affordance and coined it as a concept tied to the environment as well as the individual. Affordances are provided by the environment but are perceived uniquely by every person. This way, an affordance is neither subjective nor objective, neither physical nor psychical.

It is important to distinguish activity from affordance spaces [27]. While activity spaces should not be entered, an affordance space might be passed when not remaining in and hence blocking it [38].

Some literature [23, 38] views activity and affordance spaces as only belonging to interactions with objects. We, however, use the concept of activity and affordance space for interactions with other humans as well [27]. Gibson [10] included among others persons as an affordance, e. g., a person might afford talking to them. Since a person constitutes an affordance, there is a respective affordance and activity space related to this interaction.

The geometric shape of activity and affordance space depends on the specific activity or affordance giving rise to it [27]. Some activity spaces have well-defined footprints, while others lack a clear and unique definition [32]. Ostermann and Timpf [32] also point out that the activity footprint for people talking is the same as the personal space of the individual persons involved in the talk. In contrast, Rios-Martinez et al. [38] conclude in their survey that is important to study interactions of humans separately from their personal space. This is due to the fact that people tend to keep more space around a group than the single personal spaces indicate.

Interaction with Objects As mentioned above, the shape of activity and affordance space depends on the object interacted with. To our knowledge, there exists no widely accepted theory for the space created by interaction with objects. In order to model activity footprints, Ostermann and Timpf [32] used circular shapes, whose radius was adjusted according to the type of activity. However, as pointed out by the authors themselves, the choice of a circle was just due to reasons of simplicity. They would have preferred to actually utilize different shapes instead of different radii only. Furthermore, they did not model the activity space separately from the personal space. Instead, the personal space was influenced by the executed activity. The formalization of social space by Lindner and Eschenbach [27] is based on topological specifications and excluded the definition of geometric shapes. Nevertheless, they utilized the field of view between a person and the object they interacted with in an exemplary scenario.

Interaction with other People There is the notion of jointly focused, common focused, and unfocused interaction [17, 38, 41]. Unfocused interactions may not involve talking and are for example waiting in a queue or a gathering. This means, there are people

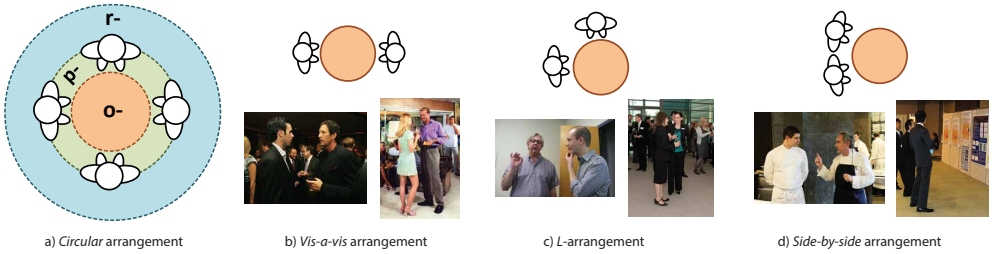


Figure 2.1: F-Formation as described by Kendon [8, 17].
a) Depiction of the three functional spaces of an F-Formation.
b), c), and d) Exemplary arrangements of two people.
Source: see Setti et al. [41]
CC BY 4.0, https://creativecommons.org/licenses/by/4.0

interacting with each other but not in an intended manner. Common focused interactions are performer-audience scenarios, parades, or demonstrations for instance. That is, there are a few people and they all do something in unison. There exist predefined formations for some unfocused and common focused interactions (e. g., waiting in a queue). Since our robot is not going to join such formations but bypasses them, we make the assumption that modelling the personal spaces of the individuals is sufficient in such cases. When people are commonly focusing on an object or another person, we include this explicit interaction for each person individually. Finally, jointly focused interactions involve meeting and talking with at least one other person. These are the interactions with other people we are interested in, which require more space than the personal spaces of the individuals together [38].

By observing spatial organisations of people, Kendon [8, 17] introduced the concept of F-Formation as a way of representing jointly focused interactions where every participant has equal rights. An F-Formation can be separated into three distinct functional spaces as depicted in Figure 2.1a. The O-Space is the shared, inner space where the actual activity of the group, such as talking, eating or playing a game, takes place. In the P-Space, which is surrounding the O-Space, all people and their belongings are positioned. Only people that are in the P-Space, are equal members of the group. The R-Space is surrounding the group and has in theory no boundary. It is not directly used by the participants of the group but still monitored by them and perceived by people who are not part of the group. When a person wants to join a group, they tend to wait in the R-Space until the people in the F-Formation either make room for them to join the P-Space or decline their bid for entry. This and the space people keep when bypassing F-Formations, gives clues on the extend of the R-Space and hence the F-Formation itself.

In Figure 2.1a, the F-Formation is illustrated for a circular arrangement. However, the spatial-orientational arrangement of people in a focused interaction generally depends on the number of people in the group, their relationship, the activity executed, and environmental constraints [17, 38]. The circular arrangement is often used for conversations including more than two people. When two people are in a conversation, they either stand in an L- or a vis-a-vis arrangement as shown in Figure 2.1b and 2.1c respectively. The chosen arrangement depends on the topic under discussion. When the topic is regarding the two people themselves, the vis-a-vis arrangement is utilized, otherwise the L-arrangement. There are occasions where people are jointly interacting with each other but at the same time commonly focus on some item in the environment. For this purpose, two people engage in a side-by-side arrangement as sketched in Figure 2.1c. This arrangement is also entered by more than two people resulting in a horseshoe-like shape. There exist more spatial arrangements of F-Formations, especially in case of two people interacting with each other [8, 38]. Nevertheless, as illustrated in Figure 2.1, we can fit an O-Space and hence a circular F-Formation to any of these arrangements.

2.2 Human-aware Robot Navigation

In human-aware robot navigation [23, 31, 38], the humans are considered during the navigation, i. e., they are not simply treated like regular obstacles. Thereby, it does not matter whether the robot explicitly interacts with the humans or is just navigating around them. Approaching and companioning humans are examples for the former. In our scenario, where the robot navigates to a given destination, no explicit interaction is scheduled. In the following, we review literature in these three areas of application and discuss their inclusion of people's social space.

2.2.1 Approaching Humans

When approaching humans, the main questions are from which direction to approach and where to stop, i. e., how close to go. The following literature can be separated into three categories depending on the scenario they investigate.

Approaching a Person In this scenario, the robot decides to approach a person and draw their attention in order to start an interaction. Torta et al. [43] created a parametric model of a person's personal space. This model indicates in a reward-based manner where the robot should navigate to such that the human will become aware of it. Resembling the results of their conducted psychometric experiments, the model takes the distance to the human and the direction of approach into account. Additionally, they apply a different parametrization of the model depending on whether the human is sitting

or standing. Based on their work, Bai [2] adds the notion of activity, affordance and blind space for interactions with objects to the model. The blind space is the area behind objects interacted with that is not observable by the human. The shape for modelling activity and affordance spaces follows the field-of-view paradigm. The individual spaces are combined by multiplying their normalized values. This way, the robot is prevented from aiming at a goal position that lies within any of these spaces. Moreover, they speculate that the stopping distance is dependent on the human activity in general. More precisely, the distance has to be adjusted for groups of humans and the distance should decrease when the human is engaging with an object at short distance. The latter allows for a shorter attention shifting distance by the human.

We borrow Bai's [2] way of modelling the affordance space, the activity space of objects and combining the individual spaces. By doing so, we need to transfer their methods from the reward-based setting to a cost-based one. Instead of making use of their results and speculations that the size of the personal space depends on the state of the human, we simply assume a conservative personal space model whose shape is only influenced by the velocity of the person. Nonetheless, our modularity allows to easily modify this in future work. Bai [2] furthermore implemented the required preceding detection of humans, objects and their interactions by a combination of existing machine learning models with hand-crafted rules and optimization procedures. We, however, leave this detection to future work and assume it to be given.

Approaching a Group of People This is similar to the above scenario, however, the robot is now approaching a group of people instead of a single person. Hence, it is required to detect F-Formations additionally to the individual persons. You and Ji [46] simulate a robotic delivery service where the aimed person is standing within a group of people. For simplicity, they set the goal of the navigation to the centre of the O-space as done in a previous work by Rios-Martinez et al. [39] as well. In both works, the F-Formation detection is performed by hand-crafted rules, taking the position and orientation of the people within the group as an input. Rios-Martinez et al. [39] add the notion of stopping distance to the task since the robot should actually join the group for conversation. Additionally, they also include scenarios where the robot is not joining the group but bypassing it (see Section 2.2.3). You and Ji [46] are more concerned with the direction of approach. They make sure that the robot approaches from the wide direction, i. e., where the gap between two succeeding persons is the largest, by introducing a guard line.

As a matter of form, we mention the work by Banisetty et al. [3] in this context as well, but go into more detail on their approach in Section 2.2.3 since they present a more general approach for navigating to a given destination in a human-aware manner. They introduced the notion of a social goal for human-robot interaction into their planning

algorithm, which the robot gets rewarded for reaching. That is, instead of navigating to the actual goal, the robot infers a social goal which then becomes the new destination. The calculation of the social goal is performed by means of geometrical computations for joining a group of people in an F-Formation and for waiting in a queue. In case of the F-Formation, they consider both stopping distance and direction of approach.

There exist more sophisticated algorithms to detect F-Formations and their centres, e. g., by means of computer vision methods [41, 45]. Barua et al. [4] even create a machine learning pipeline to detect F-Formations jointly with the direction of approach based on an egocentric view by the robot. In our work, we assume that F-Formations are detected along with their centres.

Human-initiated Approaching of a Person In the so far reviewed work, human-robot interactions were always initiated by the robot. Svenstrup et al. [42] investigate how a robot can detect that a human wants to interact with it and act accordingly. In their scenario, the robot is moving in the environment respecting people's personal space. Only when the robot senses an intention for interaction, it deliberately intrudes the personal space of the respective human. To this end, they detect individual person's position, velocity and orientation by a Kalman filter-based algorithm and apply the sum of four Gaussian distributions in order to model the potential field around them. This potential field is utilized by the navigation algorithm and depends on the likelihood that a person wants to interact with the robot.

We do not consider human-robot interactions during the navigation. Hence, this line of research is out of scope for this work. Therefore, we also do not require the expressive power of four individual Gaussian distributions for modelling the personal space.

2.2.2 Companioning Humans

In the following, we review merely two different companioning tasks. Hence, this is just a brief glimpse into this area where a robot is navigating together with humans.

Accompanying Humans How a robot can accompany one or two people while moving in a populated environment is investigated among others by Repiso et al. [37]. Their robot first selects the most probable final destination of the group by making use of points of interest and movement prediction of all humans in the environment. The navigation planner then plans a trajectory to the selected goal taking additional forces and costs introduced by the accompaniment task into account. The forces are needed when computing possible trajectories, while the costs are utilized for selecting one of the trajectories. At the same time, the robot needs to avoid other pedestrians and obstacles.

