Human-Robot Spatial Interaction using Probabilistic Qualitative Representations

by
Christian Dondrup

A thesis presented for the degree of
Doctor of Philosophy

June, 2016

UNIVERSITY OF
LINCOLN

School of Computer Science
University of Lincoln
United Kingdom

Abstract

Current human-aware navigation approaches use a predominantly metric representation of the interaction which makes them susceptible to changes in the environment. In order to accomplish reliable navigation in ever-changing human populated environments, the presented work aims to abstract from the underlying metric representation by using Qualitative Spatial Relations (QSR), namely the Qualitative Trajectory Calculus (QTC), for Human-Robot Spatial Interaction (HRSI). So far, this form of representing HRSI has been used to analyse different types of interactions offline. This work extends this representation to be able to classify the interaction type online using incrementally updated QTC state chains, create a belief about the state of the world, and transform this high-level descriptor into low-level movement commands. By using QSRs the system becomes invariant to change in the environment, which is essential for any form of long-term deployment of a robot, but most importantly also allows the transfer of knowledge between similar encounters in different environments to facilitate interaction learning. To create a robust qualitative representation of the interaction, the essence of the movement of the human in relation to the robot and vice-versa is encoded in two new variants of QTC especially designed for HRSI and evaluated in several user studies. To enable interaction learning and facilitate reasoning, they are employed in a probabilistic framework using Hidden Markov Models (HMMs) for offline classification and evaluation of their appropriateness for the task of human-aware navigation.

In order to create a system for an autonomous robot, a perception pipeline for the detection and tracking of humans in the vicinity of the robot is described which serves as an enabling technology to create incrementally updated QTC state chains in real-time using the robot's sensors. Using this framework, the abstraction and generalisability of the QTC based framework is tested by using data from a different study for the classification of automatically generated state chains which shows the benefits of using such a high-level description language. The detriment of using qualitative states to encode interaction is the severe loss of information that would be necessary to generate behaviour from it. To overcome this issue, so-called Velocity Costmaps are introduced which restrict the sampling space of a reactive local planner to only allow the generation of trajectories that correspond to the desired QTC state. This results in a flexible and agile behaviour

generation that is able to produce inherently safe paths. In order to classify the current interaction type online and predict the current state for action selection, the HMMs are evolved into a particle filter especially designed to work with QSRs of any kind. This online belief generation is the basis for a flexible action selection process that is based on data acquired using Learning from Demonstration (LfD) to encode human judgement into the used model. Thereby, the generated behaviour is not only sociable but also legible and ensures a high experienced comfort as shown in the experiments conducted. LfD itself is a rather underused approach when it comes to human-aware navigation but is facilitated by the qualitative model and allows exploitation of expert knowledge for model generation. Hence, the presented work bridges the gap between the speed and flexibility of a sampling based reactive approach by using the particle filter and fast action selection, and the legibility of deliberative planners by using high-level information based on expert knowledge about the unfolding of an interaction.

Declaration

I, *Christian Dondrup*, declare that this thesis and the work presented in it are my own and has been generated by me as the result of my own original research.

I confirm that this work was done wholly while in candidature for a research degree at the University of Lincoln and that this thesis has not previously been submitted for a degree or any other qualification at this University or any other institution. I also confirm that where I have consulted the published work of others, this is always clearly attributed and that where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work and I have acknowledged all main sources of help. I confirm that where the thesis is based on work done by myself jointly with others, I have made clear exactly what the other authors and I contributed myself which can be found in Appendix A. Parts of this work have been published as:

Dondrup, C., Lichtenthäler, C. & Hanheide, M., *Hesitation signals in human-robot head-on encounters: a pilot study*, *in* 'Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction', (2014), ACM, pp. 154–155.

Dondrup, C., Bellotto, N. & Hanheide, M., *A probabilistic model of human-robot spatial interaction using a qualitative trajectory calculus*, *in* '2014 AAAI Spring Symposium Series', (2014).

Dondrup, C., Bellotto, N. & Hanheide, M., *Social distance augmented qualitative trajectory calculus for human-robot spatial interaction*, *in* 'IEEE International Symposium on Robot and Human Interactive Communication', RO-MAN, pp. 519–524, (2014).

Dondrup, C., Bellotto, N., Hanheide, M., Eder, K. & Leonards, U., *A computational model of human-robot spatial interactions based on a qualitative trajectory calculus*, *Robotics* **4**(1), 63–102, (2015).

Dondrup, C., Bellotto, N., Jovan, F. & Hanheide, M., *Real-time multisensor people tracking for human-robot spatial interaction*, *in* 'Workshop on Machine Learning for Social Robotics', ICRA/IEEE, (2015)

Dondrup, C. & Hanheide, M., *Qualitative constraints for human-aware robot navigation using velocity costmaps*, *in* 'IEEE International Symposium on Robot and Human Interactive Communication', RO-MAN, (2016).

Christian Dondrup

Acknowledgements and Dedication

To my mother
who with her eternal support enabled me to pursue this degree.

Firstly, I would like to thank my supervisors Marc Hanheide and Tom Duckett who always offered valuable advise and insight, helped me along my journey as a PhD candidate, and became good friends. Especially Marc Hanheide always offered support even late at night close to paper deadlines, contributed very many good ideas regarding experiment and software design, asked the right questions to have me rethink my approach when I got stuck, always knew what to try next, and shared the same opinions regarding forms and bureaucracy. Thank you both for giving me this great opportunity and helping me along the way. I would also like to thank my examiners Astrid Weiss and John Shearer for all their efforts and great suggestions for the final version of this thesis.

Secondly, I would like to thank everybody who made my time in Lincoln unforgettable. First and foremost, I want to thank João Santos for being the best house mate I could have wished for, my partner in crime and good friend. I also want to thank Jaime Pulido Fentanes and Tomáš Krajník working on the STRANDS project with João and me for all their great work that was the basis for almost all the approaches developed in the project. I also want to thank Peter Lightbody, Claudio Coppola, Keerthy Kusumam, Anna Frohnwieser, Tanja Kleinhappel, Sam Penrice, John Shearer, Ben Kirman and all the others I cannot mention here for the good times in Lincoln. Thank you all for a great three years and being such good friends. Without you, Marc, and Tom it wouldn't have been the same and I don't think that I would have felt at home as much as I did.

I also want to thank everybody in the STRANDS project and especially the coordinator Nick Hawes for all their great work and the good times at review meetings and project assemblies we had. I hope to work with all of you again in the future or at least grab a drink once in a while.

This work is based on several published papers and during my time as a PhD candidate I was involved in a few others for which I would like to thank my co-authors Marc Hanheide, Nicola Bellotto, Christina Lichtenthähler, Ferdian Jovan, Kerstin Eder, Ute Leonards, Denise Hebesberger, Kathrin Gerling, Peter Lightbody, and Alyxander May for all their work. Thank you for your commitment and valuable contributions. Additionally, I would also like to thank Marc Hanheide, Michael Dondrup, and Peter Lightbody for proof reading this thesis.

Last but certainly not least, I want to thank Katrin Lohan without whom I would have never had the possibility of perusing a career in research or academia. You changed my life for the better and for that I am eternally grateful.

I would also like to acknowledge that the research leading to the results presented in this thesis has received funding from the European Community's Seventh Framework Programme under grant agreement No. 600623, STRANDS

Contents

List of Figures

List of Tables

Nomenclature

k, l	Denoting the Moving Point Objects in the Qualitative Trajectory Calculus
$d(\cdot)$	The Euclidean distance
S_B	The set of states for the Qualitative Trajectory Calculus – Basic
S_{BC}	The set of states for the Qualitative Trajectory Calculus – Basic/Double-Cross
S_C	The set of states for the Qualitative Trajectory Calculus – Double-Cross
τ_B	The set of legal transitions for the Qualitative Trajectory Calculus – Basic
τ_{BC}	The set of legal transitions for the Qualitative Trajectory Calculus – Basic/Double-Cross from Basic to Double Cross
τ_C	The set of legal transitions for the Qualitative Trajectory Calculus – Double-Cross
τ_{CB}	The set of legal transitions for the Qualitative Trajectory Calculus – Basic/Double-Cross from Double Cross to Basic
$\rightsquigarrow$	Valid Qualitative Trajectory Calculus transition
d_s	The distance threshold for the transition in the Qualitative Trajectory Calculus – Basic/Double-Cross
d_c	The conceptual distance in the original formulation of the Qualitative Trajectory Calculus
ϕ	The conceptual distance of two symbols in the Qualitative Trajectory Calculus
$\phi(\cdot)$	The function to calculate the conceptual distance of two symbols in the Qualitative Trajectory Calculus
Φ	The symbol wise conceptual distance of two states in the Qualitative Trajectory Calculus
$\Phi(\cdot)$	The function to calculate the symbol wise conceptual distance of two states in the Qualitative Trajectory Calculus
ρ	The linear velocity of the robot or the distance from the centre of a Polar coordinate system (used equivalently)
θ	The angular velocity of the robot or the distance from the 0 angle of a Polar coordinate system (used equivalently)
$\mathbf{V}_a$	The set of admissible velocities for the Dynamic Window Approach local planner

$\mathbf{V}_d$	The set of reachable velocities given the acceleration limits of the robot for the Dynamic Window Approach local planner a.k.a. the dynamic window
$\mathbf{V}_r$	The search space of velocities for the Dynamic Window Approach local planner
$\mathbf{V}_s$	The set of all velocities for the Dynamic Window Approach local planner
$\mathfrak{T}$	The set of sampled trajectories from the Dynamic Window Approach local planner
$\boldsymbol{\mathcal{C}}$	The set of critique functions for the Dynamic Window Approach local planner
$\boldsymbol{\omega}$	The set of weights for the critique functions of the Dynamic Window Approach local planner
$\gamma(\cdot)$	The cost function for trajectory samples $\mathfrak{t}_i \in \mathfrak{T}$ based on $\omega_i \in \boldsymbol{\omega}$ and $c_i \in \boldsymbol{\mathcal{C}}$ used in the Dynamic Window Approach local planner
γ	The costs of a trajectory $\mathfrak{t} \in \mathfrak{T}$ used in the Dynamic Window Approach local planner
$\boldsymbol{a}$	Low cost areas in the Velocity Costmap
α	The angular free and/or no cost area size in the Velocity Costmap
δ	The direction angle for Velocity Costmap creation
λ	The angle of the centre of the open area in the Velocity Costmap using the current reltive angle of the human to the robot and δ
ϱ	The linear free and/or no cost area size in the Velocity Costmap
ξ	The set of allowed velocities in the Velocity Costmap combining ξ_α and ξ_ϱ
ξ_α	The set of allowed angular velocities in the Velocity Costmap
ξ_ϱ	The set of allowed linear velocities in the Velocity Costmap
$\boldsymbol{\mathcal{A}}$	The set of all possible robot actions
$\boldsymbol{\mathcal{M}}$	The set of Markov Models describing specific interactions
N_p	The number of particles used in the filter
$\mathcal{O}$	The observation model of the particle filter
$\boldsymbol{\Omega}$	The set of all possible states in any given QTC variant
$\boldsymbol{\mathcal{P}}$	The set of conditional probability tables $P(\mathcal{A} \mid \Omega)$ describing the joint probabilities of every action $\mathcal{A} \in \boldsymbol{\mathcal{A}}$ given the current observation $\Omega \in \boldsymbol{\Omega}$ for action selection
ς	The starvation factor used in the particle filter to prevent models from dying
$\mathbf{X}$	The set of all possible particles
$\mathbf{x}$	The list of all instances of particles $x \in \mathbf{X}$

—The most exciting phrase to hear in science, the one that heralds the most discoveries, is not 'Eureka!' but 'That's funny...'

Isaac Asimov, Professor of Biochemistry/Author

1 Introduction

Automation has always played a big role in the progress of humanity like the invention of the steam engine leading to the industrial revolution. In the digital age, robots became ever more popular to do work that is either highly repetitive, requires precise movements, or is simply too dangerous for humans to fulfil. Recently, there are also more and more social robots available on the consumer market like Pepper and NAO from Aldebaran[1] or Paro[2] to only name a few. Hence, there is no doubt that robots are moving into our homes and workplaces even further than they already have. Thus, the field of Human-Robot Interaction (HRI) has grown significantly since its birth in the early 1990s. By now, HRI is a vast field encompassing all kinds of possible interactions like tutoring (e.g. Vollmer et al. 2009), teaching (e.g. Chang et al. 2010), therapy of children with autism (e.g. Robins et al. 2005), elder care (e.g. Broekens et al. 2009), rehabilitation (e.g. Gross et al. 2016), or acceptance in general (e.g. Weiss et al. 2008) to name only a few popular

[1] `http://www.aldebaran.com`
[2] `http://www.parorobots.com/`

examples. This thesis focuses on the subfield of Human-Robot Spatial Interaction (HRSI) which is the study of human(s) and mobile robot(s) locomotion in a confined shared space and the social signals governing this interaction. Hence, HRSI is the overarching term for applications like human-aware navigation, human-robot joint motion, etc.

HRSI has gained ever more importance since one of the first publications on human-aware navigation by Simmons (1996), because moving safely in the presences of humans does not only increase the robots acceptance (e.g. Gross et al. 2009) but also arguably makes navigation in populated environments more efficient. Hence, the safe navigation in the presence of humans is of utmost importance in such environments (Steinfeld et al. 2006). Nowadays, robots are able to navigate safely around static and dynamic obstacles. However, safe in the context of HRSI does not only mean collision avoidance, but also producing movement that is perceived as safe by the human – which comes down to factors like simply giving the human more space when avoiding (e.g. Pacchierotti et al. 2006) or more subtle factors like producing legible movements according to the definition by Lichtenthäler et al. (2012): "A robot's behaviour is legible, if a human can predict the next actions of the robot and the robot behaviour fulfils the expectations of a human interaction partner". For a robot to be a useful tool in daily life, however, it has to be deployed over long periods of time and therefore has to be able to cope with changes in its environment. To this end, the presented thesis aims at describing a human-aware navigation approach that is able to deal with an ever changing world, while still making informed decisions about the actions to be executed. This deliberation in the face of constant change has so far not been addressed by the HRSI community.

Possible interactions in HRSI are manifold and range from "simple" avoidance to more complicated scenarios like guiding or joint tasks like carrying an object between human and robot. This thesis focuses on the avoidance of humans in everyday environments, which is the main field of application for human-aware navigation approaches. Two typical example interactions for this kind of interaction on which the final evaluation of the presented work focuses in particular are

Pass-by A human and a robot walk along a corridor in opposite directions. At some point both will meet and have to communicate their goal, negotiate who goes to which side, who starts the avoidance, or gives up their right of way in cases of partially blocked

corridors. During the interaction, the behaviour of both has to be legible to the interaction partner and should respect social norms regarding the execution of the movement like distance kept, speed, acceleration, etc., but also has to be goal-directed in order to fulfil the robots primary task at the location it is travelling to.

Path crossing At a crossroads of two corridors, both human and robot try to reach goals on trajectories perpendicular to each other. Both agents will only be aware of each other at a very late point in time due to the walls of the building hiding their approach. When they meet they have to communicate their goal and negotiate who lets the other pass first. Similar to the pass-by, the behaviour of both has to be legible to the interaction partner and should respect social norms regarding the execution of the movement like distance kept, speed, acceleration, etc., but also has to be goal-directed in order to fulfil the robots primary task at the location it is travelling to.

Looking at these two examples, it becomes obvious that negotiation plays a vital role in human-aware navigation. This can for example be achieved via prompting (Peters 2011) or legible movement (Lichtenthäler et al. 2012) to communicate the intention of the robot and ensure successful HRSI. In addition to the communicative character of motion, the distance between the two agents and the speed and manner at which they travel greatly influences the perceived interaction and its task efficiency. Using the various factors that can be influenced by the robot, e.g. speed, acceleration, distance kept, trajectory executed, HRSI seeks to optimise the following criteria according to Kruse et al. (2013)

Comfort is the absence of annoyance and stress for humans in interaction with robots. This also includes perceived safety as it reduces the humans' stress level.

Naturalness is the similarity between robots and humans in low-level behaviour patterns.

Sociability is the adherence to explicit high-level cultural conventions.

The presented thesis aims at improving this experienced comfort and the perceived sociability of the robot while interacting with the human. This can be achieved in a number of ways like modelling geometric constraints (e.g. Ohki et al. 2010), learning from data (e.g. Luber et al. 2012), etc. but these approaches almost exclusively rely on metric

representations of the interactions or the space where they take place (Kruse et al. 2013) which makes them very susceptible to error if this underlying metric representation of the world changes significantly. This work, on the other hand, aims to present an approach that is robust to changes in the environment and thus also allows for the easy transfer of knowledge facilitating interaction learning. Apart from the obvious benefits of the knowledge transfer, the motivation for this lies in the very nature of human populated environments – wherever there are humans, the environment is bound to change over time. As mentioned above, in order to allow robots to reach their full potential they have to be able to cope with, model (Krajnik et al. 2014), or even exploit changes (Santos et al. 2016) and cannot rely on a static representation of the environment that might become outdated. One way of coping with change is to abstract from the underlying metric representation of the environment completely and build a qualitative model that represents the essence of a location, action, or interaction. To this end, this work introduces Qualitative Spatial Relations (QSR) for the use in HRSI to facilitate knowledge transfer.

There are of course other approaches that are agnostic to the actual environment like cost functions or social force models based on the distance to the human (e.g. Sisbot et al. 2007). The vast majority of these systems, however, are purely reactive which can easily lead to illegible behaviour because they frequently re-plan their actions which often results in contrasting behaviours from one second to the other. Due to the missing representation of the environment, it is therefore impossible to follow a deliberative planning approach that would facilitate legible behaviour by expressing some form of commitment to a specific path. These deliberative approaches on the other hand, in addition to the assumption of an a-priori known static environment, are very costly because they have to consider the whole environment when planning the next n best actions. To bridge this gap between the legible behaviour produced by deliberative and the flexibility, speed, and robustness to change of the reactive approaches, this work aims at using a qualitative representation of the interactions between a human and a robot. Hence, the system encodes the future unfolding of an interaction qualitatively and uses a dynamic belief and behaviour generation based on this model which is flexible enough to switch between different robot behaviours given a significant change in human behaviour but is also legible, perceived as safe, and ensures a high experienced comfort of the human interaction partner.

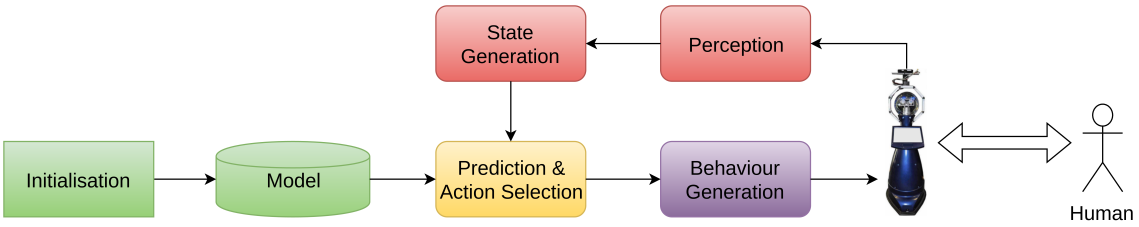


Figure 1.1: *The HRSI loop functioning as a guidance throughout this thesis by showing the interplay between the human and the behaviour generation framework. Starting with the perception of the human behaviour and the qualitative state generation, the belief generation using an online classifier, the model of the interaction initialised with prior knowledge from observation and/or demonstration used for the classification process, and the generation of behaviour based on the model and the current belief of the world. This figure is taken apart and constructed from the bottom up in the following chapters to visualise their contribution to the system.*

Using QSRs for HRSI is a novel approach to representing these interactions in a versatile, human-readable, and generalisable way that was first introduced by Hanheide et al. (2012) who used it for the analysis of interactions. This thesis builds on a similar description of the interaction, extends it to a state where it can be used for online belief and behaviour generation and shows how to use it for the control of a mobile robot. Thus, this framework allows to generalise between different people in similar situations across different environments, for the possibility of having different models for different groups of people or individuals, e.g. patients and staff in an elder care home, and to encode knowledge about the future unfolding of the interaction into an otherwise often reactive process. Hence, the approach is not only able to generate safe behaviour in the presence of humans but also shows commitment to a chosen action without requiring costly planning over a large state space and, therefore, works in real-time. A conceptual overview of the proposed system can be seen in Figure 1.1. The probabilistic qualitative model needs to be initialised from recorded data of similar interactions to ensure safe and legible robot navigation. The robot itself has to be able to perceive humans in order to classify the interaction type and generate a belief about the current state of the interaction based on this model. The resulting belief can then be used for low-level behaviour generation in the form of velocities sent to the robot's wheels. This figure represents the separate method chapters of this work by colour coding boxes that are described in conjunction and is used to guide the reader throughout the chapters by visualising their individual contribution to the overall system and showing how the system develops from the bottom up.

Last, but not least, in order to achieve the desired legibility, safety, and experienced comfort, this work presents another popular principle of robotics which is currently underused in HRSI – Learning from Demonstration (LfD). While there are many other approaches to achieve these goals, such as tuning cost functions (e.g. Lu et al. 2013), defining environmental constraints (e.g. Morales et al. 2015), or learning from data sets (e.g. Ziebart et al. 2009), LfD offers the unique opportunity to include human judgement into the robot's action selection. To this end, the robot is remote controlled by a human during a learning phase and the resulting qualitative states are then used to generate a model of the interaction and state to action mappings. This way no reasoning about any constraints has to be undertaken – the robot only relies on the expert knowledge encoded in the representation used. This results in a fast and flexible approach to human-aware navigation.

1.1 Aims and Objectives

Before formulating the aims and objectives of this work it is important to clarify the terminology used throughout the remainder of this thesis. The first distinction to make is between the terms *behaviour* and *task* where for the remainder of the thesis the word behaviour refers to how the robot navigates in the presence of a human and the word task to the abstract high-level goal the robot is trying to achieve, e.g. delivering a parcel, surveying and area, or driving to its charging station. Therefore, the task describes the main purpose of the robot which leads it to navigate through the populated environment where behaviour describes how it reacts to the presence of humans while navigating. When it comes to the evaluation of the system, apart from showing the technical feasibility of a human-aware navigation approach based on QSRs, the presented systems aims to fulfil certain HRSI criteria. In accordance with Kruse et al.'s (2013) definitions, this human-aware navigation system aims to increase the experienced comfort of the human interaction partner by increasing the perceived safety and decreasing the experienced stress and annoyance. This stress and annoyance can be caused by several environmental factors, but with regards to navigation the main influence, apart from perceived safety, stems from the robot's sociability and naturalness of the movement. The latter, however, is more focused on low-level motor control which is not the focus of this work. Hence, naturalness will not be

investigated. Moreover, in addition to Kruse et al.'s (2013) definitions, the legibility of the robot's movement according to Lichtenthäler et al. (2012) is considered arguing that by ensuring the interaction partner can predict the robot's movements and the robot fulfils their expectations the annoyance and stress during the interaction is reduced which in turn also increases the experienced comfort. Therefore, when speaking of comfort in the context of the approach presented here, it refers to achieving perceived safety, legibility, and sociability. Apart from these more HRSI oriented metrics the robot has to fulfil its task which is the sole reason for it to navigate the environment in the first place. In order to do so, the behaviour also has to be efficient with regards to the distance travelled and the travel time to ensure that the robot is able to reach its navigation goal where the task is performed given time and battery constraints. Hence, the system developed for this thesis not only aims to ensure a high experienced comfort but also task efficiency.

To summarise and formalise above introduction and building on the mentioned criteria of comfort and efficiency, the aim of the presented work is to build a generative framework for HRSI focusing on human-aware navigation that is able to abstract from metric representations of the world and, therefore, becomes robust to change in the environment and allows for the easy transfer of knowledge. Additionally, the system has to be flexible regarding changes in the behaviour of the human and adapt to that change while at the same time generate safe, legible, and sociable trajectories showing commitment to a certain action to prevent the confusion caused by purely reactive approaches. In order to achieve this legibility, safety, and adherence to social norms the system ought to be able to be trained from observation or demonstration to include human judgement in the generated qualitative models to avoid relying on constraints based on metric information like many other approaches. Moreover, and most importantly for a system deployed on a mobile robot, it has to be task efficient and work in real-time by avoiding costly planning with a large look-ahead using only the robot's on-board sensors where real-time in this use case does not refer to hard real-time but to a minimal delay of $\sim 300ms$ between sensor output and the generation of the corresponding reaction by the robot. From this and above introduction, the following objectives are derived.

Obj. 1. Robust qualitative interaction models In order to achieve robustness to change and facilitate knowledge transfer, the interaction has to be encoded qualitatively – abstracting from the underlying metric representation of the environment and interaction. This entails using models that are descriptive enough to unambiguously describe different interactions between a human and a robot which in conjunction with a probabilistic representation can be used to reliably classify different encounters. The specific requirements to such a model are:

Obj. 1.1 *Abstracting from the metric environment representation* to generate a purely qualitative model that is robust to changes in the environment and allows easy knowledge transfer.

Obj. 1.2 *Representing the qualitative character of motions* of both agents including changes in direction, stopping or starting to move, etc. It is known that small movements used for prompting (Peters 2011) are essential for a robot to interpret the intention of the human and to react in a socially adequate way.

Obj. 1.3 *Representing the relevant attributes of HRSI situations* in particular proxemics (Hall 1969), i.e. the distance between the interacting agents. This is required for behaviour generation, to analyse the perceived safety of the interaction, and to attribute intention of the implicitly interacting agents.

Obj. 1.4 *Ability to generalise* over a number of individuals and situations. A robot requires this ability to utilise acquired knowledge from previous encounters of the same or similar type. A qualitative framework that is able to create such a general model, which still holds enough information to unambiguously describe different kinds of interactions but abstracts from metric space, facilitates learning and reasoning.

Obj. 1.5 *A tractable, concise, and theoretically well-founded model* is necessary for the representation and underlying reasoning mechanisms in order to be deployed on an autonomous robot.

Obj. 1.6 *Facilitating decision processes* by having a clear association between human and robot state where the robot state has to be produced and can therefore not be used for classification and prediction. Since the robot is meant to make decisions based on the human's state, this is paramount for a generative qualitative framework.

Obj. 2. Comfortable and task effecient behaviour generation In order to increase the acceptance of a mobile robot in populated environments its movements have to ensure high experienced comfort by being safe, legible, and sociable. This entails the commitment to a specific action, and adherence to social norms like distances kept or driving on a specific side of a corridor but the behaviour also has to be task efficient by minimizing the distance and time travelled to allow the robot to fulfil its primary goal. This dilemma of task driven versus social robot behaviour requires:

Obj. 2.1 *Legibility and sociability* meaning that the robot's behaviour has to be predictable by and has to fulfil the expectations of its human interaction partner and adhere to social norms. In the case of legibility, this includes a certain degree of commitment to an action to avoid constant re-planing and the resulting changes in behaviour. In terms of sociability, the robot should adhere to social conventions like driving on the left or right of the corridor, etc.

Obj. 2.2 *Safe movement* has to be ensured. The perceived safety of the human should never be compromised if not explicitly required by the task.

Obj. 2.3 *Task efficient movement* is required to fulfil the primary task of the robot. If the robot's task is time critical, the travel time has to be kept at a minimum while still ensuring the safety of the human interaction partners encountered en route and maintaining a high degree of legibility and sociability.

Obj. 2.4 *Fast and flexible action selection* is required to adapt the behaviour of the robot, should the humans' behaviour change significantly, to still meet expectations and ensure legibility. In such a case, the belief of the current state of the world has to be adapted to best represent the actual state of the world, resulting in dynamic reclassification of the interaction type and an action selection policy change. Thus, the system has to be flexible and reactive in nature to be predictable and fulfil expectations.

Obj. 3. Autonomy The created human-aware navigation framework has to be deployable to an autonomous mobile robot. This requires seamless integration with the robotic hardware like sensors and actuators using standardised software components to allow for

easy deployment and the ability to autonomously and in real-time decide on the next best action to perform. The specific requirements for such a system are:

Obj. 3.1 *Working in real-time* is require both on the processing and behaviour generation side in order to react to the human's presence and to adapt the robot's behaviour. As mentioned above, real-time does not refer to hard real-time but to a minimal delay between sensor input and behaviour output to generate an impression of instantaneous reactions to sensor input.

Obj. 3.2 *Relying on on-board sensors and processing alone* is a constraint of most real-world environments which are either too large to put up additional sensors, have ethical restrictions against surveillance, bad WiFi connectivity, etc. Hence, all the perception and decision making has to be done on the robot.

Obj. 3.3 *Generated behaviour has to be tailored to the hardware* by taking acceleration limits, turning angles, top speed, etc. into account.

Whenever, one of these objectives is addressed specifically throughout the document it is highlighted in ***bold and italic*** and the number is referenced like [Obj. 1] or [Obj. 1.1].

1.2 Main Contributions

This section lists the novel contributions of this thesis to the field of HRSI in order to fulfil the listed objectives and create an autonomous system for human-aware navigation using QSRs on a mobile robot.