When the robot accompanies a single person, they walk in a side-by-side configuration. When the robot is walking with two other people, it forms a V-configuration with them.

Since we are not concerned with navigating together with other people, we do not make use of these additional configurations. Our method treats people walking in a group like an F-Formation that is moving in the environment. Again, one could add a special treatment of moving groups into our model in future work if required.

"Following" a Person Another companion scenario is where the robot unobtrusively follows a human in order to support them in tasks at their destination. Bruckschen et al. [6] investigate this scenario with a service robot that does not necessarily need to follow the human, but arrive at the initially unknown destination at the same time as the human. In order to infer the navigation goal, they perform a long-term movement prediction of the person's trajectory. Thereby, they assume that the robot possesses a map of the environment containing the locations of all relevant objects. Based on user studies about human comfort, they model the space around the human with costs as follows: The personal space and the information processing space (IPS) are impassable. The area behind the human comprises costs in order to ensure that the robot is not perceived as a follower. However, the robot might temporary enter this area if required. Since they inferred the complete trajectory of the human, they can create a time-dependent costmap into the future which is applied when planning the path of the robot.

In our model of social space, the IPS is implicitly entailed in the personal space whose shape depends on the velocity of the human. Additionally, moving humans – and hence most likely navigating robots as well – are not as relevant in the IPS as static obstacles or the ground surface [19]. Moreover, we do not model an additional area behind persons as it would be pure chance when the robot follows a human while navigating independently. Hence, we regard the area created by the personal space as sufficient for our scenario. However, a time-dependent costmap making use of forecasting human movement and activities is a reasonable extension that we leave to future work.

2.2.3 Navigating to a given Destination

This is our scenario, where the robot has to navigate to a given destination in a human-aware manner, i. e., the robot should not make any people feel uncomfortable and choose its trajectory in a socially accepted way. The most interesting to us are works that defined their costs manually since this resembles our approach. Nevertheless, we give a short outlook on navigation algorithms based on machine learning and human feedback.

Manually defined Costs and Rewards Kirby [18] collected and implemented constraints for social robot navigation. They explored scenarios where the robot is navigating independently among humans and passing them in a hallway. As an extension, they also implemented constraints for side-by-side escorting, which we are not reviewing here. The social constraints introduced by them are termed people avoidance, personal space, robot space, pass on the right, default velocity, face travel, and inertia. People avoidance is similar to avoiding dynamic obstacles and hence does not necessarily require a separate implementation. A major difference between people and object avoidance is that people might react to the presence of the robot, e. g., by making room for it. The robot space is the personal space of the robot. We cannot make use of this concept since we can only consider costs that can be reasonably drawn into the map applied for navigation. Finally, default velocity, face travel and inertia are related to the task at hand and cannot be applied in a costmap as well. Kirby [18] combines the individual costs by a weighted sum. The behaviour of the robot depends on the selection of the weights. They manually selected the weights, however, there is the possibility to learn these weights by machine learning methods, e. g., via inverse reinforcement learning [7, 14].

We utilize their cost function for personal space and passing on the right (or left respectively). For both costs, the underlying shape is an asymmetric Gaussian. The egg shape of the personal space additionally depends on the velocity of the pedestrian and passing costs are only applied for moving people. Note, that their modelling of personal and passing space is already implemented in the existing social navigation layers[1] [29] in ROS. We are going to utilize this implementation as a baseline. Differently to Kirby's [18] work, the two costs are combined by the maximum operator in the ROS implementation.

We have already named the work by Rios-Martinez et al. [39] for joining a group of people. They modelled the F-Formation by a two-dimensional Gaussian whose parametrization depends on the specific arrangement of the people. Additionally, they implemented the personal space according to Amaoka et al. [1] by blending two two-dimensional Gaussian functions: one for the back of the person and the other for the front. Therewith, they create an egg-shaped personal space. All in all, the robot is not only considering the personal space but interactions among people as well when navigating around them. The two-dimensional Gaussian gives rise to an elliptic rather than a circular F-Formation. However, interactions with objects are not part of their model.

Banisetty et al. [3] created objectives for the navigation in terms of costs and rewards. Compared to us, they do not manipulate the costmap utilized for navigation but the planning algorithm itself. We have introduced their notion of social goal above for joining

[1] http://wiki.ros.org/social_navigation_layers

a conversation and a queue. Additionally, the social goal can be utilized to bypass activity spaces arising from interactions with objects. In this case, the social goal functions as an intermediate destination before reaching the actual goal location. Yet, they did not infer this intermediate social goal but positioned it manually in their experiments. During the navigation, the following social costs are considered: interpersonal distance (i. e., personal space), distance from a group (i. e., interactions with humans), and the distance from the social goal. All costs and the reward for the social goal are modelled by exponential functions depending on the respective distance. Costs for interpersonal distance are only considered when the robot is 1.5m or closer to a person. The navigation problem is then solved by performing a multi-object optimization over all costs and rewards.

As noted earlier, it is a reasonable extension to include the time dimension when creating costs for a dynamically changing environment, including for instance moving people and changing activities. This was done by Kollmitz et al. [21] for the personal space of humans. They modelled the space by two Gaussian distributions and predicted the trajectory of the human into the future. The further into the future, the higher the variance and the less the amplitude of the Gaussian distributions in order to account for uncertainties in the prediction. This way, they create costmap layers for the current state and multiple future states. Moreover, their implementation in ROS is publicly available[2]. In real-world experiments, they demonstrated the advancements of their approach compared to non-time-dependent navigation. Another work by Scandolo and Fraichard [40] also takes a dynamic and hence time-dependent perspective on the navigation task and the involved costs. They model the personal space of a human by the following three parts which are then combined by the maximum operation: static personal space, back space, and personal space during motion. Moreover, they consider interactions whereby interactions can be with objects and humans. Hence, their model of social space is quite similar to ours. Nevertheless, they did not model these two types of interactions separately. The geometric shape of both kinds of interactions is the same and is based on the notion of convex hulls. The personal spaces and interaction regions are combined by summing them up.

Machine Learning and Human Feedback Instead of defining costs and their rewards manually, one can learn patterns and norms from social behaviour of humans and translate these into costs. This is even possible in an unsupervised manner. To take two examples, Johnson and Kuipers [15] demonstrate learning of social norms by detecting biases in human behaviour and Luber et al. [30] learn dynamic motion prototypes.

Moreover it is even possible to incooperate human feedback. For instance, Kollmitz et al. [22] do so by means of physical feedback and inverse reinforcement learning,

[2] https://github.com/marinaKollmitz/lattice_planner

while Karalus et al. [16] plan to utilize explanations given by humans. The latter involves methods from the field of explainable artificial intelligence and hence would allow the robot to explain its behaviour to the human as well.

2.3 Summary

We have reviewed the parts a person's social space is composed of and how existing human-aware navigation approaches model these. To our scenario, which is navigating to a given destination in a human-aware manner by not intruding people's social space, the following sub-spaces are relevant: person-related spaces and activity and affordance spaces. For the former, especially the personal space is important, which is based on the theory of interpersonal distances by Hall [12]. The latter entails interactions with objects and humans. For interactions with humans, a common theoretical model is the F-Formation by Kendon [8, 17]. We need to define a geometrical shape of the costs for all sub-spaces and a way of combining them. Personal space is most often modelled by means of Gaussian distributions, giving rise to circular, elliptic, egg-shaped, or asymmetric geometries. Interactions with humans (i. e., F-Formations) are circular or elliptic, whereas there is no clear shape definition for interactions with objects. For the latter, the reviewed works utilized circles, convex hulls, and field-of-view based shapes. Prominent functions for combining costs are sums, weighted sums, maximum, and context-dependent enabling. We also reviewed one work that utilized the product for combination.

Compared to most works, we have a more holistic view on the social space of humans. Even if they have a similar viewpoint on modelling the social space of humans, they either have not made their implementation publicly available (e. g., [40]) or modified the planning algorithm itself (e. g., [3]). Yet, we do not consider the time-dimension, which is required for better planning in dynamic environments where humans are moving and switching their activities. For now, we leave this subject to future work.

3 Methods

Before we explain the details of our modular cost function for modelling humans' social space and its employment as a ROS navigation layer [29, 36], we need to define what data is available. We assume the detection of humans and their interactions to be given. Nevertheless, it is important that this data basis is realistic such that an employment in a robot navigating in the real world is possible. If not otherwise stated, the coordinate system is in the plane of the floor, i. e., the x- and y-axis are aligned with the floor of the room and the z-axis would point towards the ceiling of the room.

A human contains information about their position in terms of x- and y-coordinates, their orientation θ_p given by an angle in $[-\pi, \pi]$, their velocity v_p in meters per second, and a confidence score c_p for the detection of the human in $[0, 1]$. An object is modelled as a line, i. e., an object has a start and an end x- and y-coordinate (this is similar to Bai [2]). A human can interact with any number of objects and groups of humans, where each group of humans can entail any amount of humans. Each interaction has a confidence value c_a in $[0, 1]$. This confidence value is not only related to the certainty of the detection but utilized for modelling the concept of activity and affordance space. For an interaction with a group of humans, we additionally make use of the detected centre of the F-Formation given by its x- and y-coordinates. Referring to our literature review in Section 2.2 and a survey by Möller et al. [31], the assumed data is realistic.

3.1 Modular Cost Function for Modelling Humans' Social Space

We model the social space of humans' in a modular manner as displayed in Figure 3.1. This involves two hierarchical steps: We create costs for the social space of each human in the scene individually. Furthermore, the social space of an individual themselves consists of the following sub-spaces: personal space, passing space, activity/affordance spaces for interactions with single objects, and activity/affordance spaces for interactions with humans. Due to the modular structure, one can easily add further sub-spaces or manipulate the existing ones in future work.

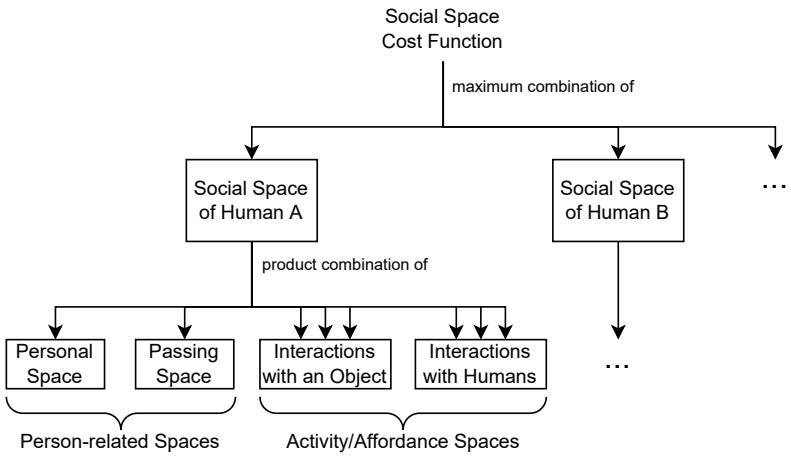


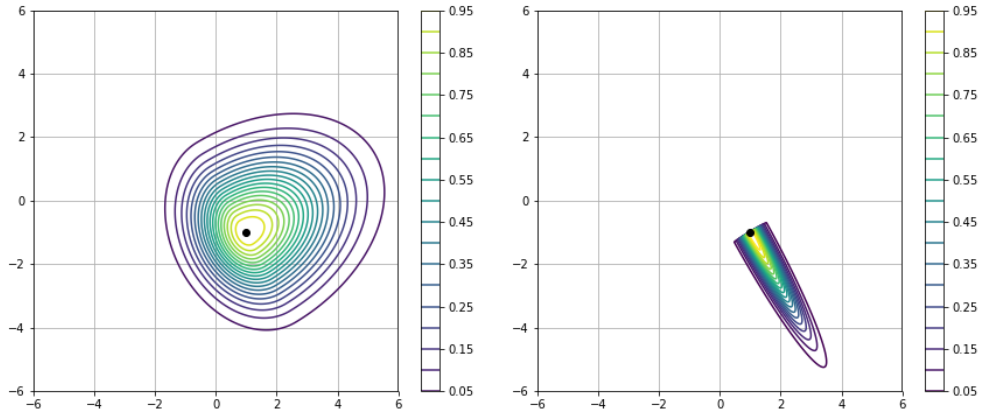
Figure 3.1: Depiction of the modular structure of our cost function for modelling humans' social space.

First, we present our modelling of the costs of the individual sub-spaces. Afterwards, we reason our combination of these costs in both hierarchy levels. The cost functions return values in $[0, 1]$, where 1 means maximal costs while 0 denotes no costs.

3.1.1 Person-related Spaces

We consider the personal space and the passing space as person-related spaces. The latter is not necessarily part of the social space of a human but used for modelling passing behaviour of humans when overtaking or coming somebody's way. These two sub-spaces are part of the baseline [29] we are comparing our approach to. In order to entail at least the parts of the baseline, we additionally model the passing space as a person-related space.

Since our literature review suggests that the personal space is not circular as defined by Hall [12] but rather enlarged to the front and depending on the velocity of the person, we make use of Kirby's [18] modelling approach. Similar to other works, they make use of a Gaussian for modelling the personal space, i. e. the costs decrease in an exponential manner. However, instead of combining different Gaussian functions, they introduce the notion of an asymmetric Gaussian. Note, the asymmetric Gaussian gives rise to an egg-shaped and not an asymmetric personal space. A depiction of the personal space modelled with this function is in Figure 3.2a. The direction of enlargement of the shape is into the direction of the orientation θ_p of the person. The amount of enlargement depends on the person's velocity. The area behind the person is narrowed. There, the person cannot see the robot and if the robot was following the person's trajectory, it



(a) Personal space. (b) Passing space for right-hand traffic.

Figure 3.2: Exemplary visualisation of the two person-related spaces for a person standing at $(1, -1)$ with orientation $\theta_p = \frac{\pi}{6}$ and velocity $v_p = 1\frac{\mathrm{m}}{\mathrm{s}}$. The confidence of the detection is 1. The x- and y-axis are given in meters.

would be pure chance as the robot is navigating independently from the person. When the person is not moving, they applied a symmetric Gaussian as a fallback since their detection is not able to infer a person's orientation in this case. We, however, assume the orientation of a person to be given independently from their velocity and hence apply the asymmetric Gaussian for non-moving people as well. The computation of the value of the asymmetric Gaussian at a point (x, y) according to Kirby [18] is detailed in Algorithm 1. The asymmetric Gaussian is described by six hyperparameters. The centre of the Gaussian is located at (x_c, y_c), it is orientated into the direction θ and σ_h, σ_s, and σ_r are the variances along the front, the sides and the rear respectively. Kirby [18] selected the following values for the hyperparameters in order to model the egg-shaped and dynamic variant of Hall's [12] personal space:

- $(x_c, y_c) = (x_p, y_p)$, i. e., the centre of the Gaussian is the position of the human.
- $\theta = \theta_p$, i. e., the orientation of the Gaussian is aligned with the person's orientation.
- $\sigma_h = \max\left\{2v_p, \frac{1}{2}\right\}$, i. e., the variance to the front increases with the velocity of the person.
- $\sigma_s = \frac{2}{3}\sigma_h$ and $\sigma_r = \frac{1}{2}\sigma_h$

We apply the same parametrization and additionally multiply the resulting value by the detection confidence c_p of the person. The size of the personal space depends on cultural, personal, and environment factors [12] and can be modified by adjusting only σ_h [18]. Due to the enlargement of the personal space for moving people, one can

Algorithm 1 Computation of the value of an asymmetric Gaussian at the coordinates (x, y) according to Kirby [18].
Source: based on Kirby [18, Algorithm A.1]

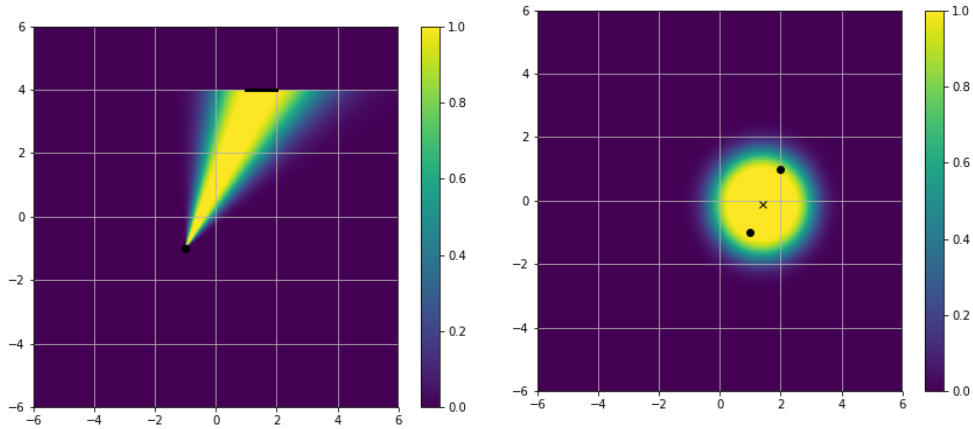
Require:
(x_c, y_c): the centre of the asymmetric Gaussian
θ: the orientation of the asymmetric Gaussian
θ_h: variance to the front, i. e., into the the θ direction
θ_s: variance to the sides, i. e., into the the $\theta \pm \frac{\pi}{2}$ directions
θ_r: variance to the rear, i. e., into the the $-\theta$ direction

1: **function** ASYMMETRICGAUSSIAN(x, y)
2: $\alpha = \text{arctan2}(y - y_c, x - x_c) - \theta + \frac{\pi}{2}$
3: normalize alpha to be in the range $[-\pi, \pi]$
4: $\sigma = \sigma_r$ if $\alpha \leq 0$ else σ_h
5: $a = \frac{(\cos\theta)^2}{2\sigma^2} + \frac{(\sin\theta)^2}{2\sigma_s^2}$
6: $b = \frac{\sin(2\theta)}{4\sigma^2} - \frac{\sin(2\theta)}{4\sigma_s^2}$
7: $c = \frac{(\sin\theta)^2}{2\sigma^2} + \frac{(\cos\theta)^2}{2\sigma_s^2}$
8: **return** $\exp\Big(-\big(a(x - x_c)^2\big) + 2b(x - x_c)(y - y_c) + \big(c(y - y_c)^2\big)\Big)$
9: **end function**

argue that the information processing space (IPS) is entailed within the personal space. Moreover, approaching pedestrians – and hence most likely moving robots as well – are the least attended obstacles in the IPS [19], which frames the notion of IPS in people's social space less relevant for a robot navigating to a given destination.

A model of passing space is given by Kirby [18] as well. Moreover, the utilized baseline actually applied Kirby's [18] computation of personal and passing space. The passing space is again an asymmetric Gaussian and is illustrated in Figure 3.2b for right-hand traffic. When the robot has to overtake or comes this person's way, it would pass on the person's left conforming to right-hand traffic rules. In their hallway passing experiments, Kirby [18] found that passing space costs are only applicable for moving people, i. e., when $v_p > 0$. Hence, we make use of context-dependent enabling for the costs of the passing space. The parameterization of the asymmetric Gaussian was designed by them to function well in most hallway environments and is as follows:

- $(x_c, y_c) = (x_p, y_p)$, i. e., the centre of the Gaussian is the position of the human.
- $\theta = \theta_p \mp \frac{\theta}{2}$ for right- and left-hand traffic respectively, i. e., the orientation of the Gaussian is to the side of the person's orientation depending on the traffic system.
- $\sigma_h = 2.0$, $\sigma_s = 0.25$, and $\sigma_r = 0.01$



(a) Interaction with an object. The human is standing at $(-1, -1)$ and interacting with an object starting at $(1, 4)$ and ending at $(2, 4)$.

(b) Interaction with a human. A Human is standing at $(1, -1)$ and interacting with another human positioned at $(2, 1)$. The centre of the F-Formation is located at $(1.4, -0.1)$.

Figure 3.3: Exemplary visualisation of the two kinds of activity spaces, i. e., the confidence is 1. The x- and y-axis are given in meters.

3.1.2 Activity and Affordance Spaces

The robot should not be within any activity spaces at any time since this will interrupt the activity of the human, but it may cross affordance spaces [27, 38]. Following Bai [2], we distinguish activity and affordance spaces by scaling the costs that would be created for a respective activity space by the confidence c_a of the interaction. This way, the confidence c_a is not just a tool for modelling the certainty of the prediction, but rather for declaring affordance spaces. Hence, $0 \leq c_a \leq 1$ forms a continuum from no interaction at all over affordance space to activity space. We apply different shapes for modelling an interaction with an object compared to an interaction with a group of people.

Interaction with an Object

We make use of the field-of-view paradigm for modelling the interaction of a person with an object [2, 27]. To us, circles and convex hulls seem more appropriate for interactions with humans, i. e., for modelling F-Formations.

Bai [2] implemented this paradigm by a circular sector whose radius is the distance between the human and the object. In order to give their shape vague bounds, they utilize exponential smoothing. To the sides, this smoothing is applied depending on the angular distance. To the edge of the circular segment, the smoothing is based on the distance to the object and additionally models the concept of blind space, i. e., the

Algorithm 2 Computation of the cost value at the coordinates (x, y) for a certain (i. e., $c_a = 1$) interaction with an object.