One of the novel contributions of this thesis is the use of Qualitative Spatial Relations (QSR) to ***abstract from the metric environment representation [Obj. 1.1]*** creating an environment agnostic interaction model for Human-Robot Spatial Interaction (HRSI) focussing on human-aware navigation. This is based on the interaction model presented in Chapter 3 introducing the QSR used which is a combination of two well known Qualitative Trajectory Calculus (QTC) variants, i.e. Qualitative Trajectory Calculus – Basic (QTC_B) and Qualitative Trajectory Calculus – Double-Cross (QTC_C), into a new model that is able to ***represent the qualitative character of the motions [Obj. 1.2]*** of human and robot and, therefore, ***facilitates decision making processes [Obj. 1.6]***.

This new QTC model is then used in a probabilistic representation which is trained from real-world data gathered in two different experiments ***generalising over a number of individuals and situations [Obj. 1.4]***. A distance threshold is used to switch between the two variants of QTC which allows to implicitly ***represent this relevant attribute of HRSI [Obj. 1.3]*** via the transition from one variant to the other. Thus, the model is able to highlight the interaction between the two agents involved in close vicinity to each other which, apart from reducing noise, can be used in later chapters to ***generate behaviour suitable for human-aware navigation [Obj. 2]***. Additionally, it allows implicitly modelling a discrete distance threshold without losing any of the qualitative properties of the underlying ***tractable, concise, and theoretically well-founded [Obj. 1.5]*** calculi. The model is evaluated using data from two different experiments to show that it can reliably classify different HRSI encounters and is, therefore, a suitable representation.

In order to use this model ***relying only on the robot's hardware [Obj. 3.2]***, the combination of existing detection and tracking approaches for human perception in the vicinity of the robot and the ***automated incremental generation of QTC state chains in real-time [Obj. 3.1]*** is described in Chapter 4. The perception framework ***tailored to the capabilities of the robot [Obj. 3.3]*** presented in Section 1.3.2 ***uses all available sensors producing people tracks in real-time [Obj. 3.1]***. This system is used to automatically generate QTC state chains for every tracked person in order to classify the type of HRSI encounter the robot ought to engage in. These generated state chains are the basis for the classification and behaviour generation approaches described in the following chapters. As a proof of concept a small experiment using a mobile robot in a real-world office environment, ***only relying on the robots on-board sensors [Obj. 3.2]***, shows how these generated QTC state chains can be classified using the Hidden Markov Models (HMMs) trained from data of a previous experiment. Thus, the evaluation not only shows that it is possible to use the robot's on-board sensors to generate a meaningful QTC representation but also that this particular QSR allows to ***transfer knowledge gathered from different sensors and in a different environment [Obj. 1.4]*** to bootstrap the classification system.

Using the generated QTC state chains, Chapter 5 describes the ***generation of movement commands [Obj. 2]*** for a mobile robot from QSRs, representing a novel approach to human-aware navigation. Using this high-level QTC based representation, low-level

command velocities that can be send to the robot's wheels are generated using so-called *Velocity Costmaps* which restrict the sample space of a local planner to generate trajectories that produce the desired QTC state. These Velocity Costmaps introduced in this chapter, on the one hand, ***produce trajectories that are safe and also perceived as safe [Obj. 2.2]*** by the human interaction partner and, on the other hand, are still ***able to minimise time and distance travelled towards the goal to generate task efficient behaviour [Obj. 2.3]***. These costmaps are based on either hand crafted rules as done in Chapter 5 or a learned conditional probability table as shown in Chapter 6 which allows incorporating human judgement via Learning from Demonstration (LfD). The Velocity Costmaps are evaluated in simulation and a real-world proof-of-concept experiment to show that this approach is able to generate behaviour that conforms with the desired QTC state.

Finally, to generate a ***fully autonomous system [Obj. 3]*** that is not dependant on hand-crafted action selection rules or a-priori knowledge about the interaction type, a particle filter for QSRs which builds on a prediction and observation model that is generic enough to allow the use with any kind of QSR is introduced in Chapter 6. In the case at hand, it uses an evolved version of the QTC model described in Chapter 3 which consists of a conglomerate of different QTC states using different variants of the calculus to create a ***representation that is meaningful enough to unambiguously distinguish different types of interactions without relying on the robot's state for classification [Obj. 1.6]*** but only taking the human observation into account. These models are learned from observation during a training phase where the robot is remote controlled by either a participant or an experimenter while another experimenter or participant interacts with it. The best action for the robot is selected from a conditional probability table that describes the joint probability for each possible action given the current belief. Compared to Chapter 5, where these were hand-crafted rules, the conditional probability tables for action selection are learned from demonstration using an "Inverse Oz of Wizard" experiment set-up to ***incorporate human judgement into the behaviour model to ensure sociability and legibility [Obj. 2.1], and safety [Obj. 2.2]*** which is shown in the experiment section of this chapter.

The whole system is evaluated in a two-part user study in Chapter 6 with the first part representing the learning phase where the robot is remote controlled by a partic-

ipant while interacting with the experimenter, recording the generated QTC states for the activity model and the conditional probability table for action selection. The second part evaluates the learned models and ***belief generation for fast and flexible action selection [Obj. 2.4]*** using a separate set of participants and a fully autonomous robot that has no prior knowledge about the interaction type showing that it is able to ***generate comfortable and task efficient behaviour [Obj. 2]***.

Publications The presented thesis is based on a number of published conference and journal articles. To disambiguate the contribution of the author of this thesis to the papers it is based on, please refer to Appendix A for a comprehensive list of publications that are either described in detail in the following chapters or are referenced throughout this thesis.

Additionally, all presented approaches are freely available as open source or pre-compiled Debian packages.[3] Please refer to Appendix B for a collection of links to the software repositories, websites with videos showing the working system and instructions on how to use it.

1.3 Context

To paint a clear picture of the scope and the underlying motivation for this work, it is important to understand the background it is based on in terms of projects, hardware and software. Hence, this section describes the European Project it was part of, the robot hardware which has been used for all the experiments described in the following chapters, and the most important characteristics of the Robot Operating System (ROS) as the underlying middleware that influenced design decisions for the presented approaches. The content of this section, therefore, describes the context of this thesis and should give an insight into target deployment areas, the influence of the robot appearance on the participants, and the software principles at the foundation of the presented system.

[3] Available under the MIT license where possible or BSD and GPL when using third party software.

1.3.1 The STRANDS Project

The presented work is part of a European FP-7 project called Spatio-Temporal Representation and Activities for Cognitive Control in Long-Term Scenarios (STRANDS)[4], grant agreement No. 600623, which greatly influenced its outcome by defining the area of application. The STRANDS project focuses on long-term deployments of robots and the resulting opportunities and challenges. Environments might change for example and the robot should be able to cope with or even exploit the change to enhance its performance and information gain. To quote the STRANDS proposal:

> *"STRANDS aims to enable a robot to achieve robust and intelligent behaviour in human environments through adaptation to, and the exploitation of, long-term experience. Our approach is based on understanding 3D space and how it changes over time, from milliseconds to months. We will develop novel approaches to extract quantitative and qualitative spatio-temporal structure from sensor data gathered during months of autonomous operation. Extracted structure will include reoccurring geometric primitives, objects, people, and models of activity. We will also develop control mechanisms which exploit these structures to yield adaptive behaviour in highly demanding, real-world security and care scenarios."*

These real-world scenarios are a security scenario where the robot is operating as a night-watchman patrolling an office building and a care scenario where the robot is deployed in an elder care home to assist the administrative and therapy staff. Since the security scenario does not involve much HRSI the main target of application of this thesis is the care scenario in an elder care home in Vienna, Austria. This care home called *Haus der Barmherzigkeit*[5] is home to 350 patients and has a total of 465 staff. All of the patients suffer either from a mobility or cognitive impairment like Dementia, caused by e.g. Alzheimer's or Cardiovascular disease, which makes legible and safe robot behaviour even more important due to people needing more time to comprehend the robot's goal or needing more space to feel safe due to them struggling to walk or using a wheelchair. Apart from that, the robot can also never be an obstacle in case a patient has to be

[4] `http://www.strands-project.eu`
[5] `http://www.hausderbarmherzigkeit.at`

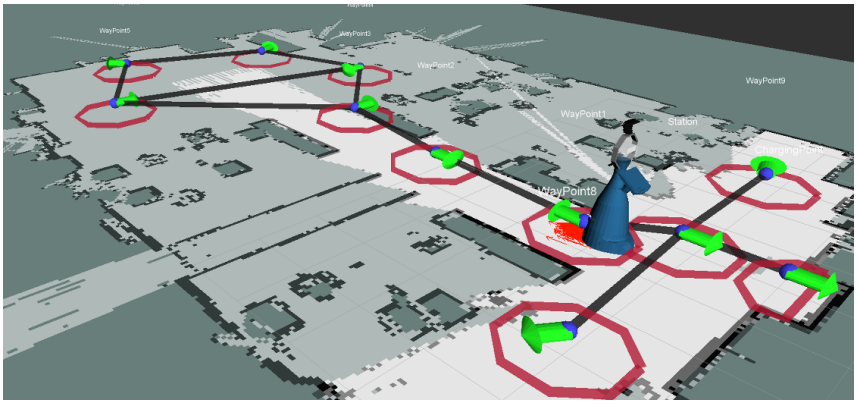


Figure 1.2: *The representation of the environment used for navigation showing the static parts of the Lincoln Centre for Autonomous Systems Research (L-CAS) office as black dots or lines on the metric map. The green arrows surrounded by the red octagons visualise the so-called waypoints in the topological map that is overlaying the metric map. The black lines connecting waypoints are called edges and together they describe the topology of the environment. The robot shown in blue travels between waypoints at which it seeks to fulfil tasks along edges. Image taken from Krajník et al. (2016)*

transported to the Emergency unit in her/his bed or in case of fire. Thus, the creation of robust behaviour is of utmost importance which is hard to near impossible to achieve using metric approaches in changing environments. Hence, the qualitative representation of HRSI has to be able to cope with change and generalise to new environments.

The robot is deployed in both scenarios for several weeks at a time starting with 14 days during the first year up to 120 days during the last year and is expected to work autonomously without any expert intervention.[6] During these times the robot is providing different services to the visitors, staff, and patients of the care home as described by Gerling et al. (2016) which include an info terminal to look up e.g. the lunch menu or the weather, a bellbot service that guides visitors to a chosen target location, and as a pacemaker in a walking group for dementia patients as described by Hebesberger et al. (2016). To get from one location to the other, the robot has to navigate through the populated areas of the building, which is where the presented approach could find its application by interacting with patients, staff, and visitors.

This navigation between different locations in the environment is based on a metric map, i.e. a 2D grid representing a discretised view of the world where each cell in the grid represents if it is occupied or free, and a topological map that represents a high-level description of the environment via so-called waypoints and edges connecting them (see

[6] At the time of writing this thesis, the robot was in its third deployment for a total of 90 days.

Figure 1.2). This navigation approach, especially the topological navigation is exploited in Chapter 6 to incorporate the robot's intention into the online belief generation.

There is so much more that could be said about the STRANDS project but that would go beyond the scope of this thesis. The most important things to keep in mind when reading about the presented work is the requirement of adapting to ever changing environments to be able to produce legible, sociable, safe, and comfortable behaviour at all times and that the robot travels between waypoints in the environment on almost straight lines.

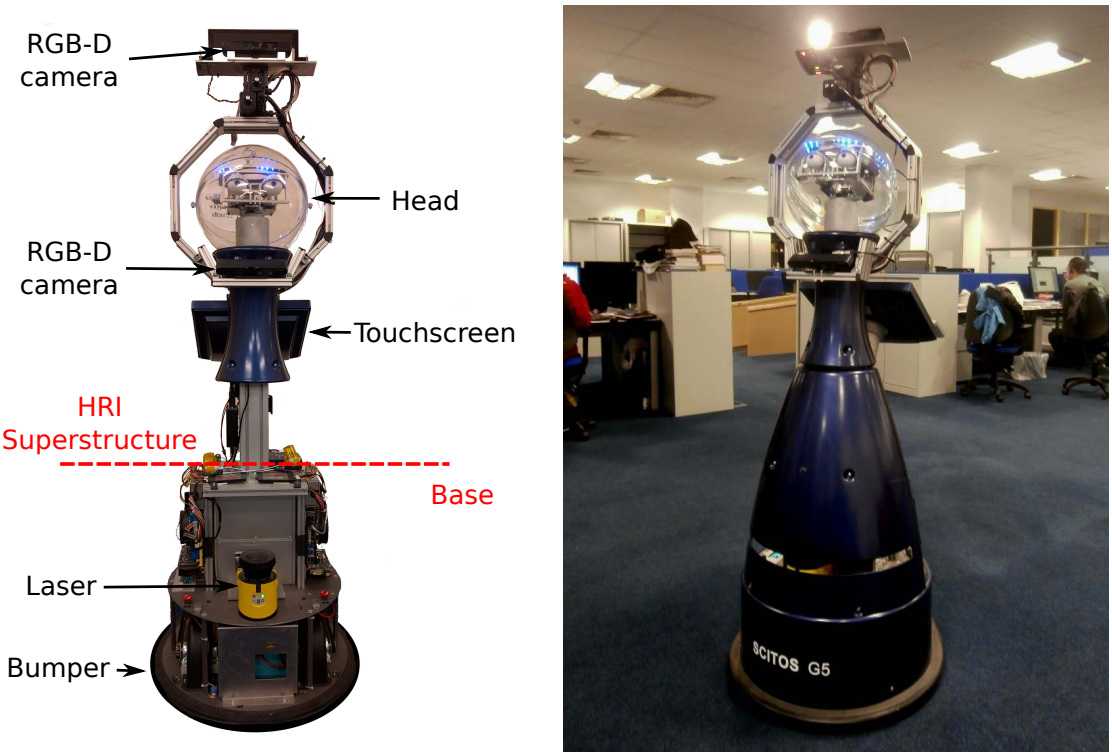
1.3.2 Linda the Robot

The STRANDS project uses a SCITOS G5 mobile base with a HRI super structure produced by the German company MetraLabs GmbH[7]. Since the main focus of the STRANDS project is the durability of software for the long-term deployment of mobile robots, this hardware was chosen due to its promising battery life and robustness. To quote the STRANDS project coordinator from the MetraLabs website:

> *"The SCITOS platform has proven a robust and reliable general purpose research platform, giving us hundreds of hours and kilometers of continuous autonomous use. Metralabs have supported us throughout our challenging project with excellent technical support and software updates as necessary."*
>
> —Nick Hawes, Coordinator of the STRANDS Project

The mobile base of the SCITOS G5 consists of the battery pack, a two wheeled differential drive plus a Caster wheel in the back for stability, an embedded PC, and two WiFi antennas. It has a ground clearance of $20mm$ and can achieve a maximum speed of $1.4^{m}/_{s}$. The mobile base alone weighs $60kg$ and has a payload capacity of $50kg$. For navigation it is equipped with wheel encoders for odometry and a front facing SICK s300 laser range finder or Lidar[8], mounted at a hight of $\sim 35cm$, which is a class 1 laser product and therefore inherently eye safe which is important for the application domains of the STRANDS project. It has a range of $30m$ with a resolution of $3cm$ and a Field of View (FoV) of $270°$ with an angular resolution of $0.5°$. However, due to the covers fitted to

[7] `http://www.metralabs.com`
[8] `http://www.sick.com`



(a) *Linda the robot without the plastic covers. Showing the sensores, and the touch screen and head. Behind the laser is the embeded PC and on eihter side of it the additional PCs. The top camera is mounted on the Pan-Tilt Unit (PTU)*

(b) *Linda the robot with her plastic covers. This is the way the robot was used in all described experiments. The light on top of the robot is used to find its charging station at night time and is turned off during all other tasks.*

Figure 1.3: *Linda the robot, used in all described experiments that use a real robot.*

the robot (see Figure 1.3b), this FoV is restricted to $-1.96rad \leq \rho \leq 1.96rad$, which is roughly 224°, with $\rho = 0$ being directly in front of the robot. The scanning frequency of the Lidar is $125Hz$ which corresponds to one full scan every $80ms$.

The HRI super structure is built on top of the mobile base and consists of an acrylic glass bowl resembling a robotic head mounted on an aluminium pillar (see Figure 1.3a). This acrylic bowl contains a pair of actuated eyes with 5 Degrees of Freedom (DoF) – pan and tilt of the head, opening and closing of the eye lids, and pan of the eyes themselves – to give it an anthropomorphic feature as suggested by Weiss, Mirnig & Förster (2011) and Förster et al. (2011). Additionally, the head comprises a ring of blue Light Emitting Diodes (LEDs) which can be used to indicate internal states of the robot by blinking, pulsing, etc. In addition to the head, the superstructure also comprises a 15″ touch screen, hosting

stereo speakers. This screen is on the "back" of the robot, meaning that it cannot be used to signal oncoming humans, but only following persons which is used in the bellbot and walking group scenario described in Section 1.3.1. To try and enhance the legibility of the navigation even further the robot deployed to the elder care home was also fitted with visual light indicators and uses its head to look into the direction of travel. For a detailed evaluation of this system please refer to the work by May et al. (2015).

Regarding the sensors of the robot, in addition to the Lidar and the wheel encoders, it has a tactile sensor in form of a rubber bumper around the base of the robot that, if pressed, cuts the power to the motors immediately and, thereby, makes it inherently safe for the deployment in populated areas which is why it was awarded with the EU CE certificate.[9] For computer vision approaches, it offers two Asus-Xtion RGB-D cameras[10] of which one of them is mounted underneath the robot's head, facing forwards and down to detect obstacles in 3D space in front of the robot in case of objects that are above or below the height of the laser. The second camera is mounted on a Pan-Tilt Unit (PTU) on top of the robot's head which allows it to pan from $-180°$ to $180°$ and tilt from $-45°$ to $45°$. These cameras offer a FoV of $58°$ horizontally and $45°$ vertically and provide a resolution of 640×480 pixels for depth and RGB. The depth sensor has a range of $0.5m$ to $\sim 7m$.

The robot base including the super structure has a hight of $1.72m$, measured from the head mounted camera, and a diameter of $61cm$ measured in the bottom of the robot. The weight of the whole system is roughly $75kg$. Due to this rather impressive size and weight the robot's maximum velocity is limited to $0.55{}^{m}/_{s}$ in accordance to Butler & Agah's (2001) work to not cause discomfort by reducing the stress stemming from a fast moving large object and increasing the perceived safety.

Computation-wise, the robot in its basic form hosts a 2nd generation Intel Core i7-2640M CPU @ 2.80GHz with 4 cores, 8GB of RAM, a 500 GB HDD, and no dedicated graphics card. During the course of the project, two additional PCs were added where each of them comprises a 4th generation Intel Core i7-4770T CPU @ 2.50GHz with 8 cores, 16GB of RAM, and a 125GB SSD. These PCs, called *side PCs*, are connected via a Local Area Network and each of them hosts one of the cameras and is therefore responsible for

[9] `http://www.gov.uk/guidance/ce-marking`
[10] `http://www.asus.com/3D-Sensor/Xtion_PRO_LIVE/`

the vision processing of the images that are recorded by that camera. To this end, the PCs also have a NVidia GeForce GT640 with 2GB dedicated memory. The robot achieves a run time of roughly 6 hours until the batteries are drained and then charges about 8 hours until they are full again. The robot is able to charge itself via a charging station that it can drive on to which allows to deploy it for several weeks continuously.

At this point, no studies about the appearance of the robot have been published and this has also not been evaluated during the course of this thesis. Judging from qualitative feedback from the two deployment sites, the shape and especially the eyes of the robot facilitate anthropomorphisation. Patients and staff at the care home often asked if the robot doesn't feel well when it is not around due to repairs and it has been kissed and hugged on several occasions. Therefore, for the remainder of this thesis, it is assumed that its appearance does not have a negative effect on the outcomes of the conducted studies. On the other hand, it has also not been investigated if it had a positive effect on participant reactions. Last, but not least, all the robots in the project (each partner has their own robot which only differ in colour) have been given names either by the researchers or the patients in the care home. Thus, the robot used for the experiments is from here on referred to as *Linda the robot* or simply *Linda*[11], whereas the robot at the care home is called *Henry*.

1.3.3 The Robot Operating System

This section only briefly describes the main principles of the Robot Operating System (ROS) to not go beyond the scope of this thesis. From the ROS webpage[12]:

> *"The Robot Operating System (ROS) is a flexible framework for writing robot software. It is a collection of tools, libraries, and conventions that aim to simplify the task of creating complex and robust robot behavior across a wide variety of robotic platforms."*

ROS itself is a middleware that is used for the communication of different programs which are called *nodes* in ROS terminology. It follows a data push approach for communication between the nodes using so-called *publishers* which make a message available on a

[11] `http://twitter.com/lindastrands`
[12] `http://ros.org`

topic and the *subscribers* receive the message from the topic as soon as it is published. An easy example is the Lidar node that grabs the scan array from the hardware and publishes it to a topic. As soon as the message is published, all the subscriber nodes get notified in a callback and receive the message. This spares the user from constantly polling for data.

In addition to this it also offers remote function calls via so-called *services* and *actions*. An action unlike a service is a non-blocking remote function call that can be interrupted, publishes feedback while it is running, and can have several different outcomes like *successful*, *preempted*, or *aborted* and return computed data at the same time. These actions are used, for example, for navigation where the argument to the so-called *goal* of the action is a tuple of (x, y, θ) for the position and orientation. The feedback given is the current position of the robot while driving and in the end it returns one of the specified outcomes based on if navigation was successful, has been preempted by the user, or failed (was aborted).

This framework allows for the seamless integration of several different components and is the most popular middleware for robotics research at the time of writing this thesis. Hence, by using ROS, standardised software components are created which increases the reusability. Therefore, the presented approach consists of several nodes using topics, services, and actions to communicate with each other which have been used on their own for other publications as well (e.g. Lightbody et al. 2015).

1.3.4 Software Management

All the approaches presented here have been implemented in ROS Indigo for Ubuntu 14.04 and are freely available for academic purposes under MIT, BSD, or GPL license (depending on the package) from the STRANDS github repositories (see Appendix B). A continuous integration server not only runs automated tests on this software but also generates Debian packages for easy installation across all the robots in the project and every interested third party. Each chapter, therefore, represents a collection of several nodes that can be used for the described purpose as a collection or separately for other applications.

—Outside of a dog, a book is a man's best friend. Inside of a dog, it's too dark to read.

Groucho Marx

2

Background and Related Work

To paint a clearer picture of the current state-of-the-art in HRSI in general and human-aware navigation in particular, this chapter describes related approaches to those presented in this thesis. This includes motion prediction, path planning, and activity modelling and recognition. The list may be incomplete as it only represents the most closely related approaches.

All robots deployed in human populated areas face the same challenges of navigating reliably and safely in the presence of humans while producing trajectories that are legible (Lichtenthäler & Kirsch 2016), avoid stress and annoyance, and are goal-directed. Some of the most commonly known examples for these kind of robots are Rhino (Burgard et al. 1999), Robox (Arras et al. 2003), Minerva (Thrun et al. 2000), Rackham (Clodic et al. 2006), Mobot (Nourbakhsh et al. 2003), and Cice (Macaluso et al. 2005) and an overview of related systems has been given by Jensen et al. (2005). This chapter shows approaches that have been used in the past to create behaviour that increases a robot's acceptance when used in these kind of scenarios and, therefore, form the background for the presented

work. Please also refer to Kruse et al. (2013) for an excellent review of this and other less related topics of human-aware navigation.

2.1 Proxemics

The most commonly used principle in HRSI is the so-called *proxemics* which is a term describing interpersonal distances and was coined by Hall (1969). This theory divides the space around a human into four distinct zones which are themselves divided into a close and far phase:

Intimate Space *0cm - 45cm* Used for love-making and wrestling, comforting and protecting

- **Close Phase** *0cm - 15cm* The vision is blurred, vocalisations are only whispers. The use of distance receptors is greatly reduced except for olfaction and sensation of radiant heat which are stepped up. Arms can encircle and the pelvis is easily accessible.
- **Far Phase** *15cm - 45cm* The hand can reach and grasp extremities of the other person but the heads, thighs, and pelvises are not easily brought into contact any more. It allows to focus the eye easily and peripheral vision includes the outline of the head and shoulders. Heat and odour of the other person's breath might be detected.

Personal Space *45cm - 1.22m* For interaction among close friends and family.

- **Close Phase** *45cm - 76cm* One can hold or grasp the other person and perceive their features without visual distortion. Additionally, the three dimensional qualities of objects like the nose are pronounced.
- **Far Phase** *76cm - 1.22m* This distance is often referred to as keeping someone at arm's length which means that this space defines the distance from outside the easy touching distance by one person to a distance where they can touch each other's fingers. The other persons features are clearly visible. Communication requires a moderate voice level. No heat be it body or breath can be perceived any more and olfaction is not present either except for strong cologne.

Social Space *1.22m - 3.7m* For interaction among strangers or acquaintances.

Close Phase *1.22m - 2.1m* Touching each other is impossible without special effort undertaken by both persons like leaning in and stretching their arms towards each other. Conversation is conducted at a normal voice level and can be overheard by others up to a distance of 6m. The visual focus extends to the nose and parts of both eyes or nose, mouth, and one eye.

Far Phase *2.1m - 3.7m* The fine details of the face like the capillaries are lost but the skin texture, hair, and condition of teeth and clothes are all readily visible. No odour can be detected.

Public Space *>3.7m* For public figures, audiences, lectures, or actors in a play.

Close Phase *3.7m - 7.6m* It is for the first time possible to take evasive or defensive action against the other person in time. The voice has to be loud but not at full volume to be heard. The vision covers the whole face but fine details of the skin or face are not visible any more. Moreover, only the white of the eyes is visible.

Far Phase *>7.6m* Subtle shades of meaning conveyed by the normal voice are lost as are the details of facial expression and movement which requires so-called "over acting" by actors in a play for example. Hence, voice, facial expressions, and movement must be exaggerated. The foveal vision takes in more and more of the other until she is entirely within the small circle of sharpest vision.

An illustration of these spaces can be seen in Figure 2.1 which shows the 4 different zones without the close and far phase.

In general interaction among strangers happens in the Public Space or beyond. Intrusions into the Personal or Intimate Space without consent are perceived as rude or even threatening and therefore create annoyance and stress. If this is unavoidable, like in crowded lifts or public transport, specific avoidance strategies are adopted like keeping ones arms to the side as close as possible, staring into the distance or the mirror in a lift, and avoiding every form of enjoyment resulting from the close contact. However, all these values are based on Caucasian and Hispanic North Americans and are highly cultural dependent. As described by Alessandra (2000) these zones can also depend on the

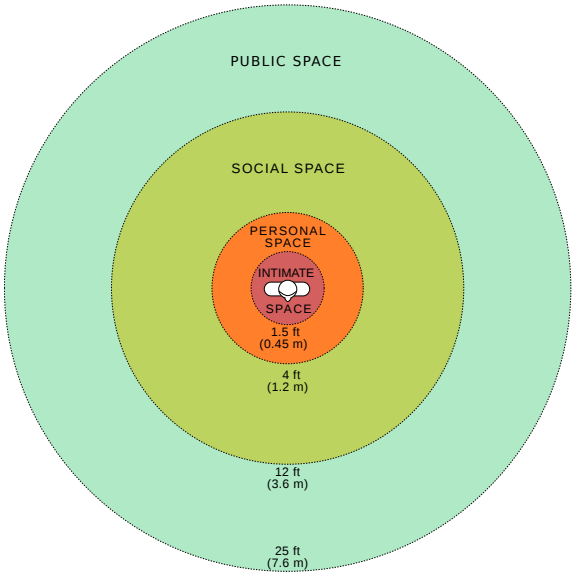


Figure 2.1: *An illustration of Proxemics showing the 4 different spaces without the close and far phase. This representation is based on the assumption that these spaces have the same dimension in all directions. Image source:* `http://en.wikipedia.org/wiki/Proxemics`

social status of the person or their age and gender as presented by Aiello & Aiello (1974). As guidelines for distances to keep in HRSI, these zones have been used and investigated heavily.

Looking at existing approaches for human-aware navigation, many works adopt the zones defined by Hall (1969) to achieve socially acceptable avoidance manoeuvres as can be seen from, e.g. Pacchierotti et al. (2005) and (2006) who investigated these distances, and most of the works on social cost functions listed below. In addition to human-aware navigation, there is also work on these distances in regards to other aspects of HRSI such as the investigation of the optimal approach distance for a robot (e.g. Torta et al. 2011) where the optimal approach distance and angle for communication between a small humanoid robot[13] and a sitting person is investigated. Torta et al. (2011) present an attractor based navigation framework that includes the definition for a *Region of Approach* which is optimal to communicate between the two agents. In the conducted experiment, where a NAO robot is approaching a sitting person from different angles, with the purpose of starting a conversation, until that person presses a button to stop the robot at a distance perceived as suitable to achieve the task, Torta et al. (2011) show that an approach from the front is preferable over an approach from the side and found that the distance at

[13]NAO - `http://www.aldebaran.com/en/humanoid-robot/nao-robot`

which the participants stopped the robot to have a conversation loosely correlates with the close phase of the social space as defined by Hall (1969). Another example, focusing on the long-term habituation effects of approach distances is the work by Walters et al. (2011). They use a standing participant and a mobile service robot instead of a NAO in an otherwise similar experimental setting as Torta et al., i.e. the robot approaches the participant from the front and is stopped via a button when it is close enough to have a conversation, and inspect the long-term effect on this most suitable approach distance. Similarly, in Chapter 3 the optimal distances for the human-aware navigation model are investigated but only in regards to the quality of the resulting model not using any form of self-assessment as was in the above approaches.

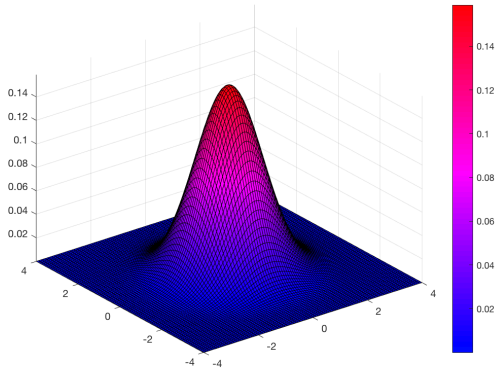
In addition to promexics being influenced by the social status and age of the person, the angle of approach of, and the familiarity to the robot as described above, the actual performance of the robot during an interaction influences the human interaction partner's proxemics preferences as well. Mead & Mataric (2015) conducted and experiment in which the human participant explained certain objects to a mobile robot via speech and gestures from a fixed location. The robot, however, altered its position during the trials. After an object had been explained, the robot would change its distance to the human before the next object explanation phase started. After each explanation, the robot signalled success or failure of understanding the explanation to the participant where the success rate depended on the distance to the human and was modelled as a normal distribution with its peak at $2.25m$ distance to the human and a standard deviation of $1.0m$. Before and after the experiment, to evaluate if the proxemics preferences of the participants changed, Mead & Mataric (2015) had the robot approach the participant until they said "stop" when they thought that the robot would be at an appropriate distance for the task. Comparing the measurements from before and after the experiment, they found that humans indeed adapt their proxemics preferences to the area of peak performance of the robot whereas in the control condition where the success rate was modelled uniformly this effect did not appear.

Despite the variability described above, proxemics is one of the most popular principles in HRI and HRSI due to the simple fact that the distance an interaction partner keeps is paramount to increase perceived safety and thereby the experienced comfort to which end proxemics provides hard thresholds as guidelines for these distances. The resulting

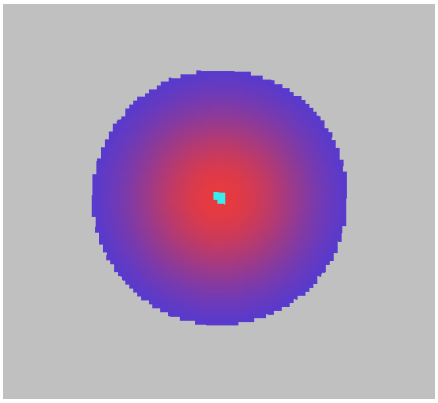
simplicity facilitates easy decision making under uncertainty based on a small set of fixed numbers. These distances, however, are only based on a very select population of North-American Caucasian and Hispanic participants and have been obtained via questionnaires and are not related to robotics. Hence, the results have to be treated with caution because they are highly culturally dependent and also change based on various different factors as discussed above. Nevertheless, proxemics is used in HRI to select interaction distances and in human-aware navigation to define the minimum distance the robot ought to keep during the encounter with a human because proxemics allows to make assumptions about the perceived safety of the executed behaviour of the robot without any additional prior knowledge. Moreover, in the case of human-aware navigation, it is often not possible to deduce the ethnicity, cultural and social background, age, gender, and other factors that influence the human's proxemics preferences from the robot's sensors during the time of an avoidance manoeuvre. Hence, proxemics offers a good initial guess of distances which are used in most cost based approaches, as described in the following, where they increase the costs of the robot's path when coming close to a human. In summary, the benefits of using Hall's (1969) model in HRI are the initial hypothesis about appropriate distances to keep during interactions without knowing anything about the interaction partner and the simple usage of hard thresholds where the variability of these values, depending on the person in question, is the major drawback that has to be considered. For avoidance manoeuvres in human-aware navigation, however, this drawback is mostly neglected due to the fact that it is impossible to acquire any knowledge about the interaction partner before or during the interaction. For these reasons, proxemics is used in many of the following approaches and also, to a certain extent, in the presented thesis. Hence, this principle ought to be kept in mind mainly as stating that a robot should keep a certain distance from a human while navigating, preferably greater than $1.22m$, to not violate the personal space.

2.2 Human-Aware Path Planning

HRSI in general and human-aware navigation and joint motion in particular, is concerned with planning paths or trajectories that are legible (Lichtenthäler & Kirsch 2016), perceived as safe, natural, and sociable leading to increased experienced comfort as defined by



(a) *2D continuous Gaussian as created according to Equation 2.1. Values used are* $x_h = y_h = 0$ *and* $\sigma = 1$.



(b) *The discretised version of the left image projected on a costmap.* x, y *represent the grid cells and* z *has been transformed into the cost value.*

Figure 2.2: *Example of a Gaussian as used for many human-aware navigation approaches.*

Kruse et al. (2013). Most current approaches focus on this increase in experienced comfort and the majority of those approaches focus on distance, i.e. proxemics, in particular to reduce stress and annoyance during the interaction. This can either be done by defining forbidden zones which the robot is not allowed to enter (e.g. Huang et al. 2010, Lam et al. 2011) which has the negative effect of the robot not being able to plan a path in confined spaces like narrow corridors, or cost function or potential fields (e.g. Sisbot et al. 2007, Tranberg Hansen et al. 2009, Kirby et al. 2009, Svenstrup et al. 2010, Scandolo & Fraichard 2011, Lu et al. 2013, Lu et al. 2014) which allow the robot to traverse through the human's personal space at high costs if required by the environment. These cost functions are mainly based on the Gaussian function

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x-x_h)^2+(y-y_h)^2}{2\sigma^2}} \tag{2.1}$$

with x, y being a continuous distribution with x_h, y_h at its centre and σ describing the standard deviation and, therefore, determining the width of the resulting distribution. This approach is used as a comparison in some of the following experiments and, therefore, shall be described in a little more detail. The resulting 2D Gaussian distribution can be seen in Figure 2.2a and represents the continuous cost function. Since the vast majority of navigation approaches do not operate in continuous space, this cost function has to

be discretised and projected to a costmap as can be seen in Figure 2.2b. This costmap represents an array of grid cells or pixels where each grid cell represents a discrete position in the world and the cost value that is associated with it. Hence, x, y are mapped to the position of the grid cells whereas z is used as the cost value for this specific cell. In order to achieve discretisation one could use the z-value of the center point of the grid cell, but in most cases the z-values are integrated in small increments within the boundaries of the grid cell to get a more precise representation because they vary non-linearly. Since a Gaussian function is continuous and will never produce 0 values, the costs are usually cut of at a distance of 3σ from the human position x_h, y_h where in practice $z \approx 0.0$. Additionally, these Gaussians can be shaped to allow for space in front of the human while walking (e.g. Ziebart et al. 2009, Kruse et al. 2010, Scandolo & Fraichard 2011). This resulting cost function is then used in path planning to allow the planner to intrude into the human's personal space if the costs for every other trajectory would be higher. Thus, these approaches integrate human-awareness with the environment by simply summing up costs for path planning.

In addition to the cost functions above, some approaches also seek to explicitly try to avoid the area behind a human to prevent discomfort (e.g. Sisbot et al. 2007, Pandey & Alami 2010, Scandolo & Fraichard 2011) which shows that the proxemics zones might not be of equal space, but also depend on the orientation of the human. In this thesis, the only assumption made is that a certain distance threshold to the human has to be kept and/or used to trigger certain behaviours like avoidance before entering the personal space. The actual unfolding of the interaction will be based on the model learned from demonstration which means that no geometric reasoning like Gaussians are required. For the sake of completeness, there are also approaches that seek to avoid as much HRSI as possible by avoiding populated areas (e.g. Diego & Arras 2011) which is an approach that is not addressed here as the target deployment area does not allow for this kind of solutions.

Naturalness, as one of the other important factors in HRSI, describes the similarity of the low-level movement of the robot (e.g. jerk, ossiclation, accelaration, etc.) to the movements of a human. In contrast to appearance and the motion of the robot's limbs, according to Kruse et al. (2013), there is no "uncanny valley" (Mori 1970) when it comes to robot navigation. Several approaches like near-minimum jerk (e.g. Arechavaleta

et al. 2008), maintaining formation for approaching a group (e.g. Althaus et al. 2004), or adapting the robot's speed instead of altering its path (e.g. Kruse et al. 2012, Lichtenthäler et al. 2013) have been proposed. Since this thesis does not aim at presenting a framework for natural robot motion with respect to speed, acceleration, and jerk, these works are only mentioned as examples towards the bigger picture of HRSI. Similarly, sociability such as social norms (e.g. walking on the right of a corridor (e.g. Helbing 1991, Kirby et al. 2009)) or using prompting to elicit a reaction and communicate ones intention (e.g. Peters 2011) are commonly investigated. This seems to be especially important for approaching humans (e.g. Butler & Agah 2001, Althaus et al. 2004, Dautenhahn et al. 2006, Koay et al. 2007, Takayama & Pantofaru 2009, Kessler et al. 2011, Mumm & Mutlu 2011) but of course also when avoiding humans (e.g. Helbing 1991, Kirby et al. 2009, Pandey & Alami 2010). This sociability will be determined by the human expert during the LfD phase in the presented thesis. Hence, no explicit reasoning is necessary.

In addition to the cost functions described above which are directly related to the position of the human, there are also other cost functions depending on the environment and the human's interaction with it that can be used for path planning. Some common examples are listed in the following, however, none of these are considered for the presented work but can be used for global path planning which can then easily be combined with the developed approach. A standard way to achieve legible and safe paths is object padding where static obstacles in the map are inflated to achieve larger obstacle clearance while navigating (e.g. Kirby et al. 2009, Svenstrup et al. 2010, Morales et al. 2015), avoid paths that are occluded by obstacles and can therefore not be observed (e.g. Chung et al. 2009), or avoid appearing out of a hidden zone and surprising the human (e.g. Sisbot 2008). Speaking about visibility, other approaches also explicitly seek to ensure that the robot only moves where it can be observed by the human (e.g. Sisbot et al. 2007, Scandolo & Fraichard 2011, Kessler et al. 2011). Lastly, the interaction between humans and/or the environment can be taken into account by avoiding paths that would lead the robot through a so-called interaction area between a group of people or between a person and an object like a television she is interacting with (e.g. Scandolo & Fraichard 2011, Rios-Martinez et al. 2012). Most of these approaches use a deliberative structure and a global path planner which makes them less flexible than fast reactive planners. Hence, these

approaches could be combined with the reactive framework presented in this thesis in a holistic navigation system.

Path planning for mobile robots itself aims at finding a safe and short path which, in the majority of cases, is done by some form of sampling based approach, or A* or Dijkstra's algorithm. HRSI does not aim to find the shortest or most energy efficient path in the way that human-unaware navigation does but tries to adhere to the numerous social norms and conventions using the cost functions introduced above and thereby arguably makes navigation in human-populated environments safer and more efficient. Path planning is either done globally over the representation of the entire environment or locally using computationally cheap sampling based approaches.

Global path planning using the previously given cost functions is very costly computation-wise, but as a deliberative approach is able generate smoother and more legible trajectories. This can even be enhanced by using temporal global planning, which predicts the motion of humans in the vicinity of the robot and therefore allows an informed choice of trajectory (e.g. Tadokoro et al. 1995, Kushleyev & Likhachev 2009, Ohki et al. 2010, Kollmitz et al. 2015). However, if there is a significant change in behaviour of the human, this plan has to be recreated which makes these approaches almost intractable for large numbers of humans as they do not scale well. An example of global planning to proactively approach a human to initiate interaction which aims at overcoming the tractability issue has been described by Carton et al. (2012) and Carton et al. (2013) who use a combination of static and dynamic path planning. The static path is generated using Bézier curves for enhanced readability of the path increasing the robot's legibility. To deal with dynamic obstacles the velocity profile of said path is adjusted. This cumulates in an optimisation problem with the goal of reaching the target location given a maximum time limit to be able to "intercept" the human interaction partner. If this cannot be achieved, a new path is generated. In order to keep this approach tractable and real-time, the optimisation is approximated via a rule-based brute force search by iteratively updating the start and end point of the curve (depending on robot and human position), and the velocity profile (depending on the position of dynamic obstacles).

Local planning, on the other hand, is computationally cheap and is mostly done using a sampling based approach, i.e. generating a set a of candidate trajectories and scoring them based on predefined metrics such as obstacle collision or cost of a grid cell, as can

be seen from a survey by Fraichard (2007) where this thesis relies on the commonly used Dynamic Window Approach (DWA) local planner by Fox et al. (1997). Local planning, however, comes at the disadvantage of being a reactive approach which can result in illegible trajectories. To overcome this, a reciprocal scheme can be assumed because according to Ducourant et al. (2005) humans also take the actions of their counterpart into consideration when planning their paths. This has been used to create avoidance manoeuvres for mobile robots which assume that the human will partake in this avoidance motion for single humans (e.g. Kluge & Prassler 2004, Van den Berg et al. 2008) and crowds (e.g. Trautman & Krause 2010, Trautman et al. 2013). The work presented here seeks to create a reactive approach that is able to predict the future unfolding of the interaction and takes the movements of the human into account when planning the robot's actions. Hence, it is closely related to the reciprocal local planning approaches mentioned here but uses a qualitative description, abstracting from the metric environment to make it robust to change.

Examples for path planning approaches that abstract from the underlying metric map are trajectory learning by Feil-Seifer & Mataric (2011) – which uses Gaussian Mixture Models created from observed trajectories to abstract from the concrete metric representation whereas Garrido & Yu (2014) used HMMs and trajectory key points. Both of these approaches use different forms of abstraction to create a general model for HRSI, but are still relying on a metric representation and are therefore very environment dependent. Heat maps are another form of abstraction that still focuses on metric space. Avrunin & Simmons (2014) used recorded trajectories of humans approaching an experimenter to create a so-called "Value Map" which can be used to represent the most commonly used paths for a specific configuration. A different form of abstraction is representing metric space via grid cells or a lattice as done by Kushleyev & Likhachev (2009), which allowed them to represent interaction in a dynamic system by a so-called time-bound lattice, using motion primitives. This interesting approach however, has only been employed for multi-robot environments and never in HRSI. All these approaches are more or less related to the presented work but still rely on metric information in some regards.

Concerning qualitative approaches to path planning, examples for qualitative route planning can be found in work by Johansson et al. (2011), Meena et al. (2012*a*), and Meena et al. (2012*b*) where dialogue is used to generate a route graph to navigate an

environment using semantic information like store and street names or house numbers. This approach can also be combined with pointing gestures to disambiguate the given instructions as shown by Buss et al. (2011). However, these approaches only generate a global path towards a goal and heavily rely on the availability of semantic information in the environment or map and cannot be used for human-aware navigation directly. Hence, all these approaches still rely on the underlying metric path planner to deal with static and dynamic obstacles along the path generated from the graph.

From all this follows that all the representations previously or currently used in HRSI path planning are based on metric space and Cartesian coordinates whereas this thesis aims at providing a qualitative model that abstracts from the actual coordinate system, environment, and metric space by representing the interaction as a sequence of states that both agents passed through. This naturally allows to incorporate the human's actions into the robot's path planning and decision making and to abstract from the environment making it robust to change.

2.3 Activity Modelling and Prediction in HRSI

Predicting the movement of humans in the vicinity of the robot is essential for all deliberative systems and the approach presented here and requires either knowledge about the environment and usual human movement (i.e. prediction based on geometric reasoning) or data from previous interactions or observations of humans (i.e. prediction based on machine learning). The works by Feurtey (2000) and Rios-Martinez (2013) give an in depth overview of the prediction of human paths, though not in the context of robotics so they do not take sensor limitations and real-time requirements into account.

Prediction based on geometric reasoning follows constraints in the usual movement of humans given a certain environment and obstacles. Tadokoro et al. (1995) use grid cells with an assigned probability – according to previous observations – of possible state transitions, meaning the likelihood of a human moving from one grid cell to the other. Ohki et al. (2010) presented a similar approach also based on grid cells and their transition probability derived from the personal space of the human. Similar to the mentioned cost functions human movement can also be predicted using social forces (e.g. Hoeller et al. 2007). In both cases, social forces and grid cells, the position of the human has to

be expressed with uncertainty which is usually achieved by using elliptical Gaussians to represent the possible future position of the human. Looking at groups of people, their motion can similarly be predicted using social force models of attraction and repulsion (e.g. Martinez-Garcia et al. 2005). All these approaches however, except for the work by Tadokoro et al. (1995) only rely on the geometry of the environment and are not based on any actual observations of human movement in it.

Prediction based on learning means the collection of data in order to build models of interactions which can be used for creation of new samples – the prediction. These approaches are highly related to the proposed probabilistic qualitative representation which works in a similar manner, but currently almost all of these approaches are exclusively based on map coordinates instead of abstract, qualitative states. Some of the more closely related works are on Motion Patterns, Feature Based Markov Decision Processes, and Short Term Trajectory Libraries. Bennewitz (2004) and (2005) use motion patterns as inputs for HMMs to not only predict the immediate future state of a human during interaction but also possible trajectories the human takes through a previously observed office environment. Ziebart et al. (2009) learn cost functions of the environment that explain previously observed behaviour and employ it in a Markov Decision Process which enables them to plan paths that balance time-to-goal and pedestrian disruption in known and unknown environments. This transferral of knowledge is due to its qualitative and abstract nature – also one of the main qualities of the proposed model. Chung & Huang (2010) observed pedestrians and created a library of short-term trajectories which they clustered to create pedestrian movement policies to predict how humans will move to avoid obstacles or each other.

All these approaches have in common that they not only rely on map coordinates or trajectories to represent the interaction, but also only represent the human side of it. Hence, none of these models allow prediction of how the robot's behaviour could influence the human's behaviour during the interaction, which is a crucial factor in HRSI. In contrast to all other approaches listed in this section, the approach presented in the following allows abstraction from metric space completely and absolutely by employing a qualitative representation that does not represent any metric information. Moreover, this model, by providing information about the movement of the two agents in relation

to one another, allows assumptions to be made about how their spatial behaviour might influence each other during the interaction, based on previous observations.

The field of modelling activities or interactions as Qualitative Spatial Relations (QSR), is of growing popularity and an overview of the most relevant applications can be found in Cohn & Renz's (2008) work. Using QSRs is a novel approach to HRSI introduced by Hanheide et al. (2012), where they use QTC to analyse the movement of a human and a robot and the work by Bellotto (2012) and (2013), where motion was generated based on hand-crafted QTC state chains on which this thesis builds. There are no examples of using QSRs for human-aware navigation or human-robot joint motion. Compared to the mentioned approaches, this thesis automatises the generation of movement behaviour, learns the models from interaction, employs them for belief generation to determine the current interaction type and the current state of the world, and uses a state-of-the-art motion planner to generate inherently safe behaviour using a flexible and fast action selection process. Moreover, none of the previous generative approaches have considered the environment, so while they are human-aware, they might have the robot drive into obstacles.

A recent PhD thesis by Schiffer (2015) describes QSRs for reasoning about object positions and possible goals for robot movement but not for the generation of low-level movement commands to control a robot during the interaction. Schiffer uses a combination of the so-called cardinal directions and distances as described by Clementini et al. (1997) which allows representing the position of an object in relation to a different object or reference frame using directions like *North-West* and distances like *close* or *far* with different granularities. While this can be used to describe the position of humans in the vicinity of the robot qualitatively, it does not represent any movement which is the most important component of HRSI. Another representation that describes the positions of an object in relation to a second object is the well known Region Connection Calculus (RCC) by Randell et al. (1992). This claculus represents relative positions using connections, i.e. *disconnected*, *edge connected*, *partially overlapping*, *tangential proper part*, *inverse tangential proper part*, *non-tangential proper part*, *inverse non-tangential proper part*, *overlapping*, and *equal*. The main problem of this representation for HRSI is that all of the states except for *disconnected* have to be prevented in robot navigation. Thus,

unifying all of the motion of human and robot under this one label is not suitable for HRSI.

Robot navigation, however, has been attempted using QSRs before. Kuipers & Byun (1991) for example describe an approach for exploration, mapping and navigation in large scale environments, and Liu & Daneshmend (2012) and Wagner & Hübner (2004) describe qualitative navigation approaches. All of these rely on landmarks in the environment in order to be able to generate movement commands for the robot and are not concerned with a reactive behaviour for human-aware navigation. Hence, there is no specific background apart from the mentioned work by Bellotto (2012) and (2013) on this form of HRSI that can be given.

When looking at recognising activities, the vast majority of approaches require the complete sequence of actions to recognise the activity reliably. This would mean, in the presented case the interaction of avoiding an oncoming human would only be classified as a pass-by encounter after it has been completed. Poppe (2010) and Ke et al. (2013) for example show a vast range of solutions for the problem of RGB-video-based activity recognition. Since *Linda* the robot is equipped with an RGB-D camera, a quick overview of RGB-D based activity recognition methods is given. This overview is by no means complete as it is not the main focus of this thesis, but paints a picture of the general field. Faria et al. (2014) and (2015) use dynamic Bayesian mixture models, whereas Wang et al. (2014) follow a bio-inspired approach of neural networks to classify activities. Sung et al. (2011) and (2012) use recordings of unstructured office environments for a similar task. More closely related to the approach presented in Chapter 3, different forms of HMMs can be used for activity recognition (e.g. Oliver et al. 2002, Wojek et al. 2006, Coppola et al. 2015, Piyathilaka & Kodagoda 2015) where Coppola et al. (2015) follow also a similar approach of using QSRs, namely a 3D variant of QTC. As mentioned earlier, all these approaches require the full sequence of actions to classify the activity. To be able to generate behaviour, however, the classification of the activity and the generation of the belief of the world has to happen in real-time using incrementally updated state chains.

There are of course many more publications on activity modelling and recognition but since this thesis focuses on HRSI and to not go beyond its scope, only the most relevant ones have been listed above.

2.4 Learning from Demonstration in HRSI

While there are examples of using data collected from human-human interaction such as the work by Weiss, Mirnig, Buchner, Förster & Tscheligi (2011) where they investigate different approach strategies of a participant to engage in a conversation with another human and derive rules and guidelines for robotic navigation, this transferral comes with certain risks. Human interaction partners react differently towards being approached by a robot compared to another human and the robot has different movement constraints depending on its physical body and motors. Hence, Weiss, Mirnig, Buchner, Förster & Tscheligi (2011) only devise guidelines because the behaviour cannot be transferred as is. To still be able to incorporate human expert knowledge and judgement, Learning from Demonstration (LfD) has emerged as a popular principle in robotics, creating policies from example state to action mappings (e.g. Argall et al. 2009). To this end, a human controls the robot to demonstrate a certain task or behaviour which the robot ought to repeat. In HRSI, however, there are only very few examples that make use of this approach. One example being Yuan et al. (2010) who use trajectories recorded while being guided through an unknown environment to achieve reliable navigation exploiting the human's knowledge about said environment. Remotely related to this, an approach of learning from observing a human performing an action is later used to create the activity model for online recognition. To acquire the set of robot actions, the robot is remote controlled by a participant similar to work by Lichtenthäler et al. (2013) where a naïve participant is tele-operating the robot to record the preferred trajectories in path-crossing situations using a similar calculus than the presented work. Thus, both of these approaches make use of the knowledge and experience of the human demonstrator where the first uses the acquired model to replay trajectories and the second to classify the recorded trajectories offline. Neither uses them for action selection or online classification.[14]

2.4.1 Wizard of Oz

An integral part of LfD is the human demonstrator. As mentioned in Section 1.2, to achieve objectives [Obj. 2.1] and [Obj. 2.2], the inverse Oz of Wizard method is used. The Wizard of Oz method in general has first been used in Human-Computer Interaction (HCI)

[14]To the author's best knowledge there are no other examples of LfD in HRSI.

Table 2.1: *Pros and cons of the Wizard of Oz study design*

Pros	Cons
• Focus on evaluation of human behaviour • Fast system response • No uncertainty in decision making • Repeatable experiment set-up due to deterministic wizard behaviour • Safe robot behaviour due to human decision making • No implementation of sensing and decision making required	• No evaluation of robot behaviour • Robot only functions as a mediator • No evaluation of possible robotic decision making processes • No evaluation of possible behaviour generation algorithms • No error detection in autonomous robotic system • Assumes technology that does not and might never exist

by Kelley (1984) to simulated system responses and, therefore, focus on the evaluation of the human behaviour. In HRI, the Wizard of Oz is a widely used principle – as can be seen from a recent survey by Riek (2012) – where the robot behaviour is controlled completely or in parts by a hidden human operator. Hence, the name Wizard of Oz has been chosen because the operator is hidden behind the metaphorical curtain, controlling the robot, having the participant belief that the system is fully autonomous. Such an experimental set-up is best suited to investigate human behaviour assuming technology that does not yet exists but might in a few years time. It therefore investigates human behaviour as the independent variable given the robot behaviour as the dependent variable (Steinfeld et al. 2009). A non-exhaustive list of pros and cons of this form of study design can be found in Table 2.1.

For the presented work, this particular form of study design is not suited because the aim is to create a system that is able to autonomously navigate in the presences of humans. Replacing the robot's behaviour generation with a human wizard could only be used to evaluate the human side of the interaction with the robot (e.g. Mead & Mataric 2015) but the work presented in this thesis focuses on the robot side of the interaction. According to Steinfeld et al. (2009), the Oz of Wizard method can be used to investigate the robot behaviour as the independent variable simulating the human as the dependent variable. This simulation can either be virtual/procedural or by using an experimenter taking the role of the participant. This methodology is especially used in cases of high risk for the human interaction partner, to further rapid development processes by omitting time consuming lengthy user studies, and to test robot behaviour prior to more exhaustive user

trials. In the presented work, this technique has been used to validate certain steps of the development before going into the evaluation phase. However, its main application lies in LfD where the Oz of Wizard, or to be more precise the Inverse Oz of Wizard as defined by Lichtenthäler et al. (2013) is used. Where the Oz of Wizard simulates the human and uses an autonomous robot system, the Inverse Oz of Wizard substitutes the robots autonomy with a naïve participant remote controlling the robot while it interacts with the simulated human. This can be used to find robot behaviour considered suitable by the participant in the situation created by the simulated human. Hence, for LfD the experiment creates the situation the robot is supposed to be taught a new behaviour for by the participant using their expert knowledge and best judgement.

—One often meets his destiny on the road he takes to avoid it.

Master Oogway, Kung Fu Panda

3

Probabilistic Qualitative Models for HRSI

Abstracting from the metric representation of a task or an environment is the main purpose of Qualitative Spatial Relations (QSR) which makes it a powerful tool for classification of and reasoning over the essence of the represented entity. Looking at HRSI, the essence of the interaction between a human and a robot can be defined as *the direction of their movement over a certain interval of time and the order in which it occurs*. An easy example is a pass-by situation where both agents approach each other on a straight line, then one or both start moving to either the left or right side while continuing their approach, pass each other, and move away towards their original goal. Using such a representation does not include any information about the environment which fulfils the objective of having an environment agnostic interaction model and therefore makes it transferable to any kind of environment or indeed even interacting agents as long as they engage in a similar encounter. However, this qualitative model still has to be descriptive enough to unambiguously classify different interactions. As mentioned in Chapter 2, most currently available representations for HRSI use some form of metric representation which can be clusters of trajectories (e.g.

Feil-Seifer & Mataric 2011), Gaussian cost models (e.g. Tranberg Hansen et al. 2009, Kirby et al. 2009, Sisbot et al. 2007, Lu et al. 2013, Scandolo & Fraichard 2011, Svenstrup et al. 2010), social forces (e.g. Martinez-Garcia et al. 2005, Tamura et al. 2012), grid cells (e.g. Tadokoro et al. 1995, Ohki et al. 2010), and so forth. These representations can either be learned or based on geometric constraints, but they all make the assumption that the environment will not change significantly. The representation introduced in this chapter abstracts from this metric space and unifies "similar" encounters, such as meeting people head on in a corridor of any shape or size, into a common model to reduce the complexity of subsequent learning approaches. Finally, this qualitative representation is used in a probabilistic model to allow for the learning and classification of different interactions and the subsequent behaviour generation based on the learned models.

A specific requirement to motion planning involving more than one dynamic agent, apart from the perceived saftey and sociability is the incorporation of the other agents intentions and movements into the robot's decision making to increase its legibility. According to Ducourant et al. (2005), who investigated human-human spatial behaviour, humans also have to consider the actions of others when planning their own. Hence, spatial movement is a reciprocal process that is as much about communication and coordination of movements between two agents – at least when moving in close vicinity to one another, e.g. entering each other's social space (Hall 1969) – as it is about the execution of trajectories. This requires the representation used to not only model the behaviour of the robot or the human, but of both in relation to each other to be able to tie a specific robot state to the state of the human and equally important to tie the human behaviour to the actions of the robot which is necessary for any form of action selection process.