Require:
(x_p, y_p): the position of the human
$(x_{\text{start}}, y_{start})$: the start coordinates of the object
$(x_{\text{end}}, y_{end})$: the end coordinates of the object
σ_o: variance for angular smoothing

function TRIANGLE(x, y)
  $l_{\text{object}} \leftarrow$ line generated by the line segment of the object
  $l_{\text{human}} \leftarrow$ line that is parallel to the object and goes through (x_p, y_p)
  if (x, y) is not between l_{human} and l_{object} **then**
    return 0
  end if
  $\alpha = \text{arctan2}(y - y_p, x - x_p)$
  $\theta_{\text{start}} = \text{arctan2}(y_{start} - y_p, x_{start} - x_p)$
  $\theta_{\text{end}} = \text{arctan2}(y_{end} - y_p, x_{end} - x_p)$
  if the space from θ_{start} to θ_{end} is discontinuous **then**
    transform θ_{start}, θ_{end}, and α from $[-\pi, \pi]$ to $[0, 2\pi]$
  end if
  $\theta_{\text{small}} \leftarrow \min\{\theta_{\text{start}}, \theta_{\text{end}}\}$
  $\theta_{\text{large}} \leftarrow \max\{\theta_{\text{start}}, \theta_{\text{end}}\}$
  if $\theta_{\text{small}} \leq \alpha \leq \theta_{\text{large}}$ **then**
    return 1 ▷ (x, y) lies within the triangle
  end if
  if $\alpha > \theta_{\text{large}}$ **then**
    return $\exp\left(\frac{-(\alpha - \theta_{\text{large}})^2}{2\sigma_o^2}\right)$ ▷ angular smoothing
  end if
  if $\alpha < \theta_{\text{small}}$ **then**
    return $\exp\left(\frac{-(\alpha - \theta_{\text{small}})^2}{2\sigma_o^2}\right)$ ▷ angular smoothing
  end if
end function

space behind the object that cannot be seen by the human. Since their scenario is approaching a human, the robot should not aim at a position in the blind space.

Our implementation is close to their approach. However, we create a triangle as the activity/affordance space and only apply angular, exponential smoothing as visualized in Figure 3.3a. The corners of the triangle are the position of the human, the start of the object, and the end of the object. Since the object itself is treated as an obstacle by the navigation, there is no need to explicitly add the notion of blind space, i. e. there will be costs behind the object anyway. Furthermore, modelling blind spaces as described by Bai [2] that build a smooth and continual surface with the activity/affordance space is rather complicated for arbitrary positions and orientations of a person and an object

that is interacted with. Similarly, the triangle shape is more robust to arbitrary scenarios than the circular sector, especially in case of long objects and when the human is not centrally in front of the object.

Algorithm 2 details our implementation of the shape utilized for modelling the interaction with an object. The only hyperparameter to be set is the variance σ_o utilized for the angular smoothing. Based on Bai [2], we heuristically set the smoothing to $\frac{\pi}{36} \cdot \frac{w_{\text{robot}}}{3}$, where the latter term is added by us accounting proportionally for the width of the robot in decimetres. The width of their utilized Nao robot is ≈ 30cm, while the TIAGo robot has a width of 54cm[1]. Hence we set $\sigma_o = \frac{\pi}{36} \cdot \frac{5}{3}$ for the TIAGo robot.

Interaction with Humans

The underlying model for humans interacting with each other is the F-Formation by Kendon [8, 17]. Originally, the F-Formation is created by means of circles. In order to be more generic, we utilize ellipses instead. Another work [39] in our literature review did this as well. They used Gaussian functions for the elliptic shape. We, however, fit an ellipse to the people around the centre of the F-Formation utilizing the numerically stable direct least squares ellipse fitting algorithm by Halíř and Flusser [11][2]. In case the algorithm does not succeed, we fit a circle as a special case of an ellipse. Thereby, we do not distinguish between the o- and p-Space since people are represented just by a coordinate, i. e., a point without any extension. When one has a model of humans that takes into account their occupied space as well, our model can still be applied after determining the point of the human that is the furthest from the centre of the F-Formation. This way, the costs for the p-Space are the same as for the o-Space, which represents the fact that nobody should be within these two spaces unless they are part of the group. In the r-space, we apply radial, exponential smoothing with $\sigma_f = 0.5$ as the variance. The value for σ_f is hand-selected and should roughly reflect the expanse of personal space. Algorithm 3 gives details on the implementation steps, while Figure 3.3b exemplary illustrates the resulting shape for two people in an F-Formation.

[1] https://pal-robotics.com/wp-content/uploads/2021/10/Datasheet_TIAGo-Base-2021.pdf

[2] An implementation of this algorithm is available in the scikit-image [44] library.

Algorithm 3 Computation of the cost value at the coordinates (x, y) for a certain (i. e., $c_a = 1$) interaction with humans.

Require:
humans $= \{(x_p, y_p) : \forall$people in the group$\}$
(x_c, y_c): the centre of the F-Formation
σ_f: variance for radial smoothing

```
function ELLIPSE(x, y)
    humans_mirrored ← points in humans mirrored at (x_c, y_c)
    humans = humans ∪ humans_mirrored
                ..... ▷ ensures that (x_c, y_c) is the centre of the ellipse
    if fitting an ellipse to humans via least squares succeeds then
        a, b, θ ← values from fitting algorithm
    else ..... ▷ fit a circle as a special case of an ellipse as a fallback
        r = mean{‖(x_p − x_c, y_p − y_c)‖_2 : ∀(x_p, y_p) ∈ humans}
        a, b, θ = r, r, 0
    end if
    if (x, y) lies within the ellipse then
        return 1
    else ..... ▷ radial smoothing
        d ← distance between (x_c, y_c) and (x, y)
        l ← line through (x_c, y_c) and (x, y)
        (x_l, y_l) ← the crossing point of l with the boundary of the ellipse
        d_ellipse ← distance between (x_c, y_c) and (x_l, y_l)
        return exp(−(d − d_ellipse)^2 / (2σ_f^2))
    end if
end function
```

3.1.3 Combining the individual Sub-Spaces

Since we have introduced all functions for modelling the individual sub-spaces, the question remains how they are combined on the two hierarchy levels. In general, we do not calculate the costs of the sub-spaces for individual (x, y) points but for a complete grid via matrix and vector operations.

We combine the sub-spaces $s_1, s_2, \ldots, s_n$ of one person in the inverted cost-space via the product [2] operation

$$\text{prod}(s_1, s_2, \ldots, s_n) = 1 - \left(\prod_{i=1}^{n} (1 - s_i) \right) .$$

The resulting space $S_h = \text{prod}(s_1, s_2, \ldots, s_n)$ has increased costs at points where several cost functions are overlapping. When there is no overlap, the costs stay the same. This is useful for modelling the combination of sub-spaces to an individual's social

space since the robot should be "punished" more severely for intruding areas that cause discomfort in multiple sub-spaces. Furthermore it holds that

$$\text{prod}(s_1, s_2, \dots, s_n) = \text{prod}(\dots(\text{prod}(\text{prod}(s_1, s_2), s_3), \dots), s_n) \ ,$$

which is a property we make use of in the implementation.

The social spaces of several people are combined via the commonly known maximum operator $\max$ [38]. Compared to the product operation, costs cannot be increased by applying the maximum operator. One might argue that the robot should be "punished" greater when several people's social space is getting intruded. However, there is an additional trade-off: With the maximum operator we allow to rather intrude multiple people's social space minimally instead of severely intruding the social space of a single person. Furthermore, one does not need to take care to apply costs for F-Formations only once and not for several people that are part of the same formation. Again, we can split the computation of the operator into several binary evaluations without changing the result.

3.2 Social Space Navigation Layer

We employ the previously defined cost function for the social space of humans as a costmap layer [29] for navigation within the ROS [36] framework. The paradigm of layered costmaps, introduced by Lu et al. [29], allows to stack multiple costmap layers. These layers are then hierarchically combined into a master layer, which is the costmap actually utilized for navigating the robot. We are referring to our costmap layer as *social space layer*.

Each layer is required to implement the following two functionalities. First, the layer needs to communicate which rectangular area of the costmap it wants to update. Our social space layer does this by determining the minimum and maximum x- and y-coordinates of all detected humans and objects interacted with. Additionally, we take a square area around the robot into account in order to make sure that costs close to robot are not cut off. At every update cycle of the costmap, we remove the previously set costs in the social space layer. Hence, the earlier communicated rectangular area needs to be considered in this calculation as well, otherwise the costs would not be removed. Since this request for communicating the area of the costmap to be updated is issued layer-wise, every layer needs to consider at least the rectangular area of the previous layer. Secondly, the layer is queried to set its costs in the master grid in the rectangular area that was determined in the previous step over all layers. We obtain the costs in the area to be updated from our social space cost function. The values of

the retrieved costs are in $[0, 1]$. The cost-space of the costmap[3], however, is in $[0, 254]$. This is why we need to apply a scaling factor transforming the costs into an appropriate range for the costmap. Since the setting of the costs is again a layer-wise operation, one needs to define how the costs of one's own layer are accumulated with the costs already entered in the master costmap by the preceding costmap layer. We do so by applying the maximum operation.

Additionally, each costmap layer has hyperparameters. For our social space layer, we made the following parameters configurable:

- enabled (default: true): Whether the social space layer is applied.
- cutoff (default[4]: 10.0): The smallest cost to be set on the costmap.
- amplitude (default[5]: 77.0): Scaling factor for the costs returned by the cost function, i. e., determines the amplitude of the costs at their peak.
- robot_bounds (default: 2.0): The square area around the robot to be considered when communicating the area of the costmap to be updated $\big((x_r, y_r) \pm$ robot_bounds$\big)$.
- personal_sigma_h_factor (default: 1.0): A scaling factor applied to σ_h when calculating the costs of the personal space.
- passing_right_hand_traffic (default: true): Whether the passing space is for right- or left-hand traffic.
- passing_sigma_h (default: 2.0): The σ_h hyperparameter of the passing space.
- passing_sigma_s (default: 0.25): The σ_s hyperparameter of the passing space.
- passing_sigma_r (default: 0.01): The σ_r hyperparameter of the passing space.
- activity_object_sigma (default: 0.15): The σ_o hyperparameter of the activity/affordance space for interactions with an object.
- activity_human_sigma (default: 0.5): The σ_f hyperparameter of the activity/affordance space for interactions with humans.

[3] http://wiki.ros.org/costmap_2d#Inflation
[4] This is the default value applied by the existing social navigation layers: http://wiki.ros.org/social_navigation_layers
[5] This is the default value applied by the existing social navigation layers: http://wiki.ros.org/social_navigation_layers

3.3 Implementation Details

An implementation of the modular cost function of humans' social space is available in Python2.7 along with a jupyter notebook that functions as a playground. We had to choose Python2.7 since the TIAGo [33, 34] robot utilized in the evaluation is not compatible with a more recent version of Python. Nonetheless, the implementation of the cost function is functional with a more recent Python version (e. g., Python3.7) as well.