To summarise, the above descriptions lead to certain requirements for a model of HRSI that need to be fulfilled in order to equip a mobile robot with an understanding of the interaction and the intention of its counterpart. As described in the objectives:

Representing the qualitative character of motions [Obj. 1.2] of both agents including changes in direction, stopping or starting to move, etc.

Representing the relevant attributes of HRSI situations [Obj. 1.3], in particular proxemics (Hall 1969), i.e. the distance between the interacting agents.

Ability to generalise [Obj. 1.4] over a number of individuals and situations.

A tractable, concise, and theoretically well-founded model [Obj. 1.5] is necessary for the representation and underlying reasoning mechanisms.

The presented work, therefore, builds on the Qualitative Trajectory Calculus (QTC) inspired by Hanheide et al.'s (2012), and Bellotto's (2012) and (2013) work using these qualitative state descriptors to model HRSI. According to a survey of human-aware navigation by Kruse et al. (2013), using these kind of QSRs for the representation of HRSI is a novel concept which is why this chapter goes into detail about the two used versions of the calculus in question and also how they are combined for the more efficient use in HRSI. This combination is employed to model distance thresholds implicitly using the probabilistic representation presented in Section 3.4.

The QTC belongs to the broad research area of qualitative spatial representation and reasoning (Cohn & Renz 2008), from which it inherits some of its properties and tools. The calculus was developed by Van de Weghe (2004) to represent and reason about Moving Point Objects (MPO) in a qualitative framework. One of the main intentions was to enable qualitative queries in geographic information systems, but QTC has since been used in a much broader area of applications (e.g. Ducourant et al. 2005, Iliopoulos et al. 2014). There are several versions of QTC, depending on the number of factors considered (e.g. relative distance, speed, direction, etc.) and on the dimensions, or constraints, of the space where the points move. The two most important variants for this work are QTC_B which represents movement in 1D and QTC_C representing movement in 2D. QTC_B and QTC_C were originally introduced in the definition of the calculus by Van de Weghe (2004) and are shown here to explain their functionality and their use in the computational model.

Chapter Contributions To summarise, the main contributions in this Chapter are the combination of two well known QTC variants into a new model that ***abstracts from the metric information of the environment and interaction [Obj. 1.1]*** by ***representing the qualitative character of the motions [Obj. 1.2]*** and therefore ***generalises to a wide variety of situations [Obj. 1.4]***. This model is able to switch between the used variants of QTC based on a distance threshold which allows to ***implicitly represent an absolute distance measure [Obj. 1.3]*** via the transition from one variant to the other. Thus, the model is able to highlight the interaction between two MPO in close vicinity to each other which, apart from reducing noise, can be used in later chapters to

Figure 3.1: *This chapter's contribution to the system shown in Figure 1.1. The interaction of human and robot is observed using external sensors (visualised by the grey cone) and the data is used to initialise and test the model. This is the basis for all following approaches that are based on the information encoded in the model.*

generate behaviour suitable for human-aware navigation. Additionally, the conceptual distance between single symbols is introduced for all used QTC variants and the definitions for legal state chains are formalised. This new, ***tractable and concise QTC model, based on the two underlying well-founded variants of the calculus [Obj. 1.5]*** is then used in a novel combination with a probabilistic representation which is trained from real-world data gathered in two experiments. Figure 3.1 shows how this contribution fits into the big picture of the overall proposed system by generating a model from the observation of a human and a robot interacting with each other.

The model is evaluated by using it to classify certain encounters from the collected data and shows good results when it comes to reliability and, therefore, proves to be suited to represent HRSI.

The work presented in this chapter has been published in two conference and one journal publication. Please refer to Appendix A.1 for the author's contributions to those papers.

3.1 The Qualitative Trajectory Calculus

In order to qualitatively describe the movement of the two Moving Point Objects (MPO) k and l, a frame of reference is needed. In the case of QTC, the relative motion of these two points is expressed using qualitative symbols, therefore, the frame of reference for point k is point l and vice-versa. To obtain a more accurate description of their relative movement, each QTC state describes a time interval $T = [t_{n-1}, t_n]$. Hence to describe the movement of k during the interval T in relation to l, the position of k_{t_n} is compared to its previous position $k_{t_{n-1}}$ and put into relation to $l_{t_{n-1}}$. For example, using the Euclidean distance $d(\cdot)$, if $d(k_{t_{n-1}}, l_{t_{n-1}}) > d(k_{t_n}, l_{t_{n-1}})$, point k moved towards l from t_{n-1} to t_n. This process

is then repeated with k and l swapped to generate the QTC state. Thereby, QTC describes the velocity vector of each point for the interval T which is later on exploited to generate robot behaviour.

In its original definition by Van de Weghe (2004), QTC operates in continuous space and assumes full observability. This assumption, however, does not hold true when using deterministic sensor data. For this reason, the interval T is from here on assumed to only consist of the to discrete time-steps t_{n-1} and t_n where the resulting QTC state of the interval T is assigned to t_n. As a consequence, for the time series $T = [t_0, t_1, \ldots, t_n]$ a list of $n-1$ QTC states $\mathbf{Q} = [Q_1, \ldots, Q_n]$ is generated.[15] This approach is commonly called uni-directional discrete QTC. In the remainder of this work, when referring to QTC, it refers to this uni-directional discrete version. For the sake of completeness, another approach to handle discrete data is bi-directional discrete QTC which uses the time points $T = [t_{n-1}, t_n, t_{n+1}]$ to calculate the relative movement of the MPO k and l at time t_n, for example $d(k_{t_{n-1}}, l_{t_n}) > d(k_{t_n}, l_{t_n}) > d(k_{t_{n+1}}, l_{t_n})$ for k approaching l. This would result in $n-2$ QTC states $\mathbf{Q} = [Q_1, \ldots, Q_{n-1}]$ for the time series $T = [t_0, t_1, \ldots, t_{n-1}, t_n]$. In work by Iliopoulos et al. (2014), the authors argue that this represents continuous QTC better than the uni-directional representation but make no case why. The uni-directional version of QTC has been chosen for this work as it simplifies the generation of QTC states, does not require t_{n+1} to calculate t_n which is only possible by knowing the movements a-priori, and still represents the interaction between human and robot well enough to reliably classify different types of encounters and generate appropriate robot behaviour as can be seen in the remainder of this work. For a full description of continuous space QTC please refer to Van de Weghe's (2004) work and to Delafontaine's (2011) work for the uni-directional and bi-directional discrete versions of QTC.

3.1.1 QTC Basic

The simplest version of QTC is called Qualitative Trajectory Calculus – Basic (QTC_B) and represents the 1D relative motion of the two MPO k and l from t_{n-1} to t_n. It uses a 3-tuple of qualitative relations $(q_1\ q_2\ q_\nu)$, where each element can assume any of the values $\{-, 0, +\}$ as follows:

[15] Assuming no additional post-processing of the generated state sequence like the collapsing of similar adjacent states as described later on.

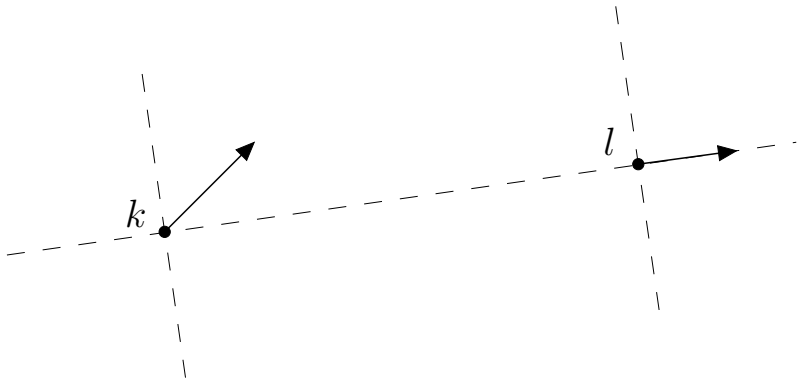


Figure 3.2: *The QTC_C double cross. The respective QTC_B and QTC_C relations for k and l are $(-+)$ and $(-+-\ 0)$. The dashed line connecting k and l is used as a directed line $\overrightarrow{k\ l}$ or $\overrightarrow{l\ k}$ to generate QTC_C states.*

q_1) movement of k with respect to l

$-$: k is moving towards l
$d(k_{t_{n-1}}, l_{t_{n-1}}) > d(k_{t_n}, l_{t_{n-1}})$

0 : k is stable with respect to l
$d(k_{t_{n-1}}, l_{t_{n-1}}) = d(k_{t_n}, l_{t_{n-1}})$

$+$: k is moving away from l
$d(k_{t_{n-1}}, l_{t_{n-1}}) < d(k_{t_n}, l_{t_{n-1}})$

q_2) movement of l with respect to k: as above, but swapping k and l

q_ν) relative speed of k with respect to l

$-$: k is slower than l
$\left|\overrightarrow{\nu^k_{t_n}}\right| < \left|\overrightarrow{\nu^l_{t_n}}\right|$

0 : k has the speed of l
$\left|\overrightarrow{\nu^k_{t_n}}\right| = \left|\overrightarrow{\nu^l_{t_n}}\right|$

$+$: k is faster than l
$\left|\overrightarrow{\nu^k_{t_n}}\right| > \left|\overrightarrow{\nu^l_{t_n}}\right|$

To create a more general representation, the simplified version QTC_{B11} is used which consists of the 2-tuple $(q_1\ q_2)$ as can be seen in Figure 3.2. Hence, this simplified version is ignorant of the relative speed of the two agents and restricts the representation to model

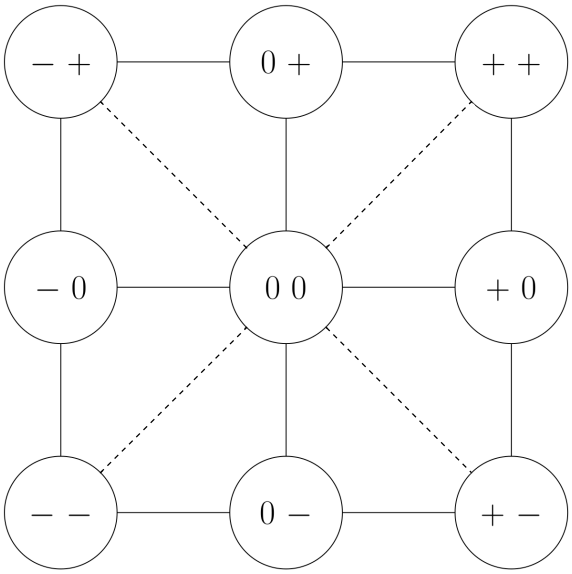


Figure 3.3: *Conditional Neighbourhood Diagram (CND) of QTC_{B11}. Given continuous observation it is impossible for k to transition from moving towards l to moving away from l without passing through the* 0*-state of being stationary even if it is only for an infinitisimal interval. Hence, whereas* + *and* − *always described intervals in time,* 0 *states can be of infisitinimal length. Also, note that due to the original formulation by Van de Weghe (2004), there are no direct transitions in the CND between some of the states that, at a first glance, appear to be adjacent (e.g.* (-0) *and* $(0-)$*). The dashed lines represent a conceptual distance of* $d_c = 2$ *(both symbols change) whereas the solid lines represent a distance of* $d_c = 1$ *(see Section 3.3).*

moving *apart* or *towards* each other or being *stable* with respect to the last distance.[16] Therefore, the state set $S_B = \{(q_1, q_2) : q_j \in \{-, 0, +\}\}$ for QTC_{B11} has $|S_B| = 3^2$ possible states and $|\tau_B| = |\{s \rightsquigarrow s' : s, s' \in S_B \wedge s \neq s'\}| = 32$ legal transitions as defined in the Conditional Neighbourhood Diagram (CND) shown in Fig. 3.3. From here on, Van de Weghe's (2004) notation $s_1 \rightsquigarrow s_2$ for valid transitions according to the CND is adopted throughout the remainder of this work. Given the continuous observation assumption, by restricting the number of possible transitions a CND reduces the search space for subsequent states, and therefore the complexity of temporal QTC sequences (see Section 3.3 for a detailed explanation).

[16]Since this representation only concerns distance, a (0 0) QTC state does not mean that the MPO are not moving but that their distance did not change. Hence, parallel movement at the same speed would still result in (0 0) or even (0 0 0) including the relative speed.

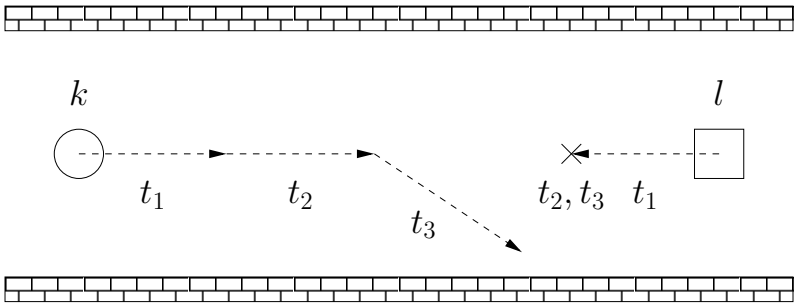


Figure 3.4: *Example of a typical passby situation in a corridor. The respective QTC_C state chain is $(--\ 0\ 0)_{t_1} \rightsquigarrow (-\ 0\ 0\ 0)_{t_2} \rightsquigarrow (-\ 0+0)_{t_3}$. t_0 is not shown because at least two discrete points in time are necessary to generate a QTC state. Hence, the state for t_1 describes the movement of k and l for $[t_0, t_1]$.*

3.1.2 QTC Double-Cross

The other version of the calculus used in the model, called Qualitative Trajectory Calculus – Double-Cross (QTC_C) for 2D movement, extends the previous one to include also the side the two points move to, i.e. *left*, *right*, or *straight*, and the absolute angle of k compared to l, with respect to the reference line $\overrightarrow{k\ l}$ or $\overrightarrow{l\ k}$ connecting them (see Figure 3.2). Figure 3.4 shows an example human-robot interaction in a corridor, encoded in QTC_C. In addition to the 3-tuple $(q_1\ q_2\ q_\nu)$ of QTC_B, the relations $(q_3\ q_4\ q_\alpha)$ are considered, where each element can assume any of the values $\{-, 0, +\}$. Let $k_{t_{n-1}}, k_{t_n}, l_{t_{n-1}}, l_{t_n}$ be discrete points in 2D space and $\overrightarrow{\nu_{k_{t_n}}} = \begin{bmatrix} x_{k_{t_n}} \\ y_{k_{t_n}} \end{bmatrix}$, $\overrightarrow{k\ l} = \begin{bmatrix} x_{l_{t_{n-1}}} \\ y_{l_{t_{n-1}}} \end{bmatrix}$ be normalised vectors with origin $\left(x_{k_{t_{n-1}}}, y_{k_{t_{n-1}}}\right)$ and similarly for $\overrightarrow{\nu_{l_{t_n}}}$ and $\overrightarrow{l\ k}$ with origin $\left(x_{l_{t_{n-1}}}, y_{l_{t_{n-1}}}\right)$ (see Figure 3.5), then:

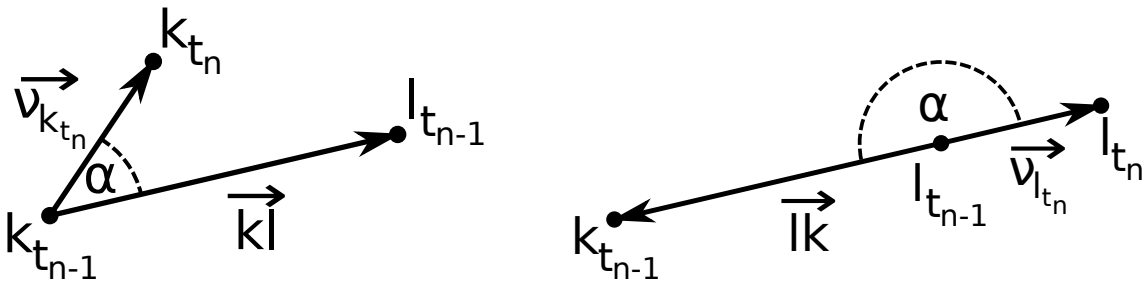
q_3) movement of k with respect to $\overrightarrow{k\ l}$ from t_{n-1} to t_n

$-$: k is moving to the left side of $\overrightarrow{k\ l}$
$\text{atan2}(\overrightarrow{\nu_{k_{t_n}}}) - \text{atan2}(\overrightarrow{k\ l}) > 0$

0 : k is moving along $\overrightarrow{k\ l}$
$\text{atan2}(\overrightarrow{\nu_{k_{t_n}}}) - \text{atan2}(\overrightarrow{k\ l}) = 0$

$+$: k is moving to the right side of $\overrightarrow{k\ l}$
$\text{atan2}(\overrightarrow{\nu_{k_{t_n}}}) - \text{atan2}(\overrightarrow{k\ l}) < 0$

q_4) movement of l with respect to $\overrightarrow{l\ k}$ from t_{n-1} to t_n: as above, but using $\overrightarrow{\nu_{l_{t_n}}}$ and $\overrightarrow{l\ k}$



(a) *Movement of k with respect to l.* ***(b)*** *Movement of l with respect to k.*

Figure 3.5: *Movement of the two MPOs k and l for $T = [t_{n-1}, t_n]$ as seen in Figure 3.2.*

q_α) the angle of k compared to the angle of l

$$
\begin{aligned}
- &: \left|\text{atan2}(\overrightarrow{\nu_{k_{t_n}}}) - \text{atan2}(\overrightarrow{k\ l})\right| < \left|\text{atan2}(\overrightarrow{\nu_{l_{t_n}}}) - \text{atan2}(\overrightarrow{l\ k})\right| \\
0 &: \left|\text{atan2}(\overrightarrow{\nu_{k_{t_n}}}) - \text{atan2}(\overrightarrow{k\ l})\right| = \left|\text{atan2}(\overrightarrow{\nu_{l_{t_n}}}) - \text{atan2}(\overrightarrow{l\ k})\right| \\
+ &: \left|\text{atan2}(\overrightarrow{\nu_{k_{t_n}}}) - \text{atan2}(\overrightarrow{k\ l})\right| > \left|\text{atan2}(\overrightarrow{\nu_{l_{t_n}}}) - \text{atan2}(\overrightarrow{l\ k})\right|
\end{aligned}
$$

Similar to QTC_B, the simplified version of QTC_C, QTC_{C21} is used. This simplified version inherits from QTC_B the ability to model if the agents are moving *apart* or *towards* each other or are *stable* with respect to the last distance and in addition is also able to model to which side of the connecting line the agents are moving. The resulting 4-tuple $(q_1\ q_2\ q_3\ q_4)$ representing the state set $S_C = \{(q_1, q_2, q_3, q_4) : q_j \in \{-, 0, +\}\}$, has $|S_C| = 3^4$ states, and $|\tau_C| = |\{s \rightsquigarrow s' : s, s' \in S_C \wedge s \neq s'\}| = 1088$ legal transitions as defined in the corresponding CND by Delafontaine (2011) shown in Figure 3.6 and detailed in Section 3.3.

For simplicity, from here on when speaking of QTC_B and QTC_C it refers to the simplified versions of QTC, i.e. QTC_{B11} and QTC_{C21} respectively, as defined by Van de Weghe (2004). The reasoning behind all this simplification is purely based on the size of the state space. Using the full QTC_C tuple $(q_1, q_2, q_\nu, q_3, q_4, q_\alpha)$ results in $3^6 = 729$ states which requires more training data and would quickly become intractable. Even though Delafontaine (2011) showed that only 305 of these states are achievable using agents moving in 2D space abiding by the known laws of physics, this state space is still large compared to the 81 states used in the simplified version of QTC_C. Moreover, the results in Section 3.6 show that even this reduced state space is descriptive enough to reliably classify the current interaction and generate appropriate behaviour.

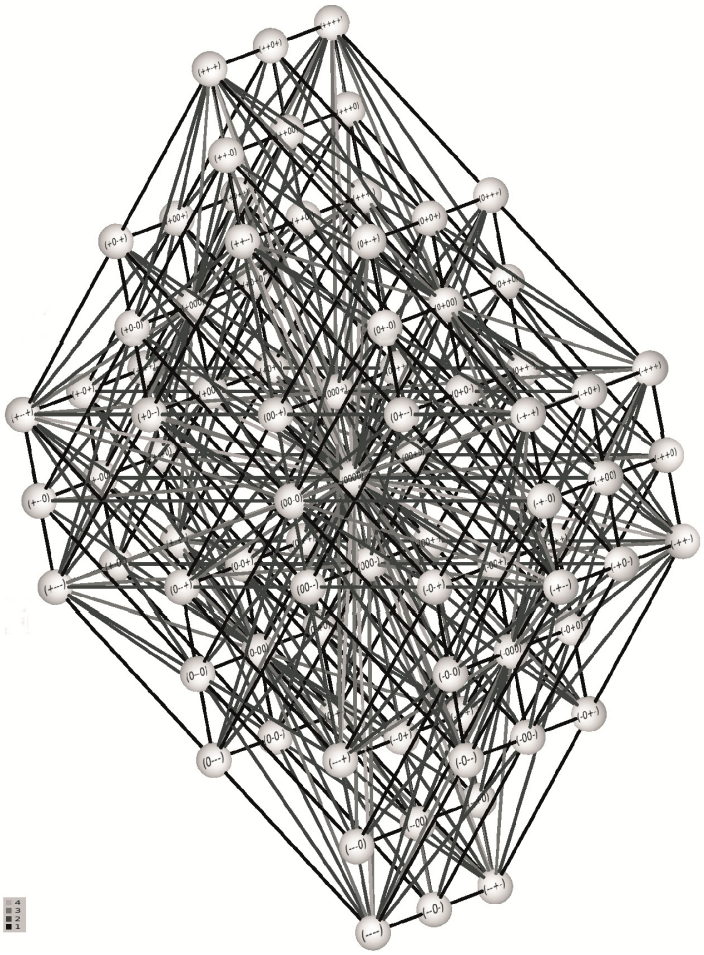


Figure 3.6: *The Conditional Neighbourhood Diagram (CND) of the simplified Qualitative Trajectory Calculus – Double-Cross (QTC_C) (image taken from (Delafontaine 2011)). Note that, similar to the CND for QTC_B, due to the original formulation of CNDs, there are no direct transitions between some of the states that at first glance look adjacent, e.g.* $(-\ 0\ +\ -)$ *and* $(-\ +\ 0\ -)$*. The grey level of the connecting lines represents the conceptual distance between the state from 1 dark grey to 4 light grey, see legend in bottom left.*

3.2 Combined Qualitative Trajectory Calculus

A crucial missing factor in order to be able to use this representation also for behaviour generation is absolute distance because proxemics, as defined by Hall (1969), is an important metric in human-human interaction that should also be considered in human-robot interaction (e.g. Pacchierotti et al. 2005). While the original definition of QTC_C can be used to identify HRSI encounters as shown in Dondrup, Bellotto & Hanheide (2014*a*), to also implicitly model a distance threshold d_s, QTC_B and QTC_C are combined into one

unified model called Qualitative Trajectory Calculus – Basic/Double-Cross (QTC_{BC}).[17] As shown in Bellotto et al.'s (2013) work, QTC_B and QTC_C can be combined using hand crafted and simplified state chains and transitions to represent and reason about HRSI. In (Dondrup, Bellotto & Hanheide 2014*b*, Dondrup, Bellotto, Hanheide, Eder & Leonards 2015), however, QTC_{BC} is formalised and automatised, ultimately enabling the model to use real-world data to learn the transitions between the two variants of QTC instead of predefining them *manually* as in Bellotto et al.'s (2013) work.

This approach is not only used for noise reduction and to highlight the interaction of human and robot in close vicinity but also to trigger state changes during behaviour generation. Using the transition from QTC_B to QTC_C a simple "approach" can change to "approach and avoid" which is paramount for the solution to this problem presented in Chapter 5. In the following, this combination of the two variants is presented as the basis for the encoding of HRSI.

The set of possible states for QTC_{BC} is a simple unification of the fused QTC variants. In the presented case the integrated QTC_{BC} states are defined as:

$$S_I = S_B \cup S_C \tag{3.1}$$

with $|S_I| = |S_B| + |S_C| = 90$ states. This results in

$$S_I = \{(q_1\ q_2\ q_3\ q_4) : q_1, q_2 \in \{-, 0, +\}\,; q_3, q_4 \in \{-, 0, +, \emptyset\}\} \tag{3.2}$$

where $q_3, q_4 \in \{-, 0, +\}\ \forall\ d(k, l) \leq d_s$ and $q_3, q_4 \in \{\emptyset\}\ \forall\ d(k, l) > d_s$.[18]

The transitions of QTC_{BC} include the unification of the transitions of QTC_B and QTC_C – as specified in the corresponding CNDs (see Figures 3.3 and 3.6) – but also the transitions from QTC_B to QTC_C

$$\tau_{BC} = \{s_b \rightsquigarrow s_c : s_b \in S_B, s_c \in S_C\} \tag{3.3}$$

[17] The combined variants are QTC_{B11} and QTC_{C21} which results in QTC_{B11C21}, from here on referred to as QTC_{BC} for simplicity.

[18] The $\emptyset$ might also be omitted in state descriptors, e.g. $(+ + \emptyset\emptyset) \Leftrightarrow (++)$.

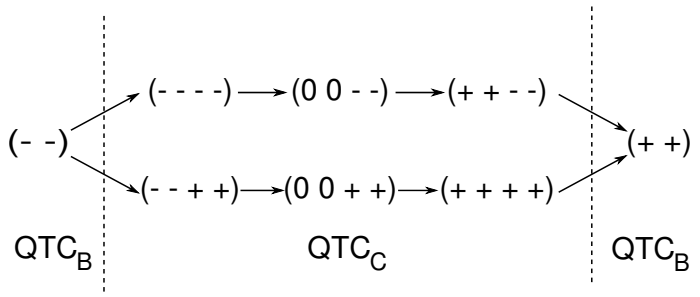


Figure 3.7: *Conceptual temporal sequence of QTC_{BC} for a pass-by encounter. From left to right: approach, pass-by on the left or right side, moving away. Dashed lines represent instants where the distance threshold d_s is crossed.*

and from QTC_C to QTC_B

$$\tau_{CB} = \{s_c \rightsquigarrow s_b : s_b \in S_B, s_c \in S_C\} \tag{3.4}$$

This leads to the definition of QTC_{BC} transitions as

$$\tau_I = \tau_B \cup \tau_C \cup (\tau_{BC} \cup \tau_{CB}) \tag{3.5}$$

To preserve the characteristics and benefits of the underlying calculus τ_{BC} and τ_{CB} are simply regarded as an increase or decrease in granularity, i.e. switching from 1D to 2D or vice-versa. As a result there are two different types of transitions:

1. Pseudo self-transitions where the values of $(q_1\ q_2)$ do not change, plus all possible combinations for the 2-tuple $(q_3\ q_4)$: $|S_B| \cdot 3^2 = 81$, e.g. $(++) \rightsquigarrow (+ + --)$ or $(+ + --) \rightsquigarrow (++)$.

2. Legal QTC_B transitions, plus all possible combinations for the 2-tuple $(q_3\ q_4)$: $|\tau_B| \cdot 3^2 = 288$, e.g. $(+0) \rightsquigarrow (+ + --)$ or $(+0 - -) \rightsquigarrow (++)$.

Resulting in

$$|\tau_{BC}| + |\tau_{CB}| = 2 \cdot (81 + 288) = 738 \tag{3.6}$$

transitions between the two QTC variants. This leads to a total number of QTC_{BC} transitions of:

$$\begin{aligned}\tau_I &= |\tau_B| + |\tau_C| + (|\tau_{BC}| + |\tau_{CB}|) \\ &= 32 + 1088 + 738 \\ &= 1858\end{aligned}$$

As for the set of states S_I, these transitions depend on the previous and current Euclidean distance of the two MPO $d(k,l)$ and d_s representing an arbitrary distance threshold

$$\tau_i \in \begin{cases} \tau_B & \text{if } d(k,l)_{t-1} > d_s \wedge d(k,l)_t > d_s, \\ \tau_{BC} & \text{else if } d(k,l)_{t-1} > d_s \wedge d(k,l)_t \leq d_s, \\ \tau_{CB} & \text{else if } d(k,l)_{t-1} \leq d_s \wedge d(k,l)_t > d_s, \\ \tau_C & \text{otherwise} \end{cases} \tag{3.7}$$

Since the actual quantitative distance threshold d_s is not explicitly included in the QTC_{BC} tuple, it is modelled implicitly via the transition between the two enclosed variants. As a result, d_s can be chosen freely or learned from observation and even altered at runtime depending on the environment or the person to interact with. This threshold is later on used to trigger certain behaviours in the robot like starting the avoidance manoeuvre or stopping to let the human pass and has to be chosen carefully during training and then adjusted to achieve the desired robot behaviour. A conceptual example of a pass-by interaction encoded in QTC_{BC} can be seen in Figure 3.7.