The social space navigation layer is implemented in C++ as there are only tutorials available in this programming language. The layer is subscribed to messages containing the data about detected humans and their interactions. Before performing any calculations, all detected humans and objects need to be tranformed into the coordinate frame of the costmap. Since the social space layer is implemented in C++, while the cost function for humans' social space is written in Python, we make use of the ROS client-service paradigm when obtaining the costs. The above implementation of the cost function is wrapped into a ROS Python service that is callable by our C++ implementation of the social space layer. This separation is a further modularity that allows modifications of the cost function to take effect in the costmap layer without changing any code of the layer itself. Last but not least, the parameters of the social space layer are dynamically configurable via the rqt_reconfigure[6] GUI.

[6] http://wiki.ros.org/rqt_reconfigure

4 Evaluation

We are qualitatively evaluating the modular cost function for modelling humans' social space by applying the respective social space navigation layer to the navigation stack of the TIAGo [33, 34] robot. The experiments are executed in the simulation framework Gazebo [20] and require the robot to navigate from an initial start position to a given destination. Thereby, we consider different scenarios which are specified in Section 4.1. The measures we apply for the qualitative evaluation are listed in Section 4.2.

As mentioned earlier, we do not perform the detection of humans and their interactions but assume it to be given. As such, we take a global perspective where detections are independent of the field of view of the robot. This means, all humans and interactions in the scenario are detected and forwarded at any point in time. To this end, we implemented mocked publishers sending messages with the respective data.

We add the social space navigation layer to the existing navigation stack of the TIAGo robot for both the global and the local costmap. The global costmap is utilized by the global planner creating a trajectory from the start to the goal, while the local costmap serves as an input to the local planner adapting the path calculated by the global planner. The local costmap is updated more frequently compared to the global one. We leave the navigation stack of the TIAGo robot, including all planners, costmap layers, and hyperparameters, as is; we only add the social space layer as the lowest layer in costmap layer hierarchy. Where not otherwise stated, the parameters of the social space layer are the default ones presented in the previous chapter. For comparison, we build a baseline by exchanging our social space layer with the already existing social navigation layers[1] [29] in ROS. Again, where not otherwise stated, the default values of the social layers are applied. The two social navigation layers implement the personal and the passing space as defined by Kirby [18], i. e., the baseline should in theory roughly reflect the behaviour of our layer when no interactions with objects and humans are modelled. Differently to Kirby [18] and us, they combine the costs of the personal and passing space via the maximum operation. For this purpose, Kirby [18] originally made use of the sum operator, while we are combining the personal and passing space of individuals with the product operation and use the maximum operator to combine costs of several individuals. Since our scenarios are static, i. e., they do not contain any moving humans,

[1] http://wiki.ros.org/social_navigation_layers

there will be no passing space costs anyway. Due to this, the difference in combination between our approach and the baseline is not observable in the experiments.

4.1 Scenarios

We experiment with three different static scenarios, i. e., for all humans $v_p = 0$. The first and third one are aimed at verifying the modelling of interactions with objects and humans, respectively. The second scenario investigates the difference between activity and affordance spaces when navigating. All scenarios are based on the simple_office world in the Gazebo [20] simulation. We orient the robot in steps of $45°$ such that it can walk approximately straight to the goal destination without needing to rotate.

The possible variants a navigation can be performed in our scenarios, i. e., how the social navigation layers and our social space navigation layer are employed, are listed below. The naming pattern follows <employed layer>−<amplitude>−<specialities>.

- baseline-77: We employ the social navigation layers with their default values.
- socialSpace-77-noInteraction: We employ the social space navigation layer with its default values but do not model the interaction of the human with the object. This way, our navigation layer should be performing equivalent to the baseline.
- socialSpace-77: We employ the social space navigation layer with its default values.
- socialSpace-154: We double the amplitude of the social space navigation layer.
- socialSpace-231: We triple the amplitude of the social space navigation layer.
- socialSpace-253: We increase the amplitude of the employed social space navigation layer to the highest value possible before the costs are considered as a collision area.
- socialSpace-253-local: The social space navigation layer is only employed in the local costmap. Other than that, the configuration of the social space navigation layer is the same as in the socialSpace-253 variant.

4.1.1 Scenario1: Interaction with an Object

In this scenario a human interacts with an object. The human is positioned at $(2.0, 1.5)$ with the orientation $\theta_p = \frac{17}{18}\pi$. This way, the human is oriented towards the object they interact with. From the viewpoint of the human, the line representing the object is starting at $(-3.5, 2.0)$ and ending at $(-3.0, 3.0)$. The confidence for detecting the human

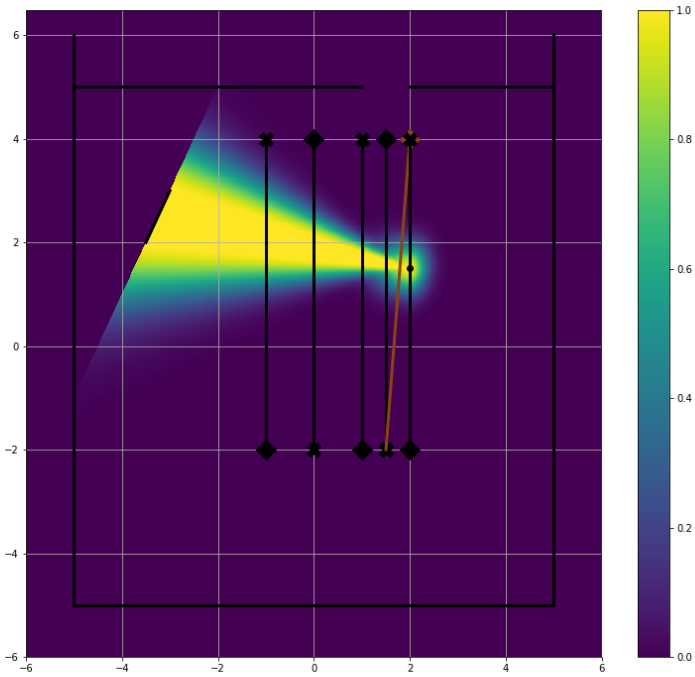


Figure 4.1: Visualisation of Scenario1, where a human interacts with an object. The marked costs are from our social space cost function. The shortest path between the respective start (♦) and goal (**X**) positions the robot has to navigate to is indicated by a line.

and the interaction is 1. See Figure 4.1 for a depiction with the costs returned by our social space cost function. The figure additionally contains the pairs of start (♦) and goal (**X**) positions we let the robot navigate to. The lines between the start and goal points group together the pairs and represent the shortest path.

Since the start and goal positions are on the 0.5 grid, they can be easily read off the figure. We perform the navigation in the following variants: baseline-77, socialSpace-77-noInteraction, socialSpace-77, socialSpace-154, socialSpace-231, socialSpace-253, and socialSpace-253-local.

4.1.2 Scenario2: Activity vs Affordance Space

This scenario investigates the preference of the robot to navigate through an affordance space rather than an activity space. A visualisation of this scenario along with the

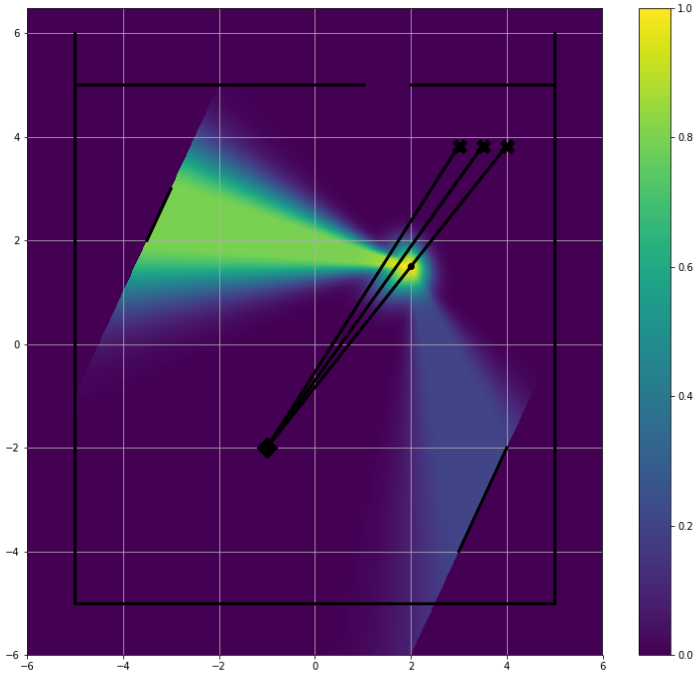


Figure 4.2: Visualisation of Scenario2, where a human interacts with two different objects. One interaction has a confidence of 0.8, while the other has a confidence of 0.2. The marked costs are from our social space cost function. The shortest path between the respective start (♦) and goal (**X**) positions the robot has to navigate to is indicated by a line.

costs by our social space cost function is in Figure 4.2. There are two objects in the scene: Object1 starts at $(-3.5, 2.0)$ and ends at $(-3.0, 3.0)$, while object2 starts at $(3.0, -4.0)$ and ends at $(4.0, -2.0)$. The human stands at $(2.0, 1.5)$ with the rotation $\theta_p = -\frac{38}{45}\pi$. Based on the orientation of the human, the confidence of interacting with object1 is 0.8, while the confidence of interacting with object2 is 0.2. Therefore, interrupting the interaction between the person and object1 is more severe than doing so for the interaction with object2. For easier differentiation, we term the interaction with object1 an “activity” and the interaction with object2 an affordance.

The robot always starts at $(-1.0, -2.0)$ and navigates to the following three destinations: $(4.0, 3.83)$, $(3.5, 3.83)$, $(3.0, 3.83)$. For the first destination, the shortest path would go straight through the human. For the other destinations the shortest path is

through the “activity” and not the affordance space. We perform the navigation in the socialSpace-77, socialSpace-154, socialSpace-231, and socialSpace-253 variant.

4.1.3 Scenario3: Interaction between two Humans

Figure 4.3 shows the setting for this evaluation scenario, where two humans are interacting with each other. One human is positioned at $(2.5, 2.0)$ with orientation $\theta_p = -\frac{3}{4}\pi$, while the other human is at $(0.5, -0.5)$ with $\theta_p = \frac{1}{4}\pi$. Hence, the humans are facing each other and we assume that the centre of the F-Formation was detected at $(1.4, 0.8)$. Furthermore, the confidence for the interaction is 1.

The robot has to perform two different navigation tasks. The first is navigating from $(1.5, -4.0)$ to $(1.5, 4.0)$, i. e., the robot would navigate vertically in the middle of the two humans when selecting the shortest trajectory. In the second task, the robot

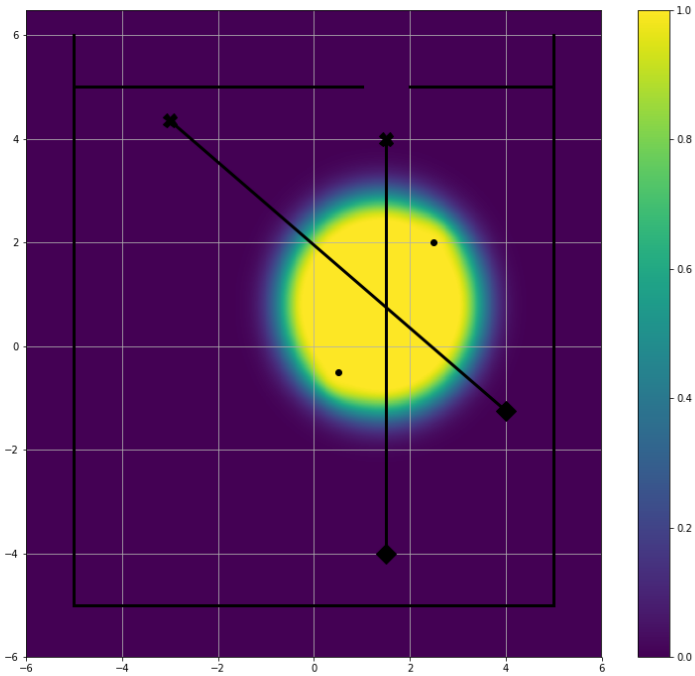


Figure 4.3: Visualisation of Scenario3, where two humans are interacting with each other. The marked costs are from our social space cost function. The shortest path between the respective start (♦) and goal (**X**) positions the robot has to navigate to is indicated by a line.

has to navigate from $(4.0, -1.25)$ to $(-3.0, 4.35)$. The line created by this start and goal position crosses the centre of the detected F-Formation and is perpendicular to the line between the two humans. The variants we perform the navigation in are: baseline-77, socialSpace-77, socialSpace-154, socialSpace-231, socialSpace-253, and socialSpace-253-local.

4.2 Measures

We performed a literature review regarding metrics applied for evaluating human-aware robot navigation scenarios without performing a user study. Based on Pimentel and Aquino-Jr [35] and especially Gao and Huang [9], we investigate the following measures for our qualitative evaluation.

The navigation performance is evaluated in terms of success and path efficiency. The former merely addresses whether the robot was able to navigate from the start to the goal position. The latter qualitatively reasons about the deviation of the executed navigation trajectory from the shortest path. For behavioural naturalness, we check the smoothness of trajectory, i. e., whether the robot rapidly changes its movement direction while navigating. We evaluate human comfort by investigating whether any collisions or intrusions of social spaces occurred. For the latter, we examine to what extend the robot respects the costs of our social space navigation layer. Hence, this is the main evaluation measure for verifying the validity of our cost function utilized for human-aware robot navigation to a given destination. Since humans are additionally treated as obstacles, there should be no collisions independent of whether the social or social space costmap layer is enabled.

5 Results

The robot was always able to reach the goal and did not collide with any humans or objects. The latter was expectable since collision avoidance is implemented independently of the social or social space costmap layer. In the following we present the results per scenario evaluating path efficiency, intrusion of social space, and trajectory smoothness.

5.1 Scenario1: Interaction with an Object

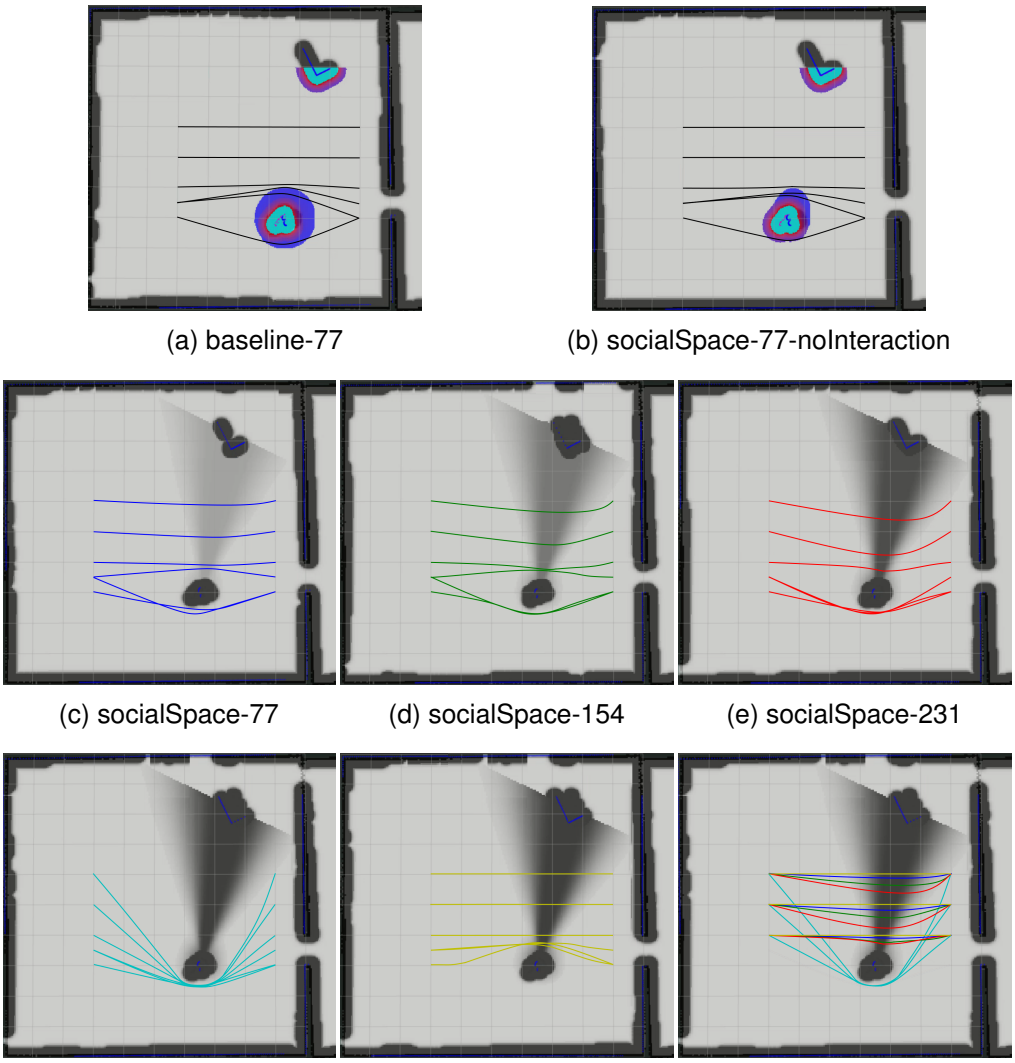
The resulting navigation trajectories for all executed variants are presented in Figure 4.1. All trajectories look smooth, i. e., the robot does not suddenly change its direction of movement.

In theory, the baseline-77 (Figure 5.1a) and the socialSpace-77-noInteraction (Figure 5.1b) variant should lead to similar results. We can see that in both cases the interaction with the object is not considered and hence the trajectories pass the activity space. The personal space around the people is modelled differently, leading to different trajectories when the robot is close to the human. The baseline strictly applies Kirby's [18] approach making use of the symmetric Gaussian when the person is not moving. Our social space navigation layer assumes the orientation of non-moving people to be given as well, allowing us to apply the asymmetric Gaussian. This follows the suggestion by Kirby [18]: When the orientation of people is known utilize the asymmetric Gaussian, otherwise fall back to a symmetric Gaussian. The resulting trajectories show that baseline-77 leaves more space between the human and the robot.

Comparing the socialSpace-77 (Figure 5.1c), socialSpace-154 (Figure 5.1d), socialSpace-231 (Figure 5.1e), and socialSpace-253 (Figure 5.1f) variants, we can see that the robot deviates from the shortest path more, the higher the costs for passing the activity space are (see Figure 5.1h for a combined plot). Only socialSpace-253 does not pass the activity space. Hence the social space of the person is not intruded, while the trajectories are inefficient in terms of path lengths. The other variants minimize the costs of intrusion while remaining closer to the shortest path.

The socialSpace-253-local (Figure 5.1g) variant does again pass the activity space. Its resulting trajectories are closest to the socialSpace-77-noInteraction variant although the socialSpace-253 variant did completely omit the activity space. The poor performance of socialSpace-253-local in terms of intruding social space can be explained by the

degree of deviation the local planner is allowed to perform compared to the global plan. Since the global planner is not aware of any personal and activity space costs, it simply plans the shortest trajectory treating the human as an obstacle that needs to be avoided.

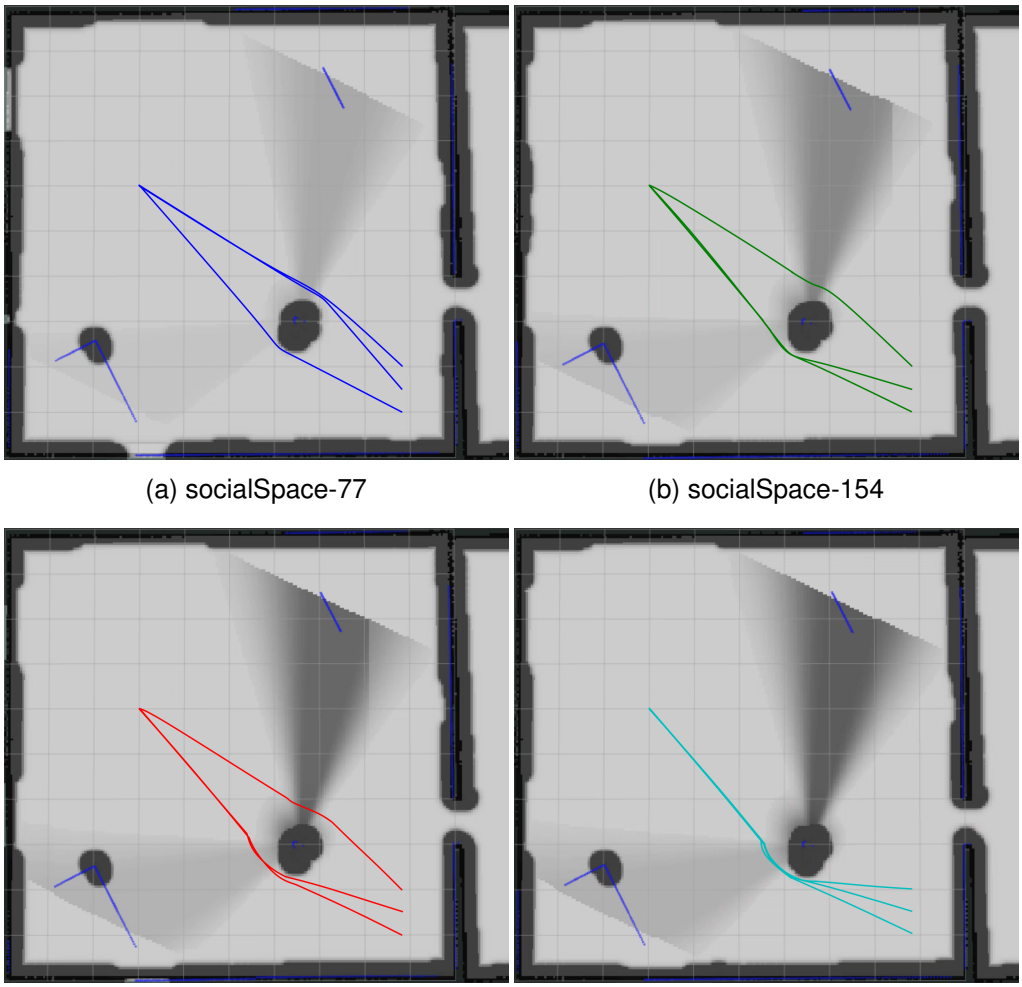


(a) baseline-77 (b) socialSpace-77-noInteraction

(c) socialSpace-77 (d) socialSpace-154 (e) socialSpace-231

(f) socialSpace-253 (g) socialSpace-253-local (h) combined plot (colours as before)

Figure 5.1: Resulting navigation trajectories for Scenario1.