3.3 Conceptual Neighbourhood and Distance

The principle of conceptual neighbourhood and distance is a widely used approach in Qualitative Spatial Relations (QSR) to make assumptions about the similarity of two states. Using RCC3 an easy example which defines the 3 states of *Disconnected (dc)*, *Partially Overlapping (po)*, and *Equal (e)*. The continuous neighbourhood diagram for this representation can therefore be defined as $dc \leftrightarrow po \leftrightarrow e$ which means that dc can never transition to e without going through po in between. Assigning a cost of 1 for each

transition between states, the conceptual distance between dc and e is 2 whereas in all other cases it is 1.

Looking at the slightly more complicated case of QTC, the Conditional Neighbourhood Diagram (CND) of the variant, i.e. QTC_B in Figure 3.3 or QTC_C in Figure 3.6, can be used to find the conceptual distance d_c. In all variants of QTC, the conceptual distance depends on the number of symbols that changed from one state to the other. Looking at the case of QTC_B with $S_B = \{(q_1, q_2) : q_j \in \{-, 0, +\}\}$, it is apparent that $d_c \in [0, 1, 2]$ as there are only 2 symbols that can change. Hence, for QTC_C with $S_C = \{(q_1, q_2, q_3, q_4) : q_j \in \{-, 0, +\}\}$ the conceptual distance is $d_c \in [0, 1, 2, 3, 4]$ where in both cases $d_c = 0$ represents that the states are equal. From this definition it follows that the value the symbol changes to is irrelevant for the distance d_c because symbols jumping from $-$ to $+$ or the other way around are invalid and therefore "impossible" which holds true for continuous space. For HRSI, resulting from the discrete nature of the incoming sensor data, another conceptual distance measure ϕ is introduced in this thesis. This measurement describes the conceptual distances not between states but between symbols which is required for the observation model presented in Chapter 6 and used in the formal definitions of valid QTC state chains later on. QTC by definition is *zero-dominated* which means that each of the symbols can either transition to 0 or out of 0. This results in the symbol 0 representing either an interval of time in which the MPO was stationary, or an infinitesimal instant of time where the MPO transitioned from $-$ to $+$ by passing through the 0 state or vice-versa. Since there are 3 different symbols, the CND for each of the QTC symbols can be expressed similar to RCC3 as $- \leftrightarrow 0 \leftrightarrow +$. Hence, their distance is defined as $\phi \in [0, 1, 2]$.

In QTC_{BC} with $S_I = \{(q_1\ q_2\ q_3\ q_4) : q_1, q_2 \in \{-, 0, +\}\,; q_3, q_4 \in \{-, 0, +, \emptyset\}\}$, 4 symbols have to be considered. Therefore,

$$\phi\left(q_i, q_i'\right) \triangleq 3 \ \forall\ \left(q_i \in \emptyset \oplus q_i' \in \emptyset\right) \tag{3.8}$$

where q_i and q_i' are the symbols of two QTC states at position i and $\oplus$ symbolises an exclusive or. Otherwise, if $q_i = q_i' = \emptyset$, $\phi = 0$. As a result, for QTC_{BC} the distance is defined as $\phi \in [0, 1, 2, 3]$ where 0 indicates that the symbol did not change and 3 signifies the transition from or to $\emptyset$. The latter is necessary to prevent $\emptyset$ from replacing 0 as the

intermediate state. Since every symbol can transition to and from the $\emptyset$ in every step, it would otherwise be equivalent with the 0 representing a different meaning. In reality, the $\emptyset$ transition is only possible if $d(k_t, l_t) > d_s \wedge d(k_{t+1}, l_{t+1}) \leq d_s \vee d(k_t, l_t) \leq d_s \wedge d(k_{t+1}, l_{t+1}) > d_s$ but conceptually, the $\emptyset$ is a symbol like all the others so transitions have to be punished by a high conceptual distance value ϕ.

An example for this conceptual distance on a symbol level would be $(- - + \ 0) \rightarrow (- - - +)$ which is an illegal transition according to the original QTC definition but might happen using sensor data. The conceptual distance

$$\Phi\left(Q_1, Q_2\right) = \sum_{i=1}^{n} \phi(q_i, q_i'); \ q_i \in Q_1, q_i' \in Q_2 \tag{3.9}$$

with $n = |Q_1| = |Q_2|$ would be $\Phi((- - + \ 0), (- - - +)) = \sum(0, 0, 2, 1) = 3$.

Apart from conceptual distance, looking at the CNDs in figures 3.3 and 3.6 it becomes apparent that they do not represent a fully connected graph and therefore there are certain transitions that are illegal in QTC in addition to the transition $- \leftrightarrow +$ mentioned above. This means that each symbol has to either remain unchanged or transition to the 0-state or out of it. In addition to that, due to physical constraints of the real world, certain symbols are not allowed to change at the same time even though they do not violate the $- \leftrightarrow 0 \leftrightarrow +$ rule. One example for such an illegal transition is $(- \ 0 \ 0 \ 0) \leftrightarrow (0 - 0 \ 0)$ which means that k stops approaching l at the same time that l starts approaching k. On the first glance this seems like a reasonable real-world action but keeping the nature of QTC in mind, a 0-state is the only state that can either be an interval in time or just an instant whereas both $-$ and $+$ have to be intervals in time due to inertia and the impossibility of infinite acceleration. Thus, given continuous observation and limited acceleration of the MPOs it is physically impossible for k to stop moving at the exact instant in time that l starts to move. However, there is no limit on how many symbols can transition to or out of the 0-state at the same time only the combination of one symbol changing to 0 and another changing from 0 to $+$ or $-$ is illegal. The following rule for legal transitions can be defined

Definition 3.1. *Let Q be a QTC state and Q' the successive QTC state, then $Q \rightsquigarrow Q'$ represents a valid transition if and only if, $\exists \phi(q_i, q_i') \neq 2 \ \forall \ q_i \in Q, q_i' \in Q'$ and if for any $q_i \in Q, q_i = \{+, -\} \rightarrow q_i' = 0$, then $Q' = \{q_i' : q_i' \in \{0, q_i\}, q_i \in Q\}$.*

Apart from the creation of valid transitions based on Definition 3.1, the conceptual distance is also used for another important aspect of QTC that hasn't been mentioned so far – QTC does not represent absolute time but only a series of events in chronological order. This is achieved by collapsing equal adjacent states in the chain into one state. Thus, if $\Phi = d_c = 0$ the two states are merged into one and thereby the resulting state chain is shortened significantly and makes it impossible to infer the duration of the interaction.

Definition 3.2. *Let* $\mathbf{Q}$ *be a QTC state chain, then* $\mathbf{Q}$ *represents a* valid *state chain if and only if,* $\nexists (Q_i, Q_{i+1}) \in \mathbf{Q} : Q_i = Q_{i+1} \; \forall \; (Q_i, Q_{i+1}) \in \mathbf{Q}$.

As mentioned, this robs QTC of the ability to represent absolute time, which has benefits and detriments. The most obvious benefit is the resulting generalisability being able to represent, for example, an overtaking action in QTC_B. This action would always be described as $(-+) \rightarrow (0\ 0) \rightarrow (+-)$ regardless of the actual time it took k to overtake l. Hence, it is very straightforward to use this representation to classify all kinds of overtaking actions in all kinds of scenarios or application domains – be it one car overtaking the other, pedestrians walking, or celestial objects in outer space because it is completely agnostic of discrete speed or time values. When it comes to human-aware navigation, time and speed, however, play a vital role in the interaction, but only for behaviour generation and not necessarily for classification. The following chapters outline how this problem is tackled for the behaviour generation, while it can safely be ignored for classification as can be seen from the evaluation in this chapter.

As mentioned in the beginning, robotic systems work on discrete sensor data which is why in this case the uni-directional QTC has been chosen for all the introduced variants. Looking at above definitions, keeping in mind that 0 is also a transitional state that can only last a short instant of time, it becomes apparent that discrete sensors will most likely never produce 0 QTC states. For a moving agent, the transitional 0 state when "−" transitions to "+" or vice-versa will never be observed and even if the agent is standing still the sensor noise will always produce false movement by oscillating around its actual position. Sensor noise and oscillation around either the connecting line or the two perpendicular lines in Figure 3.2 is also the reason for not observing 0 states when one agent is moving on a straight line towards the other for example. Hence, using discrete data from real-world sensors which are subject to noise will never result in any 0 states. These

0 states, however, hold important information, e.g. there is a big difference in approaching someone head-on or moving to either side while doing so regarding communication and the need for avoidance, therefore, in the following and the remainder of the thesis a so-called quantisation factor is used, i.e. an agent has to diverge from either of the connecting lines by a certain distance for it to be counted as a non-0-state because minute oscillations can safely be ignored. Additionally, the incoming sensor data is smoothed by averaging over a certain time interval to reduce the noise. Of course, this still does not guarantee that valid state chains are produce but only that it will contain 0 states as well. In the following, a mechanism to cope with these illegal transitions is introduced and the quantisation and smoothing factor effect will be investigated in the evaluation in Section 3.5.

3.4 Probabilistic Activity Models

In order to allow reasoning and classification based on the described QTC representations, a probabilistic model and training from real-world data is required. Hence, this probabilistic representation has to be able to learn QTC state chains and the corresponding emission and transition probabilities from observed trajectories of human and robot. In the case of QTC_{BC} these transitions are defined according to Equation 3.7. Using real-world data, this representation also has to be able to deal with discrete observations instead of the desired continuous space and compensate for illegal transitions using emissions of transitional states.

Hanheide et al. (2012) proposed a probabilistic model of state chains, using a Markov Chain and QTC_C to analyse HRSI. However, illegal transitions as defined in Definition 3.1, are not taken into account which results in an illegal state chain, given the original definition of QTC_C. As a first step towards modelling HRSI using qualitative states, Hanheide et al.'s (2012) work shows that even a simple Markov Chain of illegal QTC_C state transitions is well suited to describe certain encounters in HRSI such as different types of pass-by interactions. In this section, this model is taken a step further and evolved into a Hidden Markov Model (HMM) representation of QTC_B, QTC_C, and QTC_{BC}. This enables not only the representation of actual sensor data by allowing for uncertainty in the recognition process but also to reliably classify different HRSI encounters, e.g. pass-by (see Figure 3.11a) and overtaking scenarios (see Figure 3.11b).

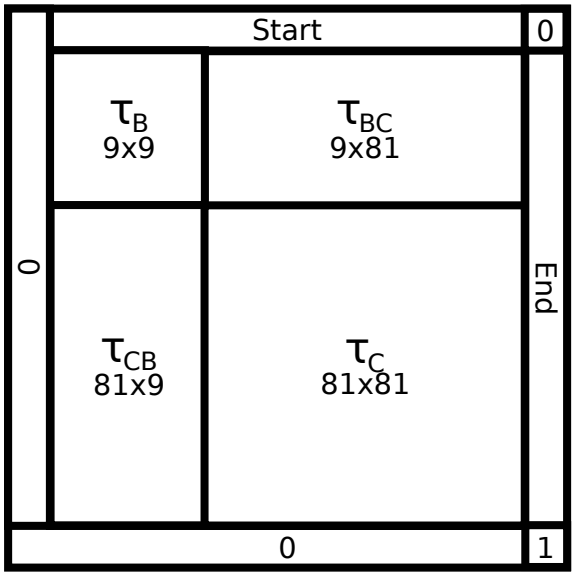


Figure 3.8: *The HMM transition matrix T for QTC_{BC}. The top left box τ_B represents the QTC_B transitions, the bottom right τ_C the QTC_C transitions. The boxes of τ_{BC} and τ_{CB} represent the transitions between the two. "Start" is the artificial start state and "End" the artificial end state. The start state can not directly transition into end and end only allows self transitions.*

A Hidden Markov Model (HMM) is defined as the 5-tuple $\mathcal{H}(T, E, I, O, S)$ and consists of the transition probabilities T, the emission probabilities E, an alphabet of input symbols I (i.e. the states of the HMM) an alphabet of output symbols O (i.e. the emissions), and the start probabilities S (Fink 2008). In the case of QTC the alphabet of input symbols is equal to the alphabet of output symbols $I = O$ and describes the QTC states of the used variant. For the presented models this amounts to $|S_B| = 9$ for QTC_B, $|S_C| = 81$ for QTC_C, and $|S_{BC}| = 90$ for QTC_{BC}. Resulting from the two alphabets having the same size, the transition and emission matrix of probabilities are both $N \times N$ matrices where $N = |I|$ is the number of states. The transition matrix T describes the transition probabilities from one state to the other meaning that $T_{i,j}$ is the transition probability of state s_i to state s_j with $s \in I$. The entries $E_{i,j}$ of the emission matrix E, on the other hand, describe the probability with which symbol o_j is emitted by state s_i with $o_j \in O$. As a result the state $(-\ 0+-)$ does not have to emit itself, but could emit any of the other states depending on the trained matrix E. Lastly, the start probabilities S describe the probability of the first symbol of an observation sequence to be producible by this specific HMM. If the first symbol in the observation sequence has 0.0 probability in S the HMM is not able to produce it and will classify it as a rejected sequence. For the presented case

of QTC, an artificial start and end state are inserted into the training and test sequences so that S is set to 1.0 for the artificial start state and to 0.0 for the remaining states. The reasoning behind this will become apparent at the end of this section.

Usually, assuming no a-priori knowledge, both T and E are initialised uniformly before the start of the training process, meaning that every transition and emission is equally likely. In the case of QTC, however, Definitions 3.1 and 3.2 can be used to restrict T to only allow legal transitions which are modelled uniformly and remove self-transitions. As a result, if this approach would have been used for Hanheide et al.'s (2012) work, the Markov Chain representation would break if the training data contains illegal transitions which, due to the deterministic nature of the sensor data, is almost a certainty. To create a valid representation, the emission matrix E is used to allow each state to also emit a state that represents a legal transition. Hence, to allow for uncertainty in the actual recognition process and deal with the resulting QTC state classification errors that arise from the discretisation of actual robot and human movement into the respective qualitative states, the "correct" emissions (e.g. the QTC_C state $(-\ 0\ +\ -)$ actually emits $(-\ 0\ +\ -)$) to occur is modelled with 95% probability and the model accounts for classification errors with 5%. For QTC_B and QTC_C this process is relatively straight forward. For QTC_{BC} on the other hand, the transition matrix T as shown in Figure 3.8 is created by combinig the transition matrices for QTC_B and QTC_C and the transitions τ_{BC} and τ_{CB} as described in Section 3.2. The emission matrix for QTC_{BC} follows the same rules as for the other QTC variants.

To represent different HRSI behaviours, the HMM needs to be trained from actual observed data (see Figure 3.9, showing an example of a trained state chain using QTC_C). For each different behaviour to be represented, a separate HMM is trained, using Baum-Welch training (Expectation Maximisation) (Fink 2008) to obtain the appropriate transition and emission probabilities for the respective behaviour. The training process allows for pseudo-transitions and emissions by "adding-one", i.e. $T = T + \frac{1}{n+1}$ with n being the number of training sequences and equivalently for the emissions E, to overcome the problem of a lack of sufficient amounts of training data and unobserved transitions therein. To create the training set the recorded data is transformed into QTC state chains of the desired variant plus the artificial start and end state. By using this start state and pseudo transitions, the model implicitly accounts for unobserved start states without having to model pseudo start

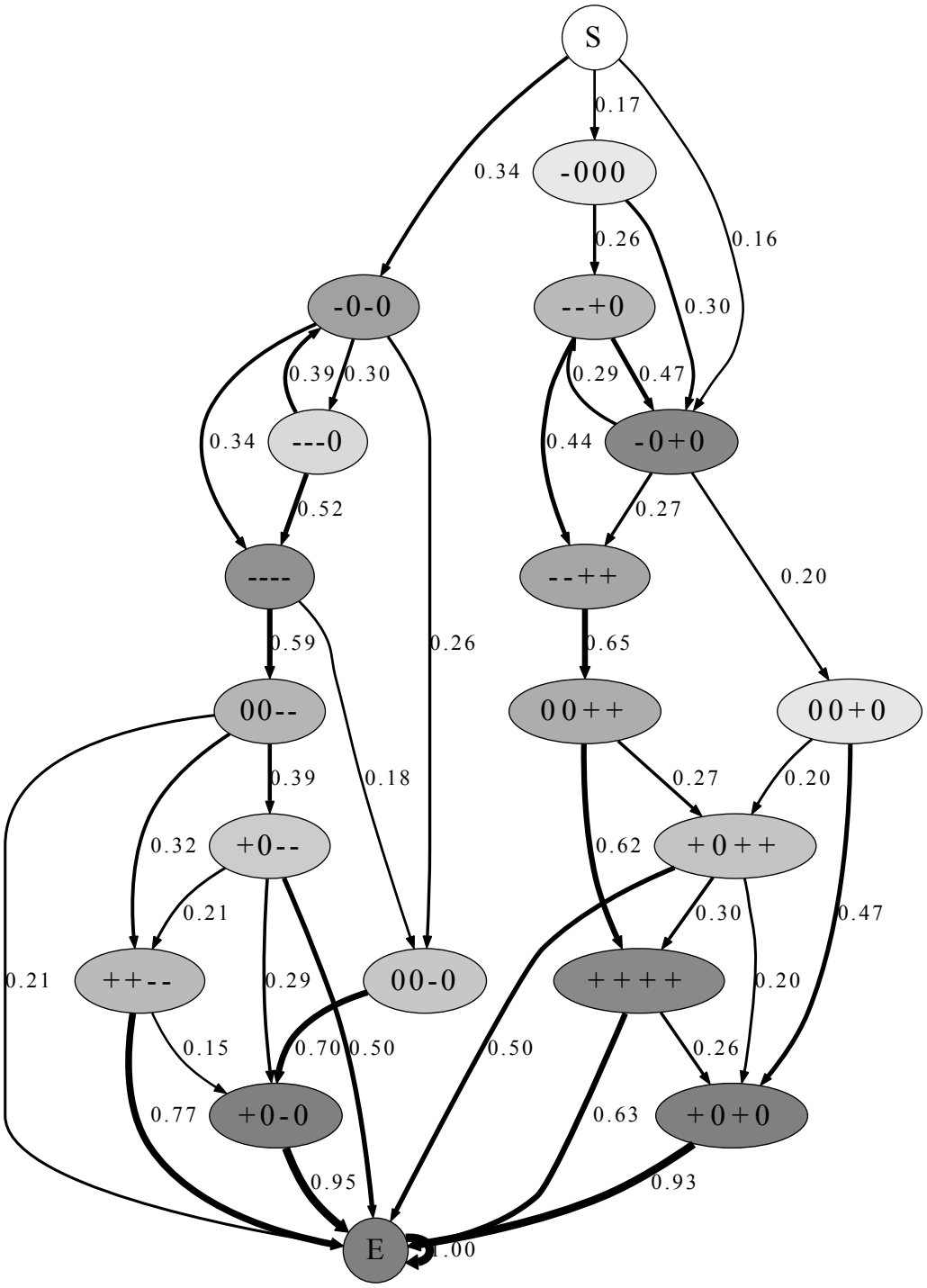


Figure 3.9: *QTC_C states for pass-by situations created by the HMM representation. Edge width and numbers represent the transition probabilities. The colour of the nodes represents the a-priori probability of that specific state to be present in any observed chain (from white = 0.0, e.g. "S", to dark grey = 1.0, e.g. "E"), i.e. its observation probability. All transition probabilities below 0.15 have been pruned from the graph, only highlighting the most probable paths within the model. Due to the pruning, the transition probabilities in the graph do not sum up to 1.0.*

states (which would be another way of tackling this problem). An example state chain for QTC_{BC} taken from Figure 3.7 would be $(- - \emptyset\ \emptyset) \rightarrow (- - - -) \rightarrow (+ + - -) \rightarrow (+ + \emptyset\ \emptyset)$ which does not contain the connecting state $(0\ 0 - -)$ and therefore contains an illegal transition $(- - - -) \not\rightarrow (+ + - -)$. Hence, after the training process, the emission for $(+ + - -)$ would also contain $(0\ 0 - -)$ to ensure valid state chains by emitting this intermediate state.

On a side note, the probabilistic model of QTC_{BC}, since it contains the full subset of QTC_B and QTC_C, is able to model both these variants as well. By setting $d_s = 0$ or $d_s = \text{inf}$ during the creation of state chains it models "pure" QTC_B or QTC_C, respectively.

3.5 Experiments

To evaluate the soundness and representational capabilities of the probabilistic model of HRSI using QTC state chains, particularly QTC_{BC} as this combines both QTC_B and QTC_C, the HMM representation is trained using real-world data from two experiments. These HMMs are then employed as classifiers to generate a comparative measurement to make assumptions about the quality of the model and the distance thresholds d_s. The two experiments both investigate the movement of two agents in confined shared spaces. The first experiment, referred to as "restaurant experiment", features the mobile service robot *Linda* and a human naïve to the goal of the experiment. The tasks were designed around a hypothetical restaurant scenario eliciting incidental and spontaneous interactions between human and robot.

The second experiment, later referred to as the "Bristol experiment", features two agents (both human) passing each other in a 2 meter wide corridor. The experimenter was dressed up as a "robot", masking her body shape, and her face and eyes were hidden behind goggles and a face mask (see Figure 3.13). This "fake robot" received automated instructions on movement direction and collision avoidance strategy from a simple program via headphones. Similar to the "restaurant experiment", the other person was a participant naïve to the goal of the experiment, but was given explicit instructions to cross the corridor with as little veering as possible, but without colliding with the oncoming agent. This second experiment does not feature a real robot but yields similar results using the presented probabilistic model, as can be seen in Section 3.6. Both experiments feature

two agents interacting with each other in a confined shared space and are well suited to demonstrate the representational capabilities of the model, showing how the approach can be effectively generalised or extended to other forms of spatial interaction.

The following sections describe the general aims and outlines of the experiments used. This is meant to paint the bigger picture of the underlying assumptions and behaviours of the robot/experimenter during the interactions and to explain some of the conditions that were compared in the evaluation. Both experiments investigated different aspects of HRSI and spatial interaction in general, which created data well suited for the analysis of the presented probabilistic model utilising QTC_{BC} and to investigate appropriate distance thresholds d_s.

3.5.1 Restaurant Experiment

This section presents a brief overview of the "restaurant experiment" set-up and tasks. Note, the original aim of the experiment, besides the investigation of HRSI using an autonomous robot in general, was finding hesitation signals in HRSI to gather feedback about the quality of the interaction for possible reinforcement learning approaches, hence the choice of conditions. Since the recorded data is appropriate for the desired evaluation of QTC, it has subsequently been used for this purpose in addition to the investigation of hesitation which has been described by Dondrup, Lichtenthäler & Hanheide (2014).

Experiment Design

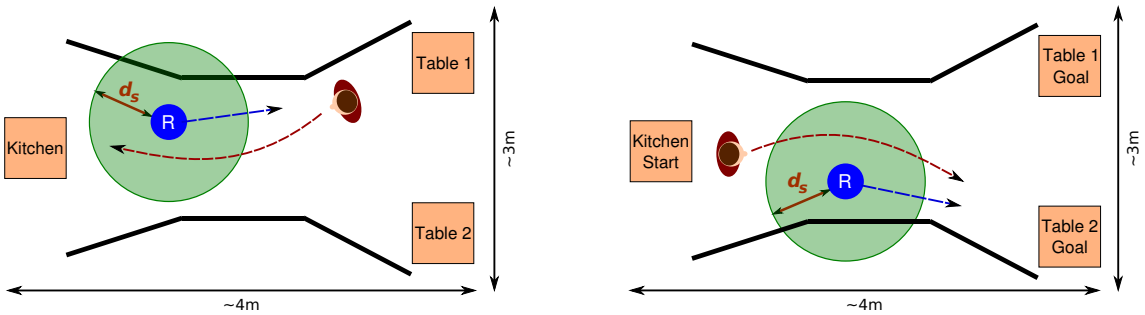
In this experiment participants were put into a hypothetical restaurant scenario together with *Linda* the robot (see Section 1.3.2). The experiment was situated in a large motion capture lab surrounded by 12 motion capture cameras (see Figure 3.10), tracking the x, y, z coordinates of human and robot with a rate of $50Hz$ and an approximate error of $1.5mm \sim 2.5mm$. The physical set-up itself was comprised of two large boxes (resembling tables) and a bar stool (resembling a kitchen counter). The tables and the kitchen counter were on different sides of the room and connected via a $\sim 2.7m$ long and $\sim 1.6m$ wide artificial corridor to elicit close encounters between the two agents while still being able to reliably track their positions (see Figure 3.11). The complete set-up was longer due to the added tables and kitchen counter plus some space for the robot and human to turn.

Figure 3.10: *The "restaurant experiment" set-up showing the robot, the motion capture cameras, the artificial corridor, and the "tables" and "kitchen counter". The shown set-up elicits close encounters between human and robot in a confined shared space to investigate their interaction.*

The specified width is taken from the narrowest point of this corridor. At the ends, the corridor widens to $\sim 2.2m$ to give more room for the robot and human to navigate as can be seen in Figure 3.11. The evaluation, however, only regards interactions in this specified corridor.

For this experiment 14 participants (10 male, 4 female) were recruited who interacted with the robot for 6 minutes each. All of the participants were employees or students at the university and 9 of them have a computer science background; out of these 9 participants only 2 had worked with robots before. No compensation was paid. The robot and human were fitted with motion capture markers on their head and shoulders to track their x, y coordinates for the QTC representation – Figure 3.12 shows an example of recorded trajectories.

The robot was programmed to move autonomously back and forth between the two sides of the artificial corridor (kitchen and tables), using a state-of-the-art local (Fox et al. 1997) and global planner (see Section 5.1 for a detailed explanation of the local



(a) *Pass-by encounter showing the human and robot trying to reach goals in opposite directions, having to avoid each other as a result.*

(b) *Overtaking encounter showing the human and robot trying to reach the same goal, having the human overtake the robot.*

Figure 3.11: *Example of the two most common types of encounters in the "restaurant experiment". The blue arrow represents the trajectory of robot ("R") and the red arrow shows the trajectory of the human (reddish figure). Experimental set-up: kitchen on the left and two tables on the right. Black lines represent the artifical corridor. The circle around the robot represents a possible distance threshold d_s.*

planner) and self-localisation using Adaptive Monte-Carlo Localisation (AMCL) (Thrun et al. 2005). Two different behaviours were implemented, i.e. *adaptive* and *non-adaptive* velocity control, which were switched at random ($p = 0.5$) upon the robot's arrival at the kitchen. The adaptive velocity control gradually slowed down the robot, when entering the close phase of the social space (Hall 1969), until it came to a complete stand still before entering the personal space (Hall 1969) of the participant. The non-adaptive velocity control ignored the human even as an obstacle (apart from an emergency stop when the two interactants were too close, approx. $< 0.5m$, to prevent injuries), trying to follow the shortest path to the goal, only regarding static obstacles. This may have yielded invalid paths due to the human blocking it, but led to the desired robot behaviour of not respecting the humans personal space. These two distinct behaviours were chosen because they mainly differ in the speed of the robot and the distance it keeps to the human. Hence, they produce very similar, almost straight trajectories which allows to investigate the effect of distance and speed on the interaction while the participant was still able to reliably infer the robot's goal. As mentioned above, this was necessary to find feedback signals in Dondrup, Lichtenthäler & Hanheide's (2014) work.

Before the actual interaction, after being introduced to the robot and the lab environment, the human participant was told to play the role of a waiter together with a robotic

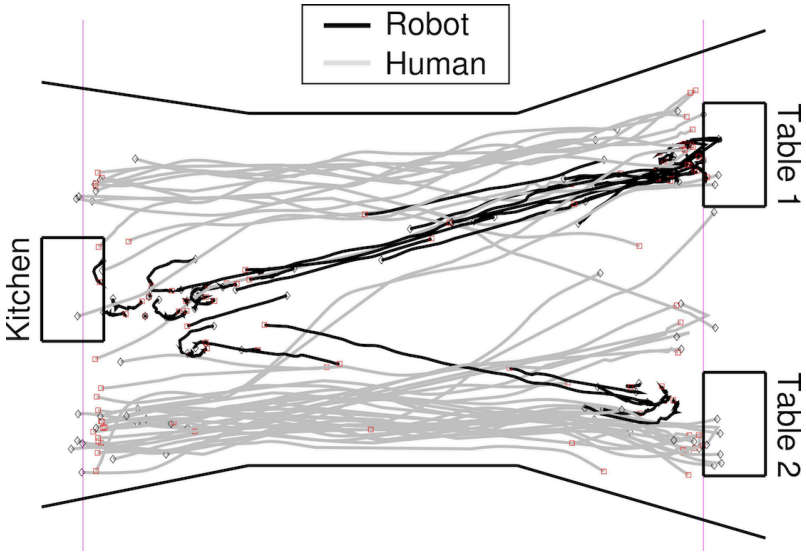


Figure 3.12: *The recorded trajectories of one of the participants (grey = human, black = robot). The rough position of the corridor walls and the furniture is also depicted. The pink lines on either side show the cut-off lines for the evaluation. The robots trajectories were not bound to the cut-off lines but to the humans trajectories' timestamps. The humans trajectories themselves might not end at the cut-off line but before due to those regions being on the outside limits of the tracking region, causing the loss of markers by the tracking system.*

co-worker for a duration of 6 minutes. Being a waiter in this particular set-up comprised the delivery of drinks (plastic bottles) from the kitchen to either table and clearing a table (bringing all bottles from the table back to the kitchen). During the experiment, the participants were instructed by the experimenter how many bottles they should deliver from the kitchen to which table and which table to clear. The experimenter followed a pseudo random approach in selecting the actions trying actively to illicit encounters between human and robot. The participants were not given any training time in order to create instantaneous feedback. This scenario allowed to create a natural form of interaction between human and robot by sending the participants from the kitchen counter to the tables and back to deliver drinks, while at the same time the robot was behaving in the described way. This task only occasionally resulted in encounters between human and robot, but due to the incidental nature of these encounters and the fact that the participants were trying to reach their goal as efficiently as possible, a more natural and instantaneous participant reaction was achieved. Figure 3.12 shows examples of these trajectories taken for one of the participants and the robot.

Figure 3.13: *The "Bristol Experiment" set-up. Corridor from the participants perspective before the start of a trial. Middle: experimenter dressed as "robot". The visual marker was attached to the wall behind the "robot" above her head.*

3.5.2 Bristol Experiment

Besides investigating general HRSI concepts, the main aim of the "Bristol experiment" was to investigate the impact and dynamics of different visual signal types to inform an on-coming agent of the direction of intended avoidance manoeuvres in an artificial agent in HRSI, hence the comparatively complex set-up of conditions.[19] For the purpose of the QTC analysis presented, however, just a specific set of conditions out of the ones mentioned in the experiment description is chosen for the evaluation.

Experiment Design

In this experiment, 20 participants (age range 19-45 years with a mean age of 24.35)[20] were asked to pass an on-coming "robotic" agent (as mentioned above, a human dressed as a robot, from now on referred to as "robot") in a wide corridor shown in Figure 3.13. The corridor was located in the Bristol Vision Institute (BVI) vision and movement laboratory, equipped with 12 Qualisys 3D-motion capture cameras. The set-up allows to

[19] A similar experiment was re-enacted by May et al. (2015) using the robot *Linda* (see Appendix A.4).

[20] No information about profession or gender were provided by the Bristol Vision Institute (BVI) who conducted this experiment.

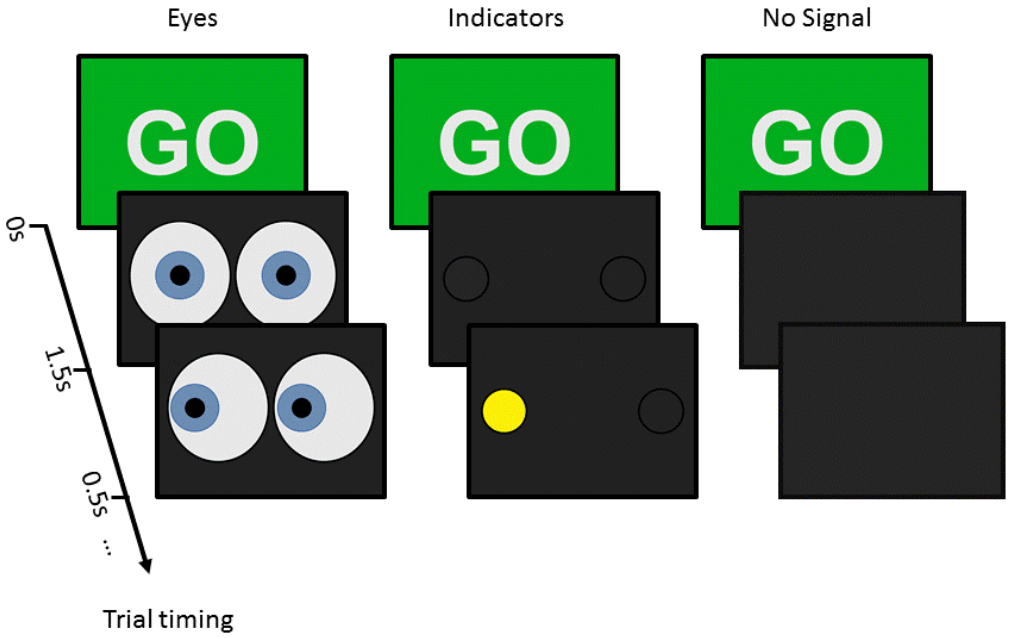


Figure 3.14: *Examples of visual signals sent by the "robot" agent. Visual signal onset occurred* $1.5s$ *after the "go" signal encouraging the participant to start walking.* $500ms$ *after the signal onset, signals could either then change to indicate a clear direction in which the "robot" would avoid the participant or remain uninformative with respect to the direction of movement of the "robot".*

track movement of motion capture markers attached to the participants and the robot in x, y, z-coordinates over an area of $12m$ (long) x $2m$ (wide) x $2m$ (high) (see Figure 3.13) with a frequency of $100Hz$ and an approximate error of $1mm$.

Participants were asked to cross the laboratory toward a target attached to the centre of the back wall (and visible at the beginning of each trial at the wall above the head of the "robot") as directly and with as little veering as possible, without colliding with the on-coming "robot", as soon as they see the "go" signal (see Figure 3.14). At the same time, the "robot" would cross the laboratory in the opposite direction, thus directly walk towards the participant. In 2/3 of the conditions, the "robot" would initiate an automated "avoidance behaviour" to the left or right of the participant that could be either accompanied by a visual signal indicating the direction of the avoidance manoeuvre or be unaccompanied by visual signals (see Figure 3.14 for the type of signals). Note that if neither robot nor participant were to start an avoidance manoeuvre, they would collide with each other approximately midway through the laboratory. A compensation of 5 GBP was paid upon completion of the experiment.

The "robot", dressed in a black long-sleeved T-shirt and black leggings, was wearing a "robot suit" comprising of two black cardboard boards (71cm high x 46cm wide) tied together over the agent's shoulders on either side with belts (see Figure 3.13). The suit was intended to mask body signals (e.g. shoulder movement) usually sent by humans during walking. To also obscure the "robots" facial features and eye gaze, the "robot" further wore a blank white mask with interiorly attached sunglasses.

A Nexus 10 Tablet (26cm x 18cm) was positioned on the cardboard suit at chest height to display a "go" signal at the beginning of each trial to inform the participant that they should start walking. The go signal was followed 1.5 seconds later by the onset of visual signals (cartoon eyes, indicators, or a blank screen as "no signal") as shown in Figure 3.14. With exception of the "no signal", these visual signals stayed unchanged in a third of the trials, and in the other two thirds of trials, they would change 0.5s later to signal the direction in which the robot would try to avoid the participant (the cartoon eyes would change from straight ahead to left or right, the indicators would start flashing left or right with a flash frequency of 2Hz). Note that no deception was used; i.e., if the "robot" indicated a direction to the left or right, it would always move in this direction. However, if the "robot" did not visually indicate a direction, it would still move to the left or right in two thirds of trials. Only in the remaining trials, the "robot" would keep on walking straight, thus forcing the participant to avoid a collision by actively avoiding it. The participants were not informed about the possible signals beforehand to achieve a more unbiased and instantaneous behaviour.

The actual/physical onset of the "robot's" avoidance manoeuvres could start 700ms before the visual directional signal was given (early), at the time of the visual direction signal (middle), or 700ms after the onset of the visual direction signal (late). These three conditions will later on be referred to as *early*, *middle*, or *late*, respectively.

3.5.3 Evaluation

The aim of the evaluation is to test the descriptive quality of the created probabilistic sequential model utilising QTC state chains in general, to evaluate possible distance thresholds or ranges of thresholds to be incorporated into the model, and to learn appropriate transitions between the QTC variants for the QTC$_{BC}$ model. To this end, the

models created from the recorded trajectories are employed as classifiers to generate comparative measurements, allowing to make statements about the representational quality of the model itself. These classifiers use a range of distance thresholds to find those values appropriate for the switch from QTC_B to QTC_C and vice-versa. The goal of this evaluation, therefore, is to evaluate the appropriateness of the presented QTC variants in combination with the probabilistic model to represent HRSI. Hence, classification is not only an important application for the model but it is also used as a tool to create a comparative measurement for evaluation.

As a reminder, the different QTC variants that are used in the following are: *i) QTC_B - 1D:* represents approach, moving away, or being stable $(-+0)$ in relation to the last position, *ii) QTC_C - 2D:* in addition to QTC_B, also includes to which side the agents are moving, left of, right of, or along $(-+0)$ the connecting line, *iii) QTC_{BC} - 1D/2D:* the combination of both according to the Euclidean distance $d(k,l)$ of the two agents and the distance threshold d_s, QTC_B (1D) when $d(k,l) > d_s$ and QTC_C (2D) when $d(k,l) \leq d_s$. The 0 states mentioned in the following are therefore instances in time when the agent was stable in its 1-dimensional and/or 2-dimensional movement.

The data of both experiments is used equally for evaluation. However, due to the different nature of the investigated effects and signals and the resulting different set-ups used, there will be slight differences in the evaluation process and therefore it will be split in two parts according to the experiments. The used model on the other hand, will be the same for both experiments to show its generalisability. The following presents the used evaluation procedures for each study.

Restaurant Experiment Two virtual cut-off lines were defined on either side of the corridor (see Figure 3.12) to separate the trajectories into trials and since only close encounters between human and robot are of interest for the evaluation, only trajectories inside the corridor were used. Out of these trajectories, 71 pass-by and 87 overtaking encounters were manually selected and two different forms of noise reduction were used to post-process the recorded data. The actual trajectories were smoothed by averaging over the x, y coordinates for $0.1s$, $0.2s$, and $0.3s$. The z coordinate is not represented in QTC. To determine 0 QTC states – one or both agents move along $\overrightarrow{k\,l}$ or along the two perpendicular lines (see Figure 3.2) – three different quantisation thresholds, i.e. $1cm$,

$5cm$, and $10cm$, were used, respectively. Only if the movement of one or both of the agents exceeded these thresholds it was interpreted as a $-$ or $+$ QTC state. This smoothing and thresholding is necessary when dealing with discrete sensor data which otherwise would most likely never produce 0 states due to sensor noise.

To find appropriate distance thresholds for QTC_{BC}, distances on a scale from pure QTC_B ($40cm$) to pure QTC_C ($3m$), in $10cm$ steps, were evaluated. The $d_s < 0.4m$ threshold represents pure QTC_B because the robot and human are represented by their centre points, therefore, it is impossible for them to get closer than $40cm$. On the other hand, the $d_s \geq 3m$ threshold represents pure QTC_C because the corridor was only $\sim 2.7m$ long. In the following these specific distance ranges will be denoted as $d_s = QTC_B$ for $d_s < 0.4m$ and $d_s = QTC_C$ for $d_s \geq 3m$, respectively.

The evaluation includes pass-by vs. overtake, passing on the left vs. right, and adaptive vs. non-adaptive velocity conditions.

Bristol Experiment Following a similar approach as described above, the recorded data is split into separate trials, each containing one interaction between the "robot" and the participant. To reduce noise caused by minute movements before the begin and after the end of a trial, data points from before the start and after the end of the individual trial were removed by defining cut off lines on either end of the corridor, only investigating interactions in between those boundaries. Visual inspection for missing data points and tracking errors by the author during post-processing yielded 154 erroneous datasets (too few data points to show the actual avoidance) out of the 1439 trials in total which were excluded from the evaluation. Similar to the restaurant data set, three different smoothing levels $0.00s$, $0.02s$, and $0.03s$ were applied. Also four different quantisation levels, $0.0cm$, $0.1cm$, $0.5cm$, and $1cm$ were used to generate QTC 0-states.[21] Unlike the "restaurant experiment", one of the smoothing and quantisation combinations, i.e. $0.0s$ and $0.0cm$, represents unsmoothed and unquantised data. This was possible due to a higher recording frequency and a less noisy motion capture system.

The evaluation includes distances on a scale from pure QTC_B ($40cm$) to $3m$, in $10cm$ steps. As mentioned above, the two agents are represented by their centre points which

[21] Due to the higher recording frequency of $100hz$ the smoothing and quantisation values are lower than for the restaurant experiment.

makes it impossible form them to be closer together than $d_s < 0.4m$. To stay in line with the other experiment, the evaluation includes distances of up to $3m$, but since the corridor had a length of $12m$, a pure QTC_C representation ($d_s = \text{inf}$) was added for comparison. Similar to the "restaurant experiment", in the following these specific distance ranges will be denoted as $d_s = QTC_B$ for $d_s < 0.4m$ and $d_s = QTC_C$ for $d_s = \text{inf}$, respectively.

Since overtaking scenarios were not part of the experimental design, these could not be evaluated. Hence, the evaluation includes passing on the left vs. right and indicator vs. no indicator separated according to their timing condition (i.e. early, middle, late), and early vs. late regardless of any other condition where indicator comprises the cartoon eyes as well as the flashing dot.

Statistical Evaluation To generate the mentioned comparative measurement to evaluate the meaningfulness of the representation, the previously described HMM based QTC_{BC} representation was used as a classifier comparing different conditions. Using this measurement, assumptions about the quality and representational capabilities of the model itself can be made.

For the classification, k-fold cross validation with $k = 5$ was used which resulted in five iterations with a test set size of 20% of the selected trajectories. This was repeated ten times for the "restaurant experiment" and 4 times for the "Bristol experiment"[22] – to compensate for possible classification artefacts due to the random nature of the test set generation – resulting in 50 and 20 iterations over the selected trajectories, respectively. Subsequently, a normal distribution was fitted over the classification results to generate the mean and 95% confidence interval and make assumptions about the statistical significance. Being significantly different from the null hypothesis (H_0; $p = 0.5$), the evaluations presented in the following section imply that the model is expressive enough to represent the encounter it was trained for. This validation procedure was repeated for all smoothing and quantisation combinations.

[22]The number of repetitions for the "Bristol experiment" is lower due to the higher number of data points and the resulting increase in computation time and decrease in feasibility.

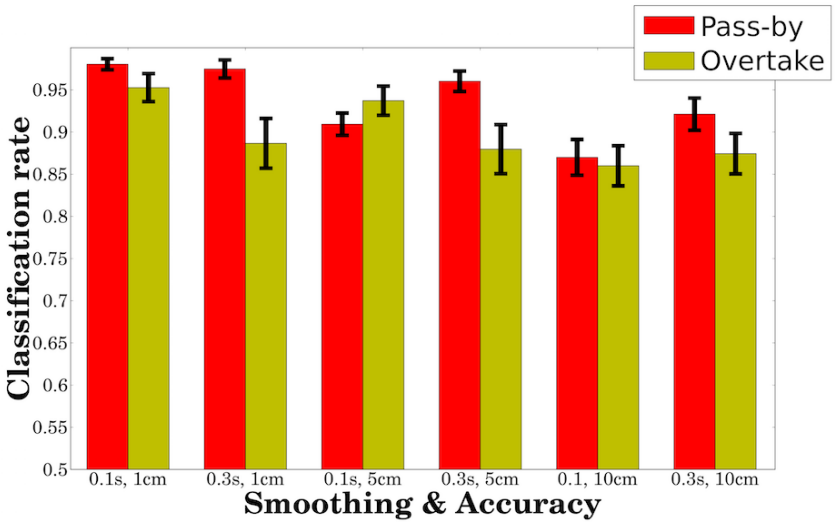


Figure 3.15: *The true positive classification rates for the different smoothing times and accuracy thresholds. Red represents the pass-by class, yellow represents the overtaking class. The error bars represent the 95% confidence interval. Note: To better visualise the small differences between the results, the bottom line does not represent 0 but $\mu = 0.5$ (the null hypothesis for the two-class problem).*

3.6 Results

A comparison of the pass-by vs. overtake scenario using QTC_C was used to evaluate if the HMM based QTC representation is able to model HRSI in general. Using the above mentioned validation process on the "restaurant" study data, classification rates from $\mu = 0.8700$ to $\mu = 0.9804$ were achieved for the pass-by class and $\mu = 0.8600$ to $\mu = 0.9527$ for the overtaking class (see Figure 3.15). The best classification rate was produced by the lowest filter settings with a (x, y) position averaging over $0.1s$ and an accuracy threshold of $1cm$. In this case, the model achieved a classification rate of $\mu = .9804 \pm .0066$ with a standard deviation of $\sigma = .0193$ for the pass-by case and a classification rate of $\mu = .9527 \pm .0166$ with a standard deviation of $\sigma = .0488$ for the overtaking case. The specified mean intervals represent the 95% confidence intervals on said mean value and the results can therefore be interpreted as being significantly different from the null hypothesis. This means that the two classes are distinguishable because the confidence interval is well above $\mu = 0.5$.

The general classification accuracy seems to be loosely correlated to the spatial accuracy used to determine which states count as 0-states in the QTC_C representation, therefore, indicating that this is the largest factor of data loss in the automatic construc-

tion of QTC_C states. Nevertheless, even the $10cm$ accuracy results are still significantly different from the null hypothesis and therefore support the hypothesis that the HMM approach is well suited to model QTC_C state sequences and classify new data.

To verify the effectiveness of the probabilistic representation of QTC_{BC} state chains given different distance thresholds, the described classifiers were used to generate a comparative measure by evaluating the classification rate for the two experiments. Pass-by vs. overtake and adaptive vs. non-adaptive velocity control is evaluated in the "restaurant experiment", passing on the left vs. passing on the right in both, and early vs. late and indicator vs. no indicator in the "Bristol experiment". Figure 3.7 shows a conceptual example of a resulting QTC_{BC} representation of a pass-by encounter which is the most dominant in both experiments.

3.6.1 Results of Restaurant Experiment

Table 3.1a shows the minimum and maximum classification rates (μ) for the general pass-by vs. overtaking case and the respective QTC_{BC} thresholds (d_s). For the majority of the different smoothing levels (7 out of 9), the best classification results were achieved using distance thresholds of $QTC_B \leq d_s \leq 0.6m$. The best result $\mu = 0.98$ was achieved using a distance of $d_s = 2.2m$ and smoothing values of $0.3s$ and a quantisation value of $1cm$. Even though the lowest and highest classification rates for the different smoothing and quantisation levels are significantly different from each other, they are all significantly different from H_0 as well. The overall worst results have been achieved using a smoothing value of $0.1s$ and a quantisation level of $10cm$ as can also be seen in Figure 3.15. Using this combination yields the highest number of 0-states compared to all the other combinations due to the fact that for a movement to be recognised it has to diverge from the previous position by $10cm$ which is very unlikely to happen in $0.1s$.

The comparison of passing on the left vs. passing on the right, is shown in Table 3.1b. All of the results show bad classification rates if $d_s \leq 0.7m$, and high classification results for values of $d_s \geq 0.9m$. Fig. 3.16a shows two typical results from the "restaurant experiment" using the lowest and highest smoothing levels. The higher smoothing and quantisation value combination, and the resulting reduced noise, show a steeper incline in classification rates than the lowest value combination, which can be seen from the smaller

Table 3.1: *Classification results "restaurant experiment"*
The mentioned confidence intervals represent the boundary cases and all the others can be considered lower. Bold face numbers are mentioned in text.

(a) *Head-on vs. Overtake. Maximum* 95% *confidence intervals* ($p < 0.05$) *for min and max classification results:* min: 0.0209, max: 0.0182

Smoothing		0.1s		0.2s		0.3s	
	Res.	μ	d_s	μ	d_s	μ	d_s
1cm	min	0.90	0.7	0.89	1.0	0.91	0.7
	max	0.97	QTC$_C$	0.96	**0.6**	**0.98**	**2.2**
5cm	min	0.84	0.8	0.88	0.8	0.87	0.7
	max	0.92	**0.5**	0.97	**QTC$_B$**	0.94	**QTC$_B$**
10cm	min	**0.70**	2.0	0.79	1.2	0.79	0.9
	max	**0.82**	**QTC$_B$**	0.87	**0.5**	0.89	**0.4**

(b) *Head-on: Left vs. Right. Maximum* 95% *confidence intervals* ($p < 0.05$) *for min and max classification results:* min: 0.0221, max: 0.0182

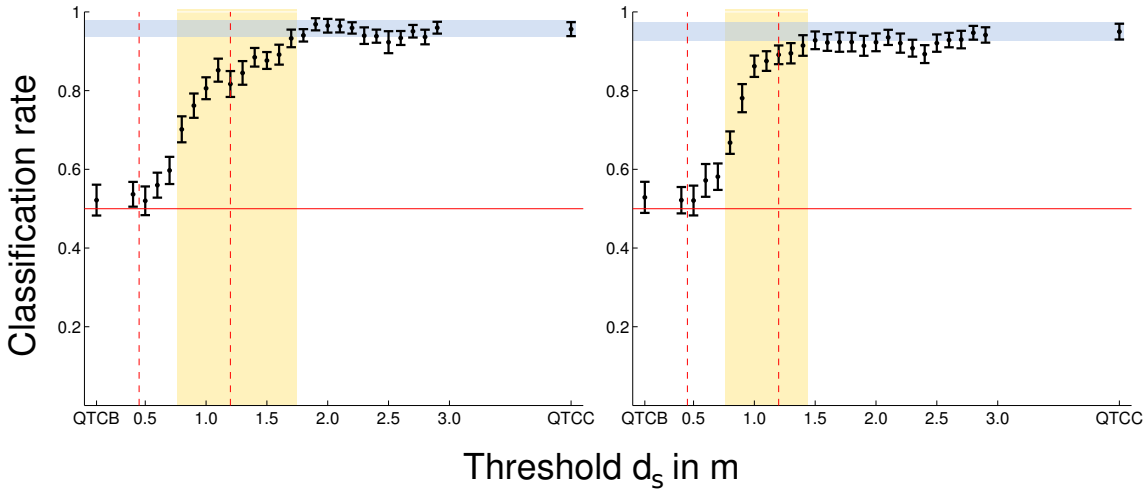
Smoothing		0.1s		0.2s		0.3s	
	Res.	μ	d_s	μ	d_s	μ	d_s
1cm	min	**0.50**	**QTC$_B$**	0.58	QTC$_B$	0.52	QTC$_B$
	max	**0.97**	**1.9**	0.95	2.4	0.96	2.3
5cm	min	0.41	QTC$_B$	0.41	QTC$_B$	0.49	QTC$_B$
	max	0.90	2.9	0.93	2.8	0.94	2.9
10cm	min	0.50	QTC$_B$	0.43	QTC$_B$	**0.52**	**0.5**
	max	0.92	QTC$_C$	0.90	1.2	**0.95**	**QTC$_C$**

(c) *Head-on: Adaptive vs. Non-Adaptive. Maximum* 95% *confidence intervals* ($p < 0.05$) *for min and max classification results:* min: 0.0202, max: 0.0251

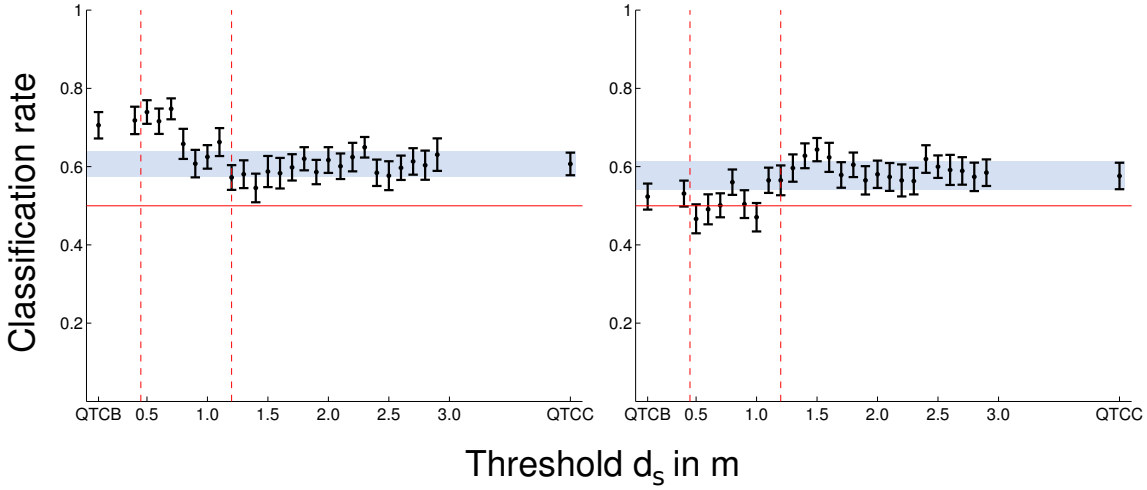
Smoothing		0.1s		0.2s		0.3s	
	Res.	μ	d_s	μ	d_s	μ	d_s
1cm	min	**0.46**	**1.4**	0.48	1.8	0.47	0.5
	max	**0.66**	**QTC$_B$**	0.60	0.8	0.64	1.5
5cm	min	0.52	1.0	**0.55**	**1.4**	0.54	1.3
	max	0.69	1.5	**0.75**	**0.7**	0.72	0.5
10cm	min	0.46	1.2	0.49	0.8	**0.59**	**1.6**
	max	0.60	1.8	0.64	1.0	**0.74**	**0.7**

yellow area in the right half of Fig. 3.16a. Nevertheless, in all of the cases, a sudden increase in performance (jumping from $\mu \approx 0.5$ to $\mu > 0.8$) can be seen at $0.9m \leq d_s \leq 1.2m$.

The third case, adaptive vs. non-adaptive robot behaviour in pass-by encounters, is shown in Table 3.1c. This behaviour did not result in different trajectories during the interaction, but only differed in the time it took the robot to traverse the corridor. Due



(a) *Classification results for pass-by passing on the left vs. right, lowest and highest smoothing parameters (see bold entries in Tab. 3.1b for min an max results). Left* $1cm$ *and* $0.1s$ *smoothing, right* $10cm$ *and* $0.3s$ *smoothing. The yellow, vertical area shows possible* d_s *where the left boundary represents the first distance* d_s *at which the two classes can be distinguished reliably and the right boundary shows the first value of* d_s *for which the classification results are not significantly different from* QTC_C *any more.*



(b) *Classification results for pass-by adaptive vs. non-adaptive. Left:* $5cm$ *and* $0.2s$ *smoothing, right:* $1cm$ *and* $0.3s$ *smoothing.*

Figure 3.16: *"Restaurant experiment" classification results. Dot: mean value, errorbar:* 95% *confidence interval, solid red line:* H_0*, left dashed red line: intimate space, right dashed red line: personal space according to the definition of Hall (1969). The blue, horizontal area represents the* 95% *confidence interval of pure* QTC_C *for comparison.*

to the Definition 3.2 QTC it is not able to represent absolute time, which makes it hard to classify these two behaviours accordingly. The best results for each quantisation level were achieved at distances of $QTC_B \leq d_s \leq 0.7m$, all lying on the diagonal of Table 3.1c. Since time is a crucial factor in this condition, it is very dependent on the right smoothing value combination. Figure 3.16b shows two exemplary results. The left hand side depicts the best classification result with classification rates of up to $\mu = 0.748$ for $d_s = 0.7m$. The right hand side shows the results for a smoothing level that did not yield the best results for low but medium distance threshold of $d_s = 1.5m$ with a classification rate of $\mu = 0.643$.

3.6.2 Results of Bristol Experiment

Table 3.2 shows the evaluation of passing on the left vs. passing on the right using QTC_{BC} for the "Bristol experiment". The *early* condition, shown in Table 3.2a, shows its lowest classification rates for $QTC_B \leq d_s \leq 0.6m$, and the first occurrence of the highest classification rates (up to 1.0) for $1.6m \leq d_s \leq 2.3m$. Reaching classification rates of 1.0 was made possible by the increase in training data for the "Bristol Experiment". Similar to the *early* condition, the *late* condition, shown in Table 3.2b, shows its lowest classification rates for $QTC_B \leq d_s \leq 0.6m$, due to the missing 2D information, and the first occurrence of the highest classification rates for $1.5m \leq d_s \leq 2.4m$. In both cases, 50% of the lowest classification rates have been generated using pure QTC_B, whereas all of the highest classification rates have been reached without using pure QTC_C. Classification rates of 1.0 with $p < 0.05$ are reached in 94% of the cases in the *early* condition and 100% in the *late* condition, using values of $d_s \geq 1.6m$ and $d_s \geq 1.5m$ respectively. Figure 3.17a shows the two unsmoothed cases for early and late. The *middle* condition is not shown here as it does not differ significantly from the two boundary cases.

Figure 3.17b shows the results for the comparison of the *early* and *late* condition. As can be seen form the figure, the two conditions can be distinguished for distances of $0.8m \leq d_s \leq 1.3m$, regardless of the actual smoothing values used. The majority of the values are not significantly different from H_0 except for the mention range of d_s. The influencing factor here is the actual minimum distances the participants kept to the experimenter in either condition. Fitting a normal distribution over the minimum distances kept in the

Table 3.2: *Classification results "Bristol Experiment":* Left vs. Right
The mentioned confidence intervals represent the boundary cases and all the others can be considered lower. Bold face numbers are mentioned in text.