(a) socialSpace-77 (b) socialSpace-154

(c) socialSpace-231 (d) socialSpace-253

Figure 5.2: Resulting navigation trajectories for Scenario2.

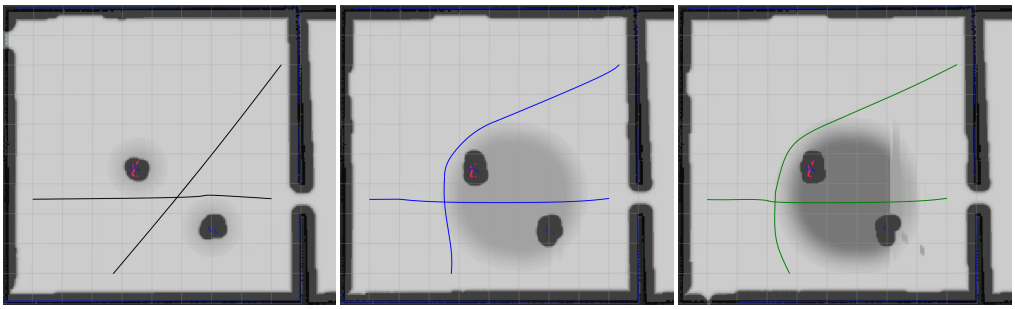
5.2 Scenario2: Activity vs Affordance Space

The resulting trajectories for Scenario2 in Figure 5.2 show that the robot prefers passing the affordance space when the detour is not too long compared to the costs of social space. The higher the costs for the social space, the more likely it is that the robot will take the detour through the affordance space instead of passing the activity space. In terms of trajectory smoothness, we can see that the paths in the socialSpace-77 (Figure 5.2a) and socialSpace-154 (Figure 5.2b) variants seem smooth, whereas there is a clear kink for the socialSpace-231 (Figure 5.2c) and socialSpace-253 (Figure 5.2d) variants before entering the affordance/activity space. This kink can be explained in the

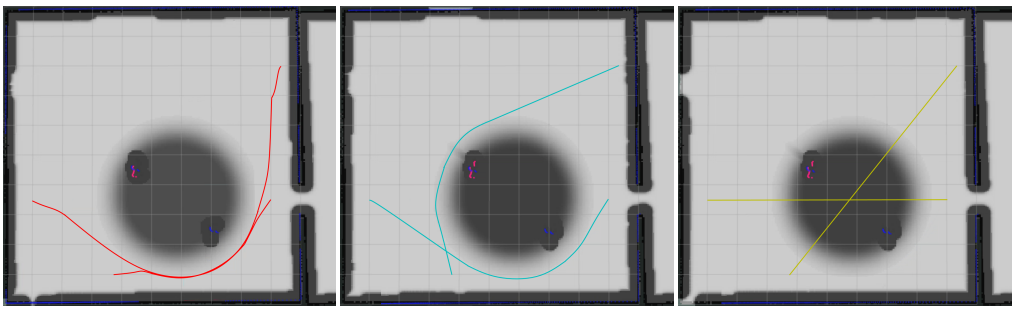
following way: The robot wants to cross the space as close as possible to the human and as orthogonally as possible. This way the triangular cost area has the least width and the path through the social space costs is the shortest.

5.3 Scenario3: Interaction between two Humans

Figure 5.3 shows the navigation trajectories for Scenario3. In the baseline (Figure 5.3a), the robot is not aware of the interaction between the humans. Hence, it takes the shortest paths with a minimal deviation accounting for personal space. The results for the socialSpace-77 (Figure 5.3b) and socialSpace-154 (Figure 5.3c) are quite similar: In one case the activity space is bypassed while it is crossed in the other. For both the socialSpace-231 (Figure 5.3d) and socialSpace-253 (Figure 5.3e) variant, the activity space between the humans is respected by taking a detour. Again, the socialSpace-253-local (Figure 5.3f) variant does not account for the social space. The explanation is the same as for Scenario1. Overall, the trajectories of the robot can be regarded as smooth for this scenario.



(a) baseline-77 (b) socialSpace-77 (c) socialSpace-154



(d) socialSpace-231 (e) socialSpace-253 (f) socialSpace-253-local

Figure 5.3: Resulting navigation trajectories for Scenario3.

6 Discussion

6.1 Social Space Navigation Layer

From the results one can see that path efficiency and not intruding people's social space are most often contradicting goals since the robot has to take a detour in order to bypass social spaces. Whether the detour is taken, depends on the ratio between the costs arising due to passing social spaces and the efforts for moving. In our scenarios, we exemplary demonstrated this trade-off by applying different amplitudes for the costs in the social space navigation layer. However, there are many more hyper-parameters in the navigation layer itself. By increasing the personal_sigma_h_factor, for instance, one can increase the area occupied by personal space costs and hence make the robot pass the human keeping more distance. Moreover, one can manipulate the other navigation layers in the navigation stack. Additionally, the global and local navigation planner have several hyperparameters and the default planners can even be exchanged by other kinds of planners. Through manipulating the planners, one might be able to improve the social behaviour of the robot when the social space costs are only entered into the local costmap. For the baseline, the additional social costs are in the surrounding of the people and hence require the robot to only dodge these costs by making a slight curve. Since we model activity and affordance spaces in addition, avoiding costs can require a completely different navigation trajectory and not only a sidestep. This circumstance needs to be considered when applying our social space navigation layer. In conclusion, the behaviour of the robot depends on many factors, such as choices of planners and navigation layers and the tuning of the hyperparameters. Therefore, it is possible to achieve different behaviours than demonstrated in this report by tuning [28, 47] the complete navigation stack for one's specific application.

As shown in the previous chapter, our social space navigation layer can improve the behaviour of the robot in terms of respecting people's social space. The robot does not always bypass the social spaces completely. However, the behaviour in Scenario1 seems reasonable: Instead of taking a large detour of several meters, the robot tries to pass the activity space closer to the human. Thus, it occupies the person's field-of-view less and hence interrupts the interaction for a shorter time compared to taking the shortest path. When the intrusion of the activity space is communicated by the robot [5], e. g., by announcing it or making eye contact with the human and getting

their implicit approval, such behaviour may be regarded as socially acceptable. The non-smooth trajectory taken by the robot in Scenario2, however, might be perceived as non-human like and confuse people in the surrounding of the robot since they cannot easily understand its behaviour. Though, this behaviour could again be changed by tuning the navigation stack punishing non-smooth trajectories more.

6.2 Social Space Cost Function

For our social space cost function, we decided to combine the costs at two hierarchy levels applying the product operation for a person's individual sub-spaces and the maximum operator for the social spaces of different people. Especially when combining the social spaces of several people, the question arises how to deal with the intrusion of multiple spaces. Lindner [25] tested four different ways of accumulating personal space intrusions, where two ways considered the amount of people while the other two did not. They found that people prefer an aggregation of the costs by the sum operation, i. e., the more people's personal spaces are intruded, the higher the costs. Nevertheless, some participants were open to the philosophical argument that the amount of people is irrelevant since every person should have the same chance of their personal space not getting intruded. With the maximum operator, we follow the latter reasoning. Furthermore, we argue that the extend of intrusion plays a role as well. With the maximum operator we rather intrude multiple people's social space slightly instead of intruding a single person's social space more extremely. An exemplary scenario making this concept more understandable is when two people are standing next to each other in a hallway but are not interacting with each other. We assume that the desired behaviour is to pass the people in the middle instead of sparing one personal space while intruding the other.

During our literature review, we already mentioned that the robot itself has a personal space. Hence, the robot should not only try to stay out of other people's social space but keep its own personal space clear from any humans. Since the personal space of the moving robot is different to the personal space of the humans in the scene, the additional modelling of the robot's personal space would lead to a change in its path leaving more space around humans when passing them. Kirby [18] included these costs for the robot's personal space into their planning algorithm. In a costmap, we cannot simply enter the personal space of the robot around it. Instead, one would need to manipulate the costs of the people's personal space such that the robot's personal space is entailed in them. As we aimed at creating a cost function for people's social space that is independent of the robot, we left this modification to future work.

Moreover, interactions with objects and humans are not the only activities performed by people. Modelling activities such as playing football or dancing are potentially difficult

with the current definition of the social space cost function. That is because these activities might occupy more space than only the one between the human and the objects or humans, e. g., football, goal, music box, or other dancers, interacted with. The costs for these activities can be modelled separately with fitting shapes. Due to the modular definition of our cost function, these can be easily added to our social space model as another kind of activity/affordance space.

6.3 Other Aspects

Our executed experiments in the simulation were not dynamic and assumed a perfect detection of humans and their interactions without any dependence on the field-of-view of the robot. Hence, we suggest to implement the detection in ROS and evaluate the navigation trajectories on a real robot with a user study in future work. When facing the robot with dynamic scenarios, time-dependent costmaps and forecasting of humans' movement and activities comes into play in order to further improve the social behaviour of the robot.

An additional improvement is to check destinations for their adequateness, i. e., whether the final position of the robot is socially acceptable. If they are not, one should manipulate them as done by Banisetty et al. [3] for instance. This cannot be implemented in a costmap, but as a preliminary – or in case of dynamic scenarios, continual – check that utilizes the social space cost function.