(a) *Early condition. Maximum* 95% *confidence intervals* ($p < 0.05$) *for min and max classification results:* min: 0.0333, max: 0.0066

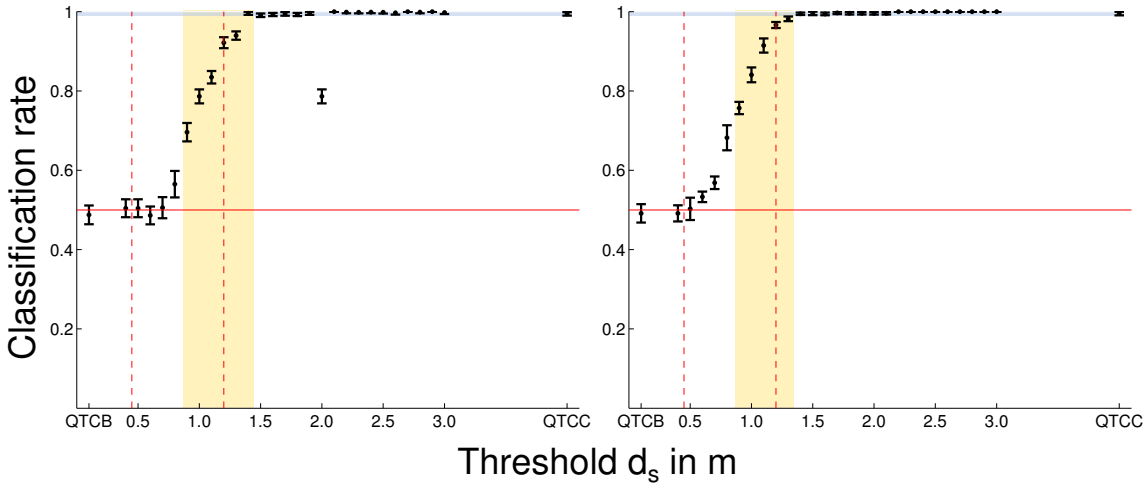
Smoothing		0.0s		0.02s		0.03s	
	Res.	μ	d_s	μ	d_s	μ	d_s
0cm	min	0.49	0.6	0.50	**0.6**	0.52	$\mathbf{QTC}_B$
	max	1.0	2.2	1.0	**2.3**	1.0	1.9
0.1cm	min	0.48	0.4	0.47	$\mathbf{QTC}_B$	0.52	$\mathbf{QTC}_B$
	max	1.0	2.2	1.0	1.9	1.0	**1.6**
0.5cm	min	0.47	0.4	0.50	0.4	0.54	$\mathbf{QTC}_B$
	max	1.0	2.0	1.0	1.6	1.0	**1.6**
1cm	min	0.58	0.4	0.47	$\mathbf{QTC}_B$	0.52	$\mathbf{QTC}_B$
	max	0.99	2.0	1.0	**1.6**	1.0	1.7

(b) *Late condition. Maximum* 95% *confidence intervals* ($p < 0.05$) *for min and max classification results:* min: 0.0327, max: 0.0036

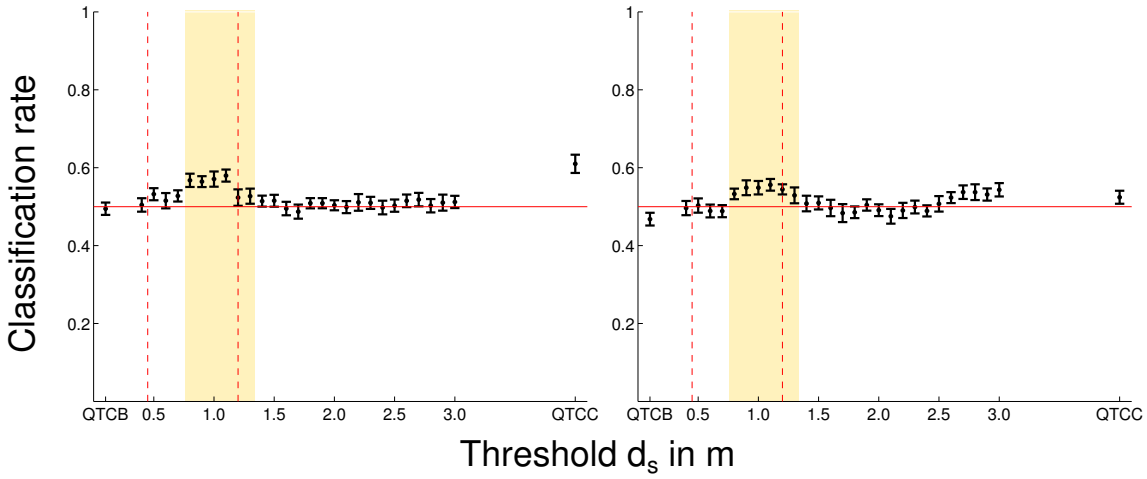
Smoothing		0.0s		0.02s		0.03s	
	Res.	μ	d_s	μ	d_s	μ	d_s
0cm	min	0.49	$\mathbf{QTC}_B$	0.49	$\mathbf{QTC}_B$	0.51	0.5
	max	1.0	2.3	1.0	**1.5**	1.0	1.6
0.1cm	min	0.53	$\mathbf{QTC}_B$	0.52	0.5	0.54	**0.6**
	max	1.0	2.3	1.0	**2.4**	1.0	1.6
0.5cm	min	0.56	0.5	0.51	$\mathbf{QTC}_B$	0.51	$\mathbf{QTC}_B$
	max	1.0	2.0	1.0	2.0	1.0	1.6
1cm	min	0.54	0.4	0.49	$\mathbf{QTC}_B$	0.47	0.5
	max	1.0	2.0	1.0	**2.4**	1.0	1.6

early and late condition yielded a significant difference ($p < 0.05$): *early*: $0.98m \pm 0.02$, *late*: $0.92m \pm 0.02$, but the actual total difference between the mean values in the minimum distances for early and late is only $0.06m$; the slightly increased reaction time of $1.4s$ in the *early* compared to the *late* condition is the determining factor for this difference. Both these facts explain the improved classification rate in the mentioned range $0.8m \leq d_s \leq 1.3m$. As above, the *middle* condition is not shown because it does not significantly differ from the two other conditions. The minimum distances kept by the participant in the middle condition are neither significantly different from the early nor the late condition. Hence, classification cannot be achieved.

The results for the comparison of the *indicator vs. no indicator* conditions are very parameter dependent when it comes to smoothing and quantisation. Figure 3.18a shows

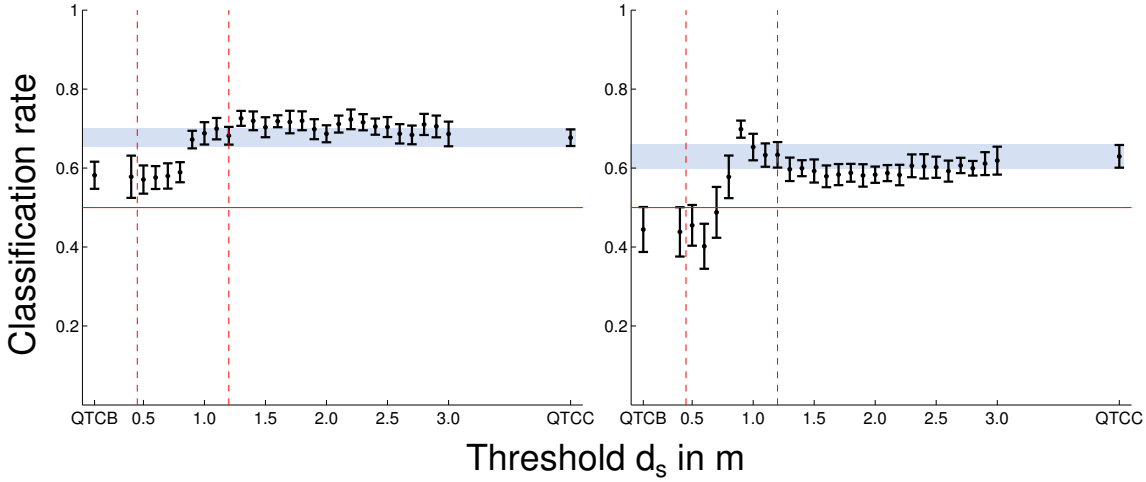


(a) *Results for the* left *vs.* right *condition using unsmoothed data.* Left: *early condition,* right: *late condition. Significant classification results have been achieved for values* $d_s > 0.8m$ *regardless of the actual condition and reach optimal results for the classification using* $d_s \approx 1.5m$*, see yellow, vertical area. The artefact at* $2.1m$ *can be explained by the physical set-up of the experiment, i.e. the corridor width. The increased confidence interval at* $2.1m$ *is due to the "robot" getting tangled up in the curtains once. Blue, horizontal area:* 95% *confidence interval of pure* QTC_C *for comparison.*

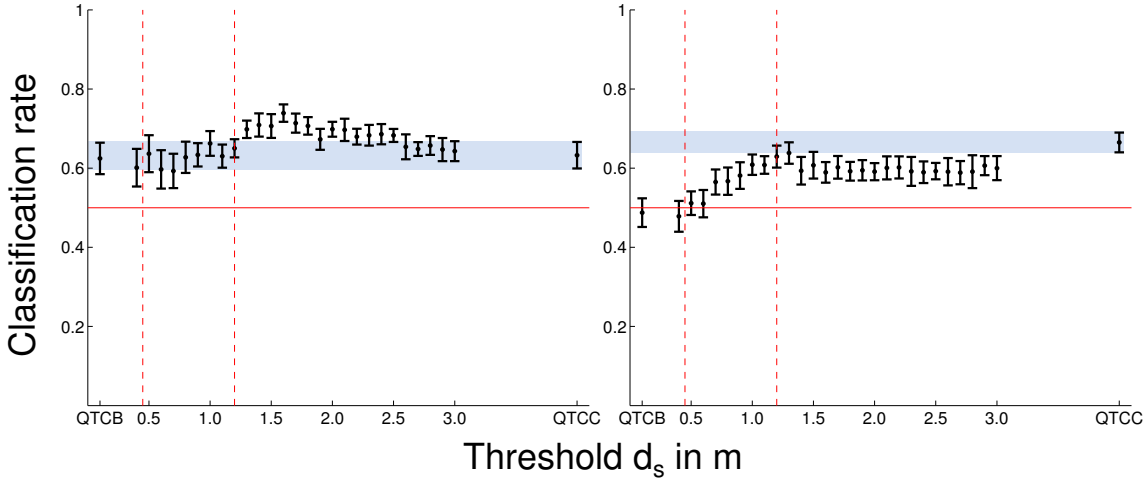


(b) *Results for the* early *vs.* late *condition.* Left: *unsmoothed data,* right: *highest smoothing values, i.e.* $1cm$ *and* $0.03s$*. Significant classification results have been achieved for values* $0.8m \leq d_s \leq 1.3m$ *regardless of the actual smoothing values, see yellow, vertical area. The good classification result for* QTC_C *with unsmoothed values might be due to artefacts from before the start or after the end of the interaction and must be very minute movements since they disappear when using even the lowest smoothing values.*

Figure 3.17: *Classification results for* left vs. right *and* early vs. late*. Dot: mean value, errorbar:* 95% *confidence interval, solid red line:* H_0*, left dashed red line: intimate space (Hall 1969), right dashed red line: personal space as defined by Hall (1969).*



(a) *Results for* indicator *vs.* no indicator *in the late condition.* Left: *overall best results, smoothing values:* $1cm$ *and* $0.0s$, right: *typical result, smoothing values:* $1cm$ *and* $0.02s$. *The overall results are very dependent on the smoothing parameters. However, a significant jump in classification rates can be observed for* $d_s = 0.9$ *regardless of the actual smoothing values which can be explained by the model highlighting the distance at which the actual avoidance by the "robot" happened if there was any.*



(b) *Results for* indicator *vs.* no indicator *in the early condition.* Left: *best result, smoothing values,* $0.1cm$ *and* $0.00s$, right: *typical result, smoothing values* $0.01cm$ *and* $0.0s$. *Results are very dependent on the smoothing parameters. Unsmoothed values contain too many artefacts to be useful for classification.*

Figure 3.18: *Classification results for* indicator vs. no indicator. *Dot: mean value, errorbar:* 95% *confidence interval, solid red line:* H_0, *left dashed red line: intimate space (Hall 1969), right dashed red line: personal space as defined by Hall (1969). Blue, horizontal area:* 95% *confidence interval of* QTC_C *showing that* QTC_{BC} *yields similar results for most values of* d_s *and significantly better results for certain distance values.*

the best result (left) and a typical result (right) for different smoothing and quantisation values in the *late* condition. The distance $d_s = 0.9m$ represents a special case where the classification rates jump to values significantly different from H_0 for all smoothing and quantisation value combinations. This can be explained by the minimum distance of $0.92m$ to $0.98m$ the participants kept to the robot at all times. Using a distance threshold of $d_s = 0.9m$ therefore highlights this part of the interaction by suppressing "unnecessary" information. The *early* condition is shown in Figure 3.18b and depicts the best result (left) and a typical result (right) in the evaluation. Similar to the *late* condition, at $d_s = 0.9$ the classification results typically jump to values close to QTC_C. In some cases QTC_{BC} even significantly outperforms QTC_C for certain d_s, see Fig. 3.18b left. The *middle* condition just provides noise and is therefore unclassifiable via QTC_B, QTC_C, or QTC_{BC}.

3.7 Discussion

This section focuses on the interpretation of the classification results presented in Section 3.6. As described above, employing the probabilistic models as classifiers is used to generate a comparative measure to make assumptions about the quality of the generated representation where significant differences between the two used classes means that the model was able to reliably represent this type of interaction. The general quality of using QTC for the representation of HRSI and the different distances or ranges of distances for the proposed QTC_{BC} based model are evaluated to find suitable regions for the switch between the two variants. Figure 3.15 shows that QTC in general is well suited to classify certain HRSI encounters. Thus, the following discusses the results for the different comparisons and lists the limitations of the presented approach.

Limitations A possible limitation is that the presented computational model was not evaluated in a dedicated user study but on two data sets from previous experiments. However, a model of HRSI should be able to represent any encounter between a robot and a human in a confined shared space. The two used experiments might not have been explicitly designed to show the performance of the presented approach but provide the type of interactions usually encountered in corridor type situations which represents a

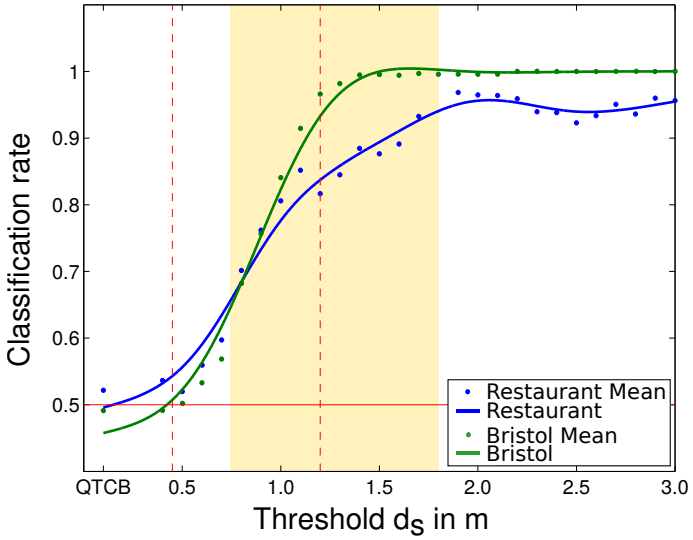


Figure 3.19: *Comparing "Restaurant" and "Bristol" experiment results: Passing on the left vs. passing on the right. The blue curve represents the "Restaurant" experiment classification rates using the lowest smoothing values and the green curve represents the unsmoothed classification rate for the "Bristol" experiment, respectively. The curve has been obtained using a smoothing spline (De Boor 1978) with a p-value of $p = 0.99$. Red line: H_0, left dashed red line: intimate space, right dashed red line: personal space as defined by Hall (1969), yellow area: interval for QTC_{BC} transitions from first classification result significantly different from H_0 to first result not significantly different from pure QTC_C. The better results for the "Bristol" experiment can be explained by the larger amount of training data.*

major part of human-aware navigation. Moreover, a similar model is used in a live system in Chapter 6 which shows its application to real robot data.

The instructions given in the "Bristol Experiment", to cross the corridor with as little veering as possible, might have also influenced the participants behaviour when it comes to keeping a safe distance and will therefore also have had an influence on their experienced comfort during the interaction which may have led to unnatural behaviour on the participant's side. However, as Figure 3.19 shows, the left vs. right conditions yielded similar results in both experiments which indicates that these instructions did not have a significant influence on the participants spatial movement behaviour based on the resulting QTC model.

The presented probabilistic QTC_{BC} uses the distance $d(k, l)$ at time t_{n-1} and t_n to determine if the representation should transition from QTC_B to QTC_C or vice-versa. This might lead to unwanted behaviour if the distance $d(k, l)$ oscillates around d_s. In practice, oscillation around d_s will influence the generated robot behaviour and might lead to the robot inching forwards or "twitching" to one side which can also be interpreted as

prompting movements as described by Peters et al. (2011) to express the robots intention to move.

A general limitation of QTC is that actual sensor data does not coincide with the constraints of a continuous observation model represented by the CND. In the "restaurant" data for example up to 521 illegal transitions were observed which indicates that raw sensor data is not suitable to create QTC state sequences without post-processing. This, however, was solved by using the proposed HMM based modelling adhering to the constraints defined in the CND, only producing valid state transitions. The particle filter based approach presented in Chapter 6, which works on Markov Chains, will need validated QTC state chains which encompasses the inclusion of transition states that have never been observed but are necessary to create legal transitions between the observed states. This is automated in the pipeline presented in Chapter 4.

A major limitation is that important HRSI concepts such as speed, acceleration, and distance, are hard to represent using QTC. While the not simplified version of QTC_B is able to represent relative speeds, it is neither possible to represent the velocity nor acceleration of the robot or the human. Therefore, QTC alone is not very well suited to make statements about *comfort*, *naturalness*, and *sociability*, as defined by Kruse et al. (2013), of a given HRSI encounter. Section 3.2, however, showed that, using implicit distance modelling is able to enrich QTC with such concepts, but many more are missing. In fact Chapter 5 shows a way of how to outsource the generation of velocity commands to pre-existing ROS components and use QTC to constraint the trajectory generation to overcome the issue of having to generate speeds and accelerations.

Another limitation of QTC is the impossibility to infer which agent executes the actual avoid action in the pass-by scenario. When interpreting the graph in Figure 3.9, it is not clear if the human, the robot, or both are avoiding each other. Only the fact that the human started the action is obvious, but it is not possible to infer if the robot participated or not. This could eventually be countered by using the full, not the simplified version, of QTC_C including the relative angles. Even then, it might not be possible to make reliable statements about that and it would also complicate the graph and deprive it of some of its generalisation abilities.

Head-on vs. Overtake The presented classification of pass-by vs. overtaking (see Table 3.1a) shows that QTC_B, QTC_C, and QTC_{BC}, regardless of the chosen d_s, are able to reliably classify these two classes. In the presented data, there are cases where pure QTC_B outperforms pure QTC_C which is not surprising because the main difference of overtaking and pass-by lies in the $(q_1\ q_2)$ 2-tuple of QTC_B, i.e. both agents move in the same direction, e.g. $(-+)$, vs. both agents are approaching each other $(--)$. The 2D information $(q_3\ q_4)$ of QTC_C can therefore be disregarded in most of the cases and only introduces additional noise. This indicates that QTC_B would be sufficient to classify pass-by and overtaking scenarios, but would of course not contain enough information to be used as a generative model or to analyse the interaction. QTC_{BC} allows to incorporate the information about which side robot and human should use to pass each other and the distance at which to start avoiding. Additionally, QTC_{BC} also allows to disregard information for interactants far apart, only employing the finer grained QTC_C where necessary, i.e. when close to each other. Since all of the found classification results were significantly different from $p = 0.5$ – the null hypothesis (H_0) for a two class problem – this distance can be freely chosen to represent a meaningful value like Hall's (1969) personal space $1.22m$. By doing so, the created model also becomes more concise and therefore tractable as mentioned in the requirements for HRSI modelling in the beginning of this chapter.

Left vs. Right The comparison of left vs. right pass-by actions in both experiments shows that using pure QTC_B does, unsurprisingly, yield bad results because the most important information – on which side the robot and the human pass by each other – is completely omitted in this 1-dimensional representation. Hence, all the classification results show that an increase in information about the 2-tuple $(q_3\ q_4)$ representing the 2D movement increases the performance of the classification. On the other hand, the results of both experiments show that the largest increase in performance of the classifier happens at distances of $d_s \geq 0.7m$ and that classification reaches QTC_C quality at $d_s \geq 1.5m$ (see yellow area in Figures 3.16a, 3.17a, 3.19), which loosely resembles the area created by the far phase of Hall's (1969) personal space and the close phase of the social space. These results could stem from the fact that the personal space was neither violate by the robot – be it fake or real – nor the participant. Judging from the data, the results indicate that

information about the side (q_3 q_4) is most important if both agents enter, or are about to enter, each others personal spaces as can be seen from the yellow areas in Figures 3.16a, 3.17a, and 3.19. The information before crossing this threshold can be disregarded and is not important for the reliable classification of these two behaviours. As mentioned in the requirements, recognising the intention of the other interactant is a very important factor in the analysis of HRSI. Reducing the information about the side constraint and only regarding it when close together, allows to focus on the part of the interaction where both agents influence each others' paths and therefore facilitates intention recognition, based on spatial movement.

Figure 3.19 shows that the model gives consistent results over the two experiments in the left vs. right condition which is the only one that could be compared in both. The blue curve shows the classification results for the "restaurant experiment" whereas the green curve shows the results for the "Bristol experiment". Both curves show the same trends of significantly increasing classification results from $0.7m \leq d_s \leq 1.5m$ reaching their pinnacle at $1.5m \leq d_s \leq 2.0m$. This implies that the model is valid for this type of interaction regardless of the actual environment set-up and that the fact that an autonomous robot was used in one of the experiments and a "fake robot" in the other does not influence the data. More importantly, it also shows a suitable distance range for this kind of HRSI that also encloses all the other found distance ranges from the other conditions and is therefore a suitable candidate for QTC_{BC} transitions ensuring that d_s is chosen larger than the lower bound.

Adaptive vs. Non-Adaptive Velocity Control Using a probabilistic model of pure QTC_C, it is not possible to reliably distinguish between the two behaviours the robot showed during the "restaurant experiment", i.e. adaptive vs. non-adaptive velocity control as can be seen from the QTC_C value in Figure 3.16b. However, the QTC_{BC} results indicate that using a very low distance threshold d_s enables QTC_{BC} to distinguish between these two cases for some of the smoothing levels. Figure 3.16b shows that some of the QTC_{BC} results are significantly different from QTC_C. Similar to pass-by vs. overtake, the main difference between the adaptive and non-adaptive behaviour seems to lie in the (q_1 q_2) 2-tuple, i.e. both approach each other ($--$) vs. human approaches and robot stops (-0). On the other hand, the classification rate drops to $p \approx 0.5$ (H_0) at $d_s = 1.3m$

most likely due to the increase in noise. Nevertheless, apart from these typical results, there is also an interesting example where this does not hold true and we see a slight increase in classification rate at $d_s = 1.5m$ which was the stopping distance of the robot (see Figure 3.16b, right). This shows that, even with QTC_{BC}, the results for adaptive vs. non-adaptive seem to be very dependent on the smoothing parameters (see Table 3.1c) and therefore this problem cannot be considered solved. Incorporating another HRSI concept, i.e. velocity or acceleration, might be able to support modelling of these kind of behaviours, but this would decrease the generalisability of the model.

Early vs. Late In the "Bristol experiment", the *early* vs. *late* (see Figure 3.17b) avoidance manoeuvres were evaluated. Just to recapitulate, *early* means the "robot" executed the avoidance manoeuvre $700ms$ before the indicator and in the *late* condition $700ms$ after. The data shows that the model is able to represent this kind of interaction for distances of $0.8m \leq d_s \leq 1.3m$. This is the distance the participants kept to the robot/experimenter in both experiments and loosely resembles Hall's (1969) personal space. In this regard, these results are consistent with the other described interactions showing that participants tried to protect their personal/intimate space. Except for the unsmoothed evaluation, the only reliable classification using QTC_{BC} was achieved inside the mentioned range of $0.8m \leq d_s \leq 1.3m$. QTC_B or QTC_C alone did not highlight the meaningful parts of the interaction and did not yield reliable results. Regarding the unsmoothed case, the fact that all the smoothing levels resulted in a significantly worse QTC_C classification than in the unsmoothed case shows that the unsmoothed result is most likely caused by artefacts due to minute movements before the start or after the end of the experiment. These movements cannot be regarded as important for the actual interaction and must therefore be considered unwanted noise.

Indicator vs. No Indicator The "Bristol experiment" also used indicators (be it flashing lights or cartoon eyes) to highlight the side the "robot" would move to. In the control condition no indicators were used. Modelling these two conditions shows for the *late* condition that for $d_s \geq 0.9m$, which resembles the mean minimum distance kept by the participant, the two cases can be reliably distinguished. The classification rate does not improve significantly for greater distances or pure QTC_C but, to reliably classify these

two conditions is always possible regardless of the distance threshold chosen. Compared to QTC_{BC} at $d_s \geq 0.9m$, pure QTC_C shows worse results for some of the smoothing levels. This indicates that the most important part of the interaction happens at close distances (the mean minimum distance of both agents $d_s \approx 0.9m$) and adding more information does not increase the accuracy of the representation or even decreases it.

3.8 Summary

This chapter presented a HMM based probabilistic sequential representation of HRSI utilising QTC, investigated the possibility of incorporating distances like the concept of proxemics by Hall (1969) into the model, and showed how to learn transitions for the combined QTC model and ranges of distances to trigger them, from real-world data. The data form the two experiments provides strong evidence regarding the ***generalisability [Obj. 1.4]*** and appropriateness of the representation, demonstrated by using it to classify different encounters observed in motion-capture data. Thereby, a ***tractable and concise representation [Obj. 1.5]*** was created that is general enough to ***abstract from metric space [Obj. 1.1]*** but rich enough to ***unambiguously model the observed spatial interactions [Obj. 1.2]*** between human and robot.

The QTC itself models the movement of two Moving Point Objects (MPO) in 1D or 2D space depending on the variant of QTC that is chosen, i.e. QTC_B or QTC_C, respectively. This representation offers all the benefits of a ***well defined qualitative calculus [Obj. 1.5]*** but also comes with certain detriments. It is not able to represent absolute time, distance, or speed. The missing time component will be solved in later chapters and the speed issue is addressed by the used motion planner. The distance, however, has to be incorporated into the model itself. Instead of incorporating a discrete distance value by including the quantitative distance between the agents into the QTC state like in Lichtenthäler et al.'s (2013) work, the presented QTC_{BC} combines QTC_B and QTC_C and switched between them based on a distance threshold d_s. Thereby, the model implicitly includes this distance threshold and the transition between the two variants can later on be used to trigger avoidance behaviours. Hence, QTC_{BC} is able to model pure QTC_B or QTC_C and is also able model the combination of both. These QTC_{BC} states are

then used in a probabilistic representation based on HMMs and are trained from real-world data which, in the case of this evaluation, stems from two different experiments.

Using the two different experiments, regardless of the modelled interaction type, the probabilistic sequential model using QTC is able to ***reliably classify [Obj. 1.2]*** most of the encounters. However, there are certain distances after which the "richer" 2D QTC$_C$ encoding about the side constraint does not enhance the classification and thereby becomes irrelevant for the representation of the encounter. Hence, QTC$_B$'s 1D distance constraint is sufficient to model these interactions when the agents are far apart. On the other hand, the results show that there are distances at which information about the side constraint becomes crucial for the description of the interaction like in passing on the left vs. passing on the right. Thus, there are intervals of distances between robot and human in which a switch to the 2-dimensional QTC model is necessary to represent HRSI encounters. These found distance intervals resemble the area of the far phase of Hall's (1969) personal space and the close phase of the social space, i.e. $0.76m$ to $2.1m$ (see Figure 3.19). Therefore, the data shows that using the full 2D representation of QTC$_C$ is unnecessary when the agents are further apart than the close phase of the social space ($\approx 2.1m$) and can therefore be omitted. This not only creates a more compact representation but also highlights the interaction in close vicinity of the robot, modelling the essence of the interaction, and allows to freely choose a distance threshold for behaviour generation as long as $d_s \geq 2.1m$ holds true. The results indicate that this QTC$_{BC}$ model is a valid representation of HRSI encounters and reliably describes the real-world interactions in the presented experiments.

As a welcome side effect of modelling distance using QTC$_{BC}$, the results show that the quality of the created probabilistic model is, in some cases, even increased compared to pure QTC$_B$ or QTC$_C$. Thereby, besides allowing the representation of distance and the reduction of noise, it also enhances the representational capabilities of the model for certain distance values and outperforms pure QTC$_C$. This shows the effect of reducing noise by filtering "unnecessary" information and focusing on the essence of the interaction.