7 Conclusion

In the herein presented work, we created a modular cost function for modelling humans' social space and employed it as a navigation layer within the ROS framework in order to improve the social behaviour of a robot navigating to a given destination. Thereby, we did not only consider the personal and passing space as person-related spaces, but included interactions with objects and humans as activity and affordance spaces. We modelled the person-related spaces with an asymmetric Gaussian [18] based on Hall's [12] theory of interpersonal distances. Interactions with an object are modelled by a triangular shape following the field-of-view paradigm, while interactions with humans form an F-Formation [8, 17] that we represent by an ellipse. The sub-spaces are combined by the product operation, whereas social spaces of different persons are aggregated via the maximum operation. We set up three different scenarios in the simulation framework Gazebo and qualitatively assessed the resulting navigation trajectories of the TIAGo robot. The evaluation verifies that our social space navigation layer can improve the social behaviour of the robot when application-specific tuning is applied to the navigation stack [28, 47]. Since our approach does only add a further costmap layer to the ROS navigation stack, our social space navigation layer can be easily added to the navigation of existing robots utilizing ROS. This way, our manually created model of humans' social space can serve as a baseline for future research.

Bibliography

[1] Toshitaka Amaoka, Hamid Laga, Suguru Saito, and Masayuki Nakajima. 2009. Personal space modeling for human-computer interaction. In *Entertainment Computing – ICEC 2009*, pages 60–72, Berlin, Heidelberg. Springer Berlin Heidelberg.

[2] L. Bai. 2018. Merging human-object interaction behavior into a personal space model: for social robot navigation. Master's thesis, Industrial Engineering and Innovation Sciences, Eindhoven University of Technology.

[3] Santosh Balajee Banisetty, Scott Forer, Logan Yliniemi, Monica Nicolescu, and David Feil-Seifer. 2021. Socially aware navigation: A non-linear multi-objective optimization approach. 11(2).

[4] Hrishav Bakul Barua, Pradip Pramanick, Chayan Sarkar, and Theint Haythi Mg. 2020. Let me join you! Real-time f-formation recognition by a socially aware robot. In *29th IEEE International Conference on Robot and Human Interactive Communication, RO-MAN 2020, Naples, Italy, August 31 - September 4, 2020*, pages 371–377. IEEE.

[5] Timo Baumann and Felix Lindner. 2015. Incremental speech production for polite and natural personal-space intrusion. In *Social Robotics*, pages 72–82, Cham. Springer International Publishing.

[6] Lilli Bruckschen, Kira Bungert, Nils Dengler, and Maren Bennewitz. 2020. Human-aware robot navigation by long-term movement prediction. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 11032–11037.

[7] Shu-Yun Chung and Han-Pang Huang. 2012. Incremental learning of human social behaviors with feature-based spatial effects. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2417–2422.

[8] T. Matthew Ciolek and Adam Kendon. 1980. Environment and the spatial arrangement of conversational encounters. *Sociological Inquiry*, 50(3-4):237–271.

[9] Yuxiang Gao and Chien-Ming Huang. 2022. Evaluation of socially-aware robot navigation. *Frontiers in Robotics and AI*, 8.

[10] James Jerome Gibson. 1979. The theory of affordances. In *The ecological approach to visual perception*, chapter 8. Houghton Miffin, Boston.

[11] Radim Halíř and Jan Flusser. 1998. Numerically stable direct least squares fitting of ellipses. In *Proc. 6th International Conference in Central Europe on Computer Graphics and Visualization. WSCG*, volume 98, pages 125–132. Citeseer.

[12] Edward Twitchell Hall. 1969. *The Hidden Dimension*. Doubleday, New York.

[13] Dirk Helbing and Péter Molnár. 1995. Social force model for pedestrian dynamics. *Phys. Rev. E*, 51:4282–4286.

[14] Peter Henry, Christian Vollmer, Brian Ferris, and Dieter Fox. 2010. Learning to navigate through crowded environments. In *2010 IEEE International Conference on Robotics and Automation*, pages 981–986.

[15] Collin Johnson and Benjamin Kuipers. 2018. Socially-aware navigation using topological maps and social norm learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '18, page 151–157, New York, NY, USA. Association for Computing Machinery.

[16] Jakob Karalus, Amar Halilovic, and Felix Lindner. 2021. Explanations in, explanations out: human-in-the-loop social navigation learning. In *ICDL Workshop on Human aligned Reinforcement Learning for Autonomous Agents and Robots*.

[17] Adam Kendon. 2010. *Spacing and Orientation in Co-present Interaction*, pages 1–15. Springer Berlin Heidelberg, Berlin, Heidelberg.

[18] Rachel Kirby. 2010. *Social Robot Navigation*. Ph.D. thesis, Carnegie Mellon University, Pittsburgh, PA.

[19] Kay Kitazawa and Taku Fujiyama. 2010. Pedestrian vision and collision avoidance behavior: Investigation of the information process space of pedestrians using an eye tracker. In *Pedestrian and Evacuation Dynamics 2008*, pages 95–108. Springer.

[20] N. Koenig and A. Howard. 2004. Design and use paradigms for gazebo, an open-source multi-robot simulator. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, volume 3, pages 2149–2154.

[21] Marina Kollmitz, Kaijen Hsiao, Johannes Gaa, and Wolfram Burgard. 2015. Time dependent planning on a layered social cost map for human-aware robot navigation. In *2015 European Conference on Mobile Robots (ECMR)*, pages 1–6.

[22] Marina Kollmitz, Torsten Koller, Joschka Boedecker, and Wolfram Burgard. 2020. Learning human-aware robot navigation from physical interaction via inverse reinforcement learning. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 11025–11031.

[23] Thibault Kruse, Amit Kumar Pandey, Rachid Alami, and Alexandra Kirsch. 2013. Human-aware robot navigation: A survey. *Robotics and Autonomous Systems*, 61(12):1726–1743.

[24] In Lee. 2021. Service robots: A systematic literature review. *Electronics*, 10(21).

[25] Felix Lindner. 2016. How to count multiple personal-space intrusions in social robot navigation. In *What Social Robots Can and Should Do - Proceedings of Robophilosophy 2016 / TRANSOR 2016, Aarhus, Denmark, October 17-21, 2016*, volume 290 of *Frontiers in Artificial Intelligence and Applications*, pages 323–331. IOS Press.

[26] Felix Lindner. 2021. Soziale Robotik und KI. In Oliver Bendel, editor, *Soziale Roboter*, chapter 6, pages 109–123. Springer Fachmedien Wiesbaden.

[27] Felix Lindner and Carola Eschenbach. 2011. Towards a formalization of social spaces for socially aware robots. In *Spatial Information Theory*, pages 283–303, Berlin, Heidelberg. Springer Berlin Heidelberg.

[28] David V. Lu, Daniel B. Allan, and William D. Smart. 2013. Tuning cost functions for social navigation. In *Social Robotics*, pages 442–451, Cham. Springer International Publishing.

[29] David V. Lu, Dave Hershberger, and William D. Smart. 2014. Layered costmaps for context-sensitive navigation. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 709–715.

[30] Matthias Luber, Luciano Spinello, Jens Silva, and Kai Oliver Arras. 2012. Socially-aware robot navigation: A learning approach. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2012, Vilamoura, Algarve, Portugal, October 7-12, 2012*, pages 902–907. IEEE.

[31] Ronja Möller, Antonino Furnari, Sebastiano Battiato, Aki Härmä, and Giovanni Maria Farinella. 2021. A survey on human-aware robot navigation. *CoRR*, abs/2106.11650.

[32] Frank Ostermann and Sabine Timpf. 2007. Modelling space appropriation in public parks. In *Proceedings of 10th AGILE International Conference on Geographic Information Science*. Aalborg University.

[33] Jordi Pages, Luca Marchionni, and Francesco Ferro. 2016. Tiago: the modular robot that adapts to different research needs.

[34] Jordi Pagès, Luca Marchionni, and Francesco Ferro. 2018. Learning advanced robotics with tiago and its ROS online tutorials. In *Proceedings of the Workshop on Teaching Robotics with ROS (held at ERF 2018) co-located with European Robotics Forum 2018 (ERF 2018), Tampere, Finland, March 15th, 2018*, volume 2329 of *CEUR Workshop Proceedings*, pages 30–40. CEUR-WS.org.

[35] Fagner de Assis Moura Pimentel and Plinio Thomaz Aquino-Jr. 2021. Evaluation of ROS navigation stack for social navigation in simulated environments. *Journal of Intelligent & Robotic Systems*, 102(4).

[36] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, Andrew Y Ng, et al. 2009. ROS: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, Japan.

[37] Ely Repiso, Francesco Zanlungo, Takayuki Kanda, Anaís Garrell, and Alberto Sanfeliu. 2019. People's v-formation and side-by-side model adapted to accompany groups of people by social robots. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2082–2088.

[38] J. Rios-Martinez, A. Spalanzani, and C. Laugier. 2014. From proxemics theory to socially-aware navigation: A survey. *International Journal of Social Robotics*, 7(2):137–153.

[39] Jorge Rios-Martinez, Anne Spalanzani, and Christian Laugier. 2011. Understanding human interaction for probabilistic autonomous navigation using risk-rrt approach. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2014–2019.

[40] Leonardo Scandolo and Thierry Fraichard. 2011. An anthropomorphic navigation scheme for dynamic scenarios. In *2011 IEEE International Conference on Robotics and Automation*, pages 809–814.

[41] Francesco Setti, Chris Russell, Chiara Bassetti, and Marco Cristani. 2015. F-formation detection: Individuating free-standing conversational groups in images. *PLOS ONE*, 10(5):1–26.

[42] Mikael Svenstrup, Søren Tranberg Hansen, Hans Jørgen Andersen, and Thomas Bak. 2011. Adaptive human-aware robot navigation in close proximity to humans. *International Journal of Advanced Robotic Systems*, 8(2):21.

[43] Elena Torta, Raymond H. Cuijpers, and James F. Juola. 2013. Design of a parametric model of personal space for robotic social navigation. *Int. J. Soc. Robotics*, 5(3):357–365.

[44] Stéfan van der Walt, Johannes L. Schönberger, Juan Nunez-Iglesias, François Boulogne, Joshua D. Warner, Neil Yager, Emmanuelle Gouillart, and Tony Yu. 2014. scikit-image: Image processing in python. *CoRR*, abs/1407.6245.

[45] Marynel Vázquez, Aaron Steinfeld, and Scott E. Hudson. 2015. Parallel detection of conversational groups of free-standing people and tracking of their lower-body orientation. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3010–3017.

[46] Sujeong You and Sanghoon Ji. 2017. F-formation based navigation planner for a mobile servant robot. In *2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 660–662.

[47] Kaiyu Zheng. 2017. ROS navigation tuning guide. *CoRR*, abs/1706.09068.