Coming back to the requirements to a model of HRSI stated in the objectives in Section 1.1 which were to

> ***Abstract from the metric representation [Obj. 1.1]*** to create a transferrable and environment agnostic model, ***represent the qualitative char-***

> ***acter of motions [Obj. 1.2]*** to recognise intention, ***represent the main concepts of HRSI [Obj. 1.3]*** like proxemics (Hall 1969), ***be able to generalise [Obj. 1.4]*** to facilitate knowledge transfer, and devise a ***tractable, concise, and theoretically well-found model [Obj. 1.5]*** that ***facilitates decision processes [Obj. 1.6]***

where the sequential model utilising QTC_{BC} has shown to be able to achieve most of these. Additionally, the representation relies on the ***well founded original variants [Obj. 1.5]*** of the calculus and allows to ***implicitly represent one of the main concepts of HRSI [Obj. 1.3]***, distances. It does so by combining the different variants of the calculus, i.e. the mentioned QTC_B and QTC_C, into one integrated model. The resulting representation is able to highlight the interaction when the agents are in close vicinity to one another, allowing to focus on the qualitative character of the movement and therefore facilitates intention recognition. By eliminating information about the side the agents are moving to when far apart, it also creates a more ***concise and tractable representation [Obj. 1.5]***. Moreover, the model also inherits all the ***generalisability [Obj. 1.4]*** a qualitative representation offers.

The objective of creating a ***qualitative representation that is able to facilitate decision processes [Obj. 1.6]*** by modelling a clear connection between human and robot state has also been fulfilled due to QTC's nature of representing both in the same state. However, for classification the human and robot state were used equally in this evaluation. For a generative framework where the robot state is deducted from the human state using an action generation policy, the robot state cannot be used to classify the current interaction type as it is not known a-priori. This problem is highlighted in Section 6.2 which introduces one possible solution in the context of particle filters for HRSI activity recognition.

Concluding from the above statements, the probabilistic model of QTC_{BC} is able to qualitatively model the observed interactions between two agents, ***abstracting from the metric 2D-space [Obj. 1.1]*** most other representations use, and ***implicitly incorporates the modelling of distance thresholds [Obj. 1.3]*** which, from the observations made in the experiments, represent one of the main social measures used in modern HRSI, proxemics (Hall 1969).

In the following chapters, QTC is used as the basis for all HRSI models. The HMM representation is used in Chapter 4 exactly as it is described here and in a slightly different form in Chapter 6.

—All we have to believe with is our senses, the tools we use to perceive the world: our sight, our touch, our memory. If they lie to us, then nothing can be trusted. And even if we do not believe, then still we cannot travel in any other way than the road our senses show us; and we must walk that road to the end.

Neil Gaiman, Science Fiction and Fantasy Author

4

People Perception and QTC State Generation

The QTC based model described in Chapter 3 works over intervals of time and thus requires the trace of positions of the robot and the trace of positions of the human(s) in close vicinity to it. The robot's position can easily be extracted from its self-localisation via Adaptive Monte-Carlo Localisation (AMCL) (Thrun et al. 2005) and then merged into a single track based on its timestamp. Regarding the trace of human positions, in the experiments described in Section 3.5 only one human at a time was detected using a sophisticated motion capture system and the detections were merged in to tracks during post-processing. This is very precise but it is infeasible to have motion capture markers attached to people living or working in a real-world environment, or very expensive cameras in all of its corridors and rooms. Additionally, for a live system these detections would have to be merged into a single track being able to deal with multiple humans at the same time using data association. Thus, the basic challenge is the ***detection and tracking of humans in the vicinity of the robot using its on-board sensors [Obj. 3.2]*** and considering problems such as the robots ego-motion, varying ambient conditions, and

occlusion as mentioned by Fong et al. (2003). Due to the importance of these kind of applications in HRI in general and HRSI in particular, there are several solutions to the problem of detection and tracking of humans or body parts, as shown for example in surveys by Gavrila (1999) or more recently by Walia & Kapoor (2014).

To detect people using *Linda* the robot, two specific body part detectors for human upper bodies, via the head mounted RGB-D camera, and for legs, via the laser sensor are used and described in this chapter. Mere detection of humans, however, as mentioned above, is not enough to generate QTC states because each state is generated over a time interval $T = [t_{n-1}, t_n]$ which requires that two detections have to be attributed to the same person. The easiest way of achieving this is to use a tracking algorithm such as a particle or Kalman filter and a data association strategy appropriate for the given environment. Hence, this chapter also describes the multisensor Bayesian tracking framework by Bellotto & Hu (2010*b*) that allows merging subsequent detections of either of the detectors into a single track increasing the reliability of the resulting estimate of the human position by using multiple sensors and creating trajectories that can be used for QTC state chain generation. This general purpose human detection and tracking system has already been used in two other European projects (e.g. Linder et al. 2016) apart from STRANDS (e.g. Hawes et al. 2016, Beyer et al. 2016, Duckworth et al. 2016) and can of course also feed directly into reactive human-aware navigation approaches (e.g Lu et al. 2014). In the presented case, however, it will be used for the ***incremental real-time generation of QTC state chains [Obj. 3.1]***.

The QTC state chains are generated in real-time between the robot and all tracked humans in its vicinity in all QTC variants introduced in Chapter 3. Additionally, in preparation for the online belief generation, i.e. state prediction and action selection, in Chapter 6, QTC_C states are also created between each human and the robot's short term goal. The motivation for this will be made clear in Section 6.2, which describes the basis for the autonomous interaction classification and action selection. Moreover, this chapter also introduces a method of creating valid QTC state chains according to Definitions 3.1 and 3.2 from discrete input data.

Detecting walking or standing pedestrians (e.g. Fod et al. 2002, Schulz et al. 2003) is the most important enabling technology for HRSI and, therefore, a widely studied field due to the advances in autonomous cars and robots. To this end, many successful full-body or

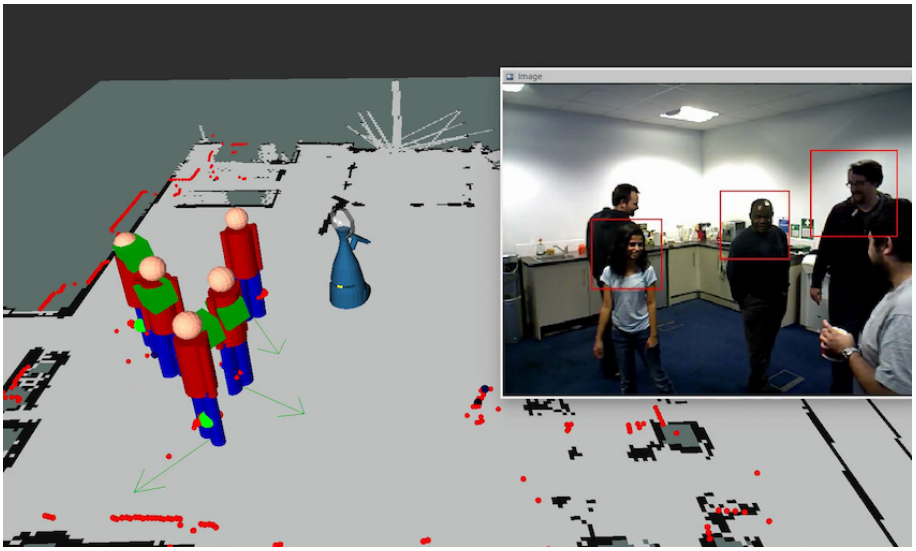


Figure 4.1: *Example output of the ROS based people perception pipeline, showing the tracker and detector results in* `rviz`*. Red dots: laser scans, green spheres: leg detector detections, green cubes and red rectangles in camera image: upper body detections, red/blue stick figures: the resulting tracks based on the Kalman filtering of the detections. The green arrows represent the direction of the motion vector of the detected person. The blue figure represents* Linda*'s position produced by AMCL on the grey metric map.*

partly occluded body detectors have been developed (e.g. Dalal & Triggs 2005, Sudowe & Leibe 2011, Wojek et al. 2011, Spinello & Arras 2011). Most of these detectors, however, suffer from high computational costs which is why currently used approaches, such as the *upper body detector* by Mitzel & Leibe (2012), rely on the extraction of a Region of Interest (ROI) to speed up detection. Moreover, given spatial constraints in narrow corridors, the robot might not be able to observe a human from a far enough distance to see the entirety of their body which is why full body detectors are more suited for outdoor applications where space is not an issue. Therefore, the presented framework employs Mitzel & Leibe's (2012) upper body detector for the real-time detection of walking or standing humans.

The second detector used is a leg detector based on Arras et al.'s (2007) work and has become a standard ROS component for people perception. Like detectors, human tracking is an important part of a perception system for human spatial movement and invaluable for QTC state chain generation. Hence, a variety of tracking systems have been introduced by the robotics community (e.g. Feyrer & Zell 2000, Arras et al. 2008, Jaffari et al. 2014, Linder et al. 2015). In order to use the wealth of provided sensors on a modern mobile robot, the principle of sensor fusion (e.g. Feyrer & Zell 2000) is used in the presented probabilistic real-time tracking framework which fuses the two mentioned sensors

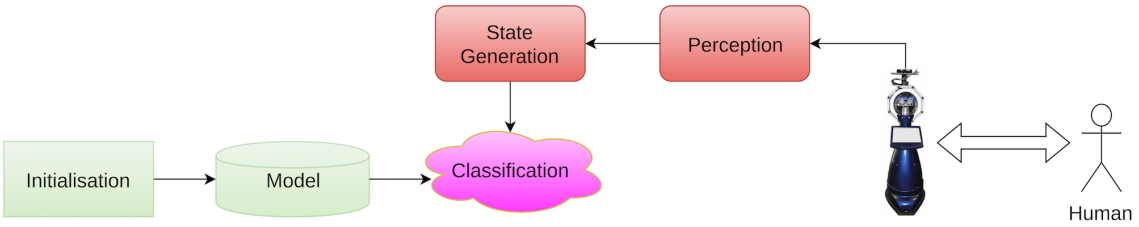


Figure 4.2: *This chapter's contribution to the system extending the framework shown in Figure 3.1 to create the overall system from Figure 1.1. The novel contribution is highlighted in red and the offline classification only used in this chapter in pink. The interaction of human and robot is observed using the robot's on-board sensors and automatically transformed into valid QTC state chains. To show the generalisability of the model it is trained with data collected in a previous study (see Section 3.5.2) and used to offline classify the interaction type observed by the robot. This people tracking and state generation approach is the basis for the classification, state prediction, and behaviour generation in later chapters and paramount for the development of an autonomous system.*

and employs an Extended or Unscented Kalman filter to track and predict the movements of humans as initially presented by Bellotto & Hu (2010*a*) and (2010*b*). However, the tracker itself does not rely on a specific detector for input and is very modular in design. Figure 4.1 shows the example output of the tracking framework.

Chapter Contributions The main contribution of the work presented in this chapter is the combination of existing detection and tracking approaches for human perception in the vicinity of the robot by Mitzel & Leibe (2012), Arras et al. (2007), and Bellotto & Hu (2010*b*) into a holistic ROS framework and the automated incremental generation of QTC state chains at runtime. The presented framework is tailored to the capabilities of *Linda* the robot using all ***available sensors producing people tracks in real time [Obj. 3.1]*** to facilitate a ***fully autonomous system [Obj. 3]***. This detection and tracking framework is used to automatically generate QTC state chains for every tracked person in order to classify the type of HRSI encounter the robot ought to engage in. These generated state chains are the basis for the classification, belief and behaviour generation approaches described in the following chapters. As a proof of concept a small experiment using *Linda* in a real-world office environment, only ***relying on the robots on-board sensors [Obj. 3.2]***, shows how these generated QTC state chains can be classified using the HMM models from the "Bristol experiment" (see Section 3.5.2). Thus, the evaluation not only ***shows that it is possible to use the robot's on-board sensors to generate a meaningful QTC representation [Obj. 3.2]*** but also that this particular

QSR ***allows to transfer knowledge gathered from different sensors and in a different environment [Obj. 1.4]***, therefore, proving the QTC model to be ***environment, agent, sensor, and perspective agnostic [Obj. 1.1]*** considering that for the creation of the model the robot and human were observed by external cameras whereas in this experiment the robot uses its self-localisation and observes the human using its own sensors. The contribution of this chapter to the overall proposed system can be seen in Figure 4.2 which shows how the automatically generated state chains and the previously created interaction model are used in an offline classification process to determine the type of interaction the robot engaged in.

People perception is necessary to create an autonomous human-aware navigation system which makes it a requirement for all the work described in this thesis but it was not the main focus of it. Nevertheless, as an enabling technology that is not readily available due to a lack of open source ROS implementations, this problem had to be solved which is why it found its way into this thesis. To not go beyond the scope of this work, the evaluation in this chapter only investigates if the created perception pipeline is suitable to generate meaningful QTC state chains. For an evaluation of all the subsystems please refer to their original publications by Mitzel & Leibe (2012) for the upper body detector, Arras et al. (2007) for the leg detector, and Bellotto & Hu (2010*b*) for the Bayesian tracking framework. Moreover, the performance of the presented perception pipeline in crowded and dynamic environments has recently been evaluated in the context of a different project by Linder et al. (2016).

The work presented in this chapter has been published at the *Workshop on Machine Learning for Social Robotics* at the *IEEE International Conference on Robotics and Automation (ICRA), 2015*. Please see Appendix A.2 for the author's contributions.

4.1 System Overview

This section presents the integrated system shown in Figure 4.3, consisting of the perception pipeline including the leg detector, upper body detector, the tracker, the qualitative spatial representation module, and the library to create the QTC state chains online.

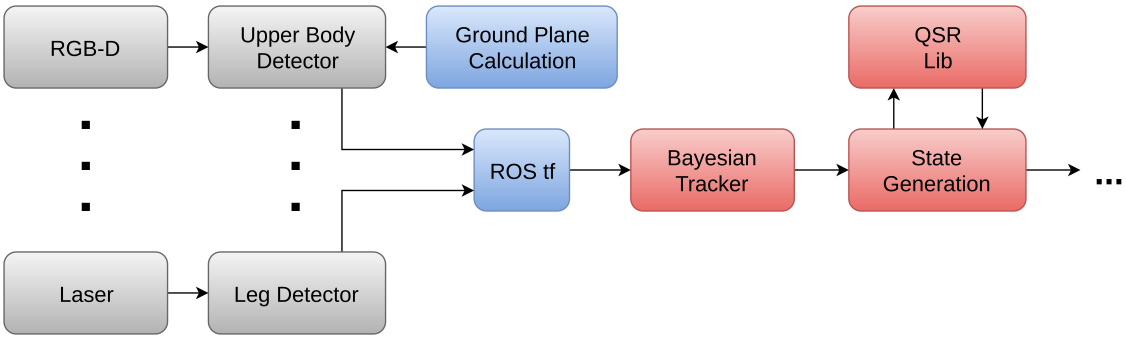


Figure 4.3: *Conceptual overview of the system architecture. The number of detectors is variable due to the modular design of the tracker and its ability to merge several detections from different sensors. The two detectors shown here are the ones used on* Linda *in all the following experiments. From left to right, the detectors produce the raw and unfiltered detections for possible legs and upper bodies using the helper functions to speed up the process via ground plane estimation and using the ROS transformation tree to unify the results in the same coordinate frame, the tracker uses data association and a Kalman filter to combine and track the persons using the detections as input, the resulting tracks are used to create QTC state chains using the so-called* QSR Lib *which QTC is part of. These state chains can than be used for any kind of learning or classification approach.*

4.1.1 Detectors

Robots use a range of sensors to perceive the outside world, enabling them to reason about its future state and plan their actions. *Linda*, the robot used here, has two main sensors that can be used for people detection, i.e. the head mounted Asus Xtion RGB-D camera and the Sick s300 laser (see Section 1.3.2). Hence, the following presents two detectors based on the RGB-D and laser scanner, respectively. Example output can been seen in Figure 4.1.

Mitzel & Leibe's (2012) so-called *upper body detector* uses the template shown in Figure 4.4 and the depth information of a RGB-D sensor to identify upper bodies (shoulders and head) as depicted in Figure 4.5, designed to work for close range human detection using head mounted cameras. This detector was originally based on stereo outdoor data; subsequently an integrated tracking system using a Kinect like RGB-D sensor and the mentioned detector was introduced by Jaffari et al. (2014). To reduce the computational load, this upper body detector employs a ground plane estimation or calculation to determine a ROI most suitable for detection of upper bodies of a standing or walking person. The actual depth image is then scaled to various sizes and the template is slid over the ROI of the image generating a distance matrix which defines the distance of a given pixel to the template. After using non-maximum suppression, bounding boxes are fitted around

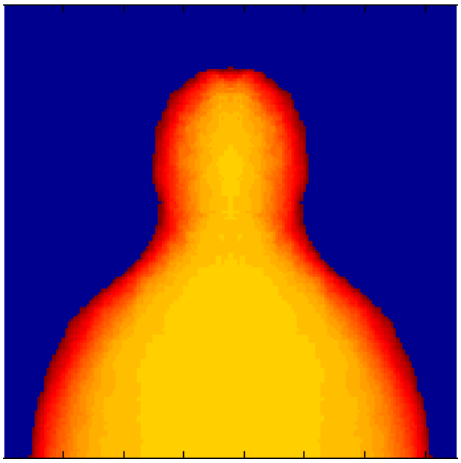
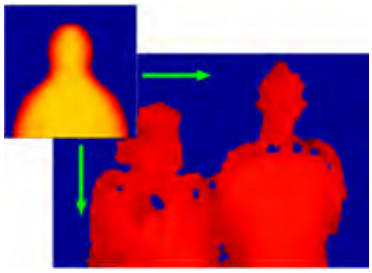
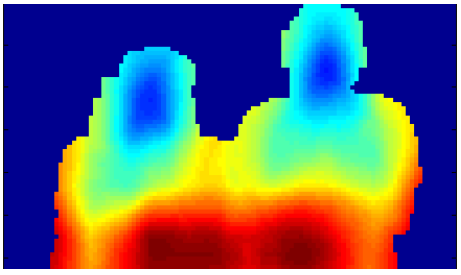


Figure 4.4: *The upper body depth template learned from 600 upper bodies and used for matching. Image taken from (Mitzel & Leibe 2012)*

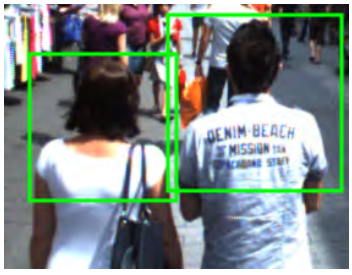
(a) *The original RGB image.*



(b) *The template is slid over the depth images in various sizes.*



(c) *The resulting distance matrix showing the similarity between depth image and template.*



(d) *The resulting bounding boxes after non-maximum suppression.*

Figure 4.5: *The upper body detection process. Images taken from (Jaffari et al. 2014)*

the found matches (see Figure 4.5 for a visualisation of that process). This detector works in real time, meaning $\approx$ 25 fps which corresponds to the frame rate of the Asus Xtion. The main advantage of this detector, compared to full body detectors, is that the camera on *Linda* is mounted at a height of $1.72m$ which only allows it to see upper bodies in normal corridor or room settings, such as offices or flats, due to the restrictions in space and the field of view of the Asus camera. This is in fact true for a large variate of currently used research service robots because mounting a camera at head height facilitates face recognition and therefore also HRI. Additionally, due to the restrictions in space in the normal working environment of the robot, it can hardly ever see the floor using the head-mounted camera which is why the ground plane calculation developed for this thesis is based on the geometry of the robot and the position of the Pan-Tilt Unit (PTU) the camera is mounted on instead of the depth image based estimation as described by Jaffari et al. (2014).

Due to the very limited field of view of the RGB-D camera, a laser based leg detector by Arras et al. (2007) is used in addition to that. Using laser scanners for people perception is popular in mobile robotics (e.g. Fod et al. 2002, Schulz et al. 2003) because most currently used platforms provide such a sensor, which also has a wider field of view than a camera and is less dependent on ambient lighting conditions. Arras et al. (2007) define a set of 14 features for the detection of legs including: the number of beams, the circularity, the radius, mean curvature, and the mean speed, to only name a few. These features are used for the supervised learning of a set of weak classifiers using recorded training data. Schapire & Singer's (1999) AdaBoost algorithm is employed to turn these weak classifiers into a strong classifier, detecting legs in laser range data. The approach was evaluate by Arras et al. (2007) in various office and corridor settings which proved it ideal for most indoor robotics environments. At the time of writing this thesis, this approach is considered state-of-the-art and the implementation of the detector[23] is part of the official ROS people stack[24]. This people stack implementation, however, does not allow for sensor fusion which is why, together with the upper body detector, the leg detector is used separately to feed into the tracker described in this chapter.

[23] `http://wiki.ros.org/leg_detector`
[24] `http://wiki.ros.org/people`

The output of both detectors can be seen in Figure 4.6a which shows the visualisations of their outputs in the ROS visualisation tool *rviz*[25]. The detections do not identify the person but just return a list of x, y, z positions, one for each person and detector, where z is projected down to the map. The detections have to be fused to generate the trace of positions of the humans in the vicinity of the robot by using the tracker described in the following.

4.1.2 Tracker

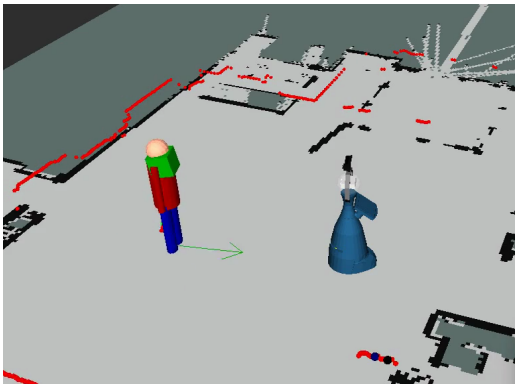
Having only detectors makes it impossible to create QTC states because, as described above, in order to create a QTC state one needs to look at the movement of human and robot over the interval $T = [t_{n-1}, t_n]$ which requires to merge subsequent detections into a single track for each human.[26] Tracking allows to create these QTC states or even chains of states by fusing separate consecutive detections of a single person into one common track. Moreover, to use the wealth of information provided by a robot equipped with multiple sensors, using purely vision based trackers such as one introduced by Jaffari et al. (2014) – from which the upper body detector was extracted – is not feasible because the tracking rate should not depend on any one of the detectors or sensors. This means that the tracker should not only execute a prediction step when one of the detectors fires but produce predictions at a constant frequency and update them when new information from a detector becomes available. Therefore, a solution for *Bayesian tracking*, originally proposed by Bellotto & Hu (2010*b*), is used as implemented by Bellotto et al. (2015) (see Figure 4.6b). This tracker allows native combination of multiple sensors and creates new predictions at a fixed frame rate, executing an update step when one of the detectors provides new information. Bellotto & Hu (2010*b*) showed that their Bayesian tracker, based on an Unscented Kalman Filter, achieves comparable results to a Sampling Importance Resampling (SIR) particle filter in several people tracking scenarios, although it is computationally more efficient in terms of estimation time.

In the current implementation different tracking configurations can be used by defining the fixed frame observation models (one for each detector) and the noise parameters of the constant velocity model to predict human motion, to for example compensate for loss

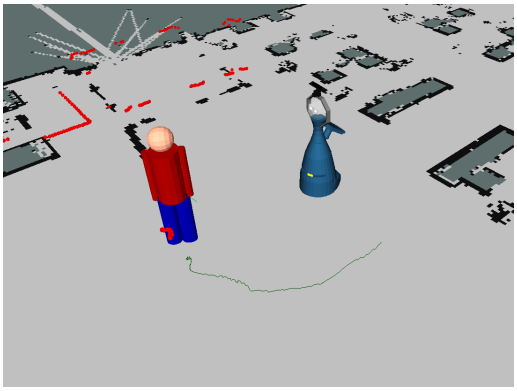
[25] `http://wiki.ros.org/rviz`

[26] Since every mobile robot needs self-localisation to navigate, this problem does not arise for the robot.

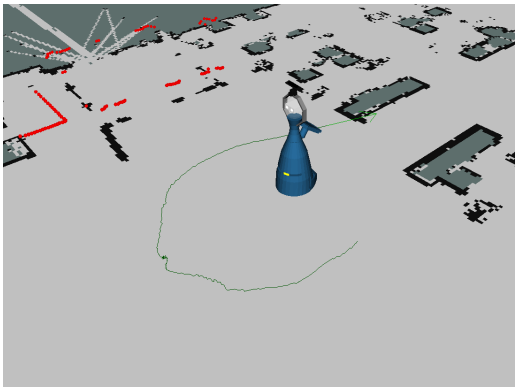
(a) The two used detectors. Green sphere: leg detector, green cube and red box in image: upper body detector.



(b) The tracker output. Overlaid red and blue figure: position of tracked human, green arrow: orientation of velocity vector.



(c) A human moving around the robot. Showing his/her current position and the path since the start of the tracking.



(d) The complete path of the human walking around the robot. Human left the field of view of the laser and is not tracked any more.

Figure 4.6: *The visualisations of the detector and tracker outputs using the ROS visualisation tool* `rviz`*. The red dots represent laser beams hitting an object.*

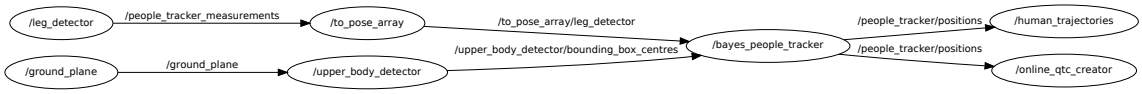


Figure 4.7: *A simplified representation of the ROS nodes and their most important connections, created using the ROS visualisation tool* `rqt_graph`.

of detection, as discussed by Li & Jilkov (2000). A gating procedure is applied using a validation region relative to the target, based on the chosen noise parameters, for each new predicted observation in order to reduce the chance of assigning false positives and incorrect observations (Bar-Shalom & Li 1995). New detections are then associated to the correct target using a Nearest Neighbour (NN) association algorithm, suitable for computationally less powerful robot systems, or a more sophisticated Nearest Neighbour Joint Probabilistic Data Association (NNJPDA), which is more reliable but also less efficient regarding computation time (e.g. Bellotto & Hu 2010*a*, Linder et al. 2015).[27] If no suitable target could be found, the detections are stored and eventually used to create a new track if they are stable over a predefined time frame, i.e. a predefined number of consecutive detections given a pre-defined maximum time delta between them.

The tracking algorithm itself, can be chosen by the user to either use a particle filter, an Extended Kalman Filter, or an Unscented Kalman Filter. All the approaches are able to deal with highly non-linear data and are, therefore, able to reliably track any kind of motion given the correct prediction model. For the relatively simple tracking problem at hand and given the linearity of the constant velocity prediction model and the used Cartesian observation model, a standard Kalman Filter would have been sufficient but for all subsequent experiments using *Linda* the robot, the Unscented Kalman Filter was chosen. This filter produces the same results when compared with a standard Kalman Filter given the used prediction model and comes at no significant increase in computational cost while at the same time being more generic. Since the tracking framework allows defining any kind of motion model, having this default option enables it to deal with any kind of user input while reliably predicting motion, tracking the humans in the vicinity of the robot. For each detected human, it provides a Universally Unique Identifier (UUID) that

[27] NNJPDA was used in all following experiments.

is consistent as long as the person is observed by the robot[28] and the x, y positions of the humans in its vicinity including their assumed constant velocity vector $\overrightarrow{v}$.

The ROS nodes making up the complete systems and their connections with each other can be seen in Figure 4.7. This shows the communication flow from the detectors to the tracker and from the tracker to the QTC generation module.

4.2 Online QTC state chain generation

The output of the tracking framework, i.e. the position, velocity, and the direction of the velocity vector of the tracked humans, can either directly be used for reactive human-aware navigation like the ROS implementation of layered costmaps by Lu et al. (2014) and the simple stop-and-wait behaviour described in the "restaurant experiment" in Section 3.5.1, or for the online generation of QTC state chains. To generate any of the QTC variants introduced in Chapter 3 for each tracked human and the robot, the output of the tracker and the self-localisation of the robot is used. This automatic generation module, however, does not only offer the simple generation of state chains, but also the generation of valid state chains in accordance with Definition 3.1 and 3.2 despite any unobserved transitions therein due to the discrete sensor measurements. The tracker decreases the number of unobserved transitions by creating a continuous signal from discrete observations, but in a physical system, where the tracker runs at $30Hz$, even this will result in unobserved states. The HMM approach introduced previously is able to deal with these missing observations due to it's emission layer, but to allow for online classification of partial state chains, this HMM-based approach is not feasible, which makes the validation of state chains necessary for the model introduced in Chapter 6.

To validate a state chain $\mathbf{Q}$, an additional state needs to be injected between two states that would represent an illegal transition $Q_i \not\rightsquigarrow Q_{i+1}$ to turn it into a legal transition $Q_i \rightsquigarrow Q^t \rightsquigarrow Q_{i+1}$ where Q^t represent an artificial transitional state. Algorithm 4.1 shows this procedure which basically consists of replacing symbols in state Q_{i+1} with 0 that would otherwise represent and illegal transition of this symbol and insert it in between the two original states. In order for this algorithm to work, the qualitative symbols $\{-, 0, +\}$

[28] Once the person is not tracked any more and then re-enters the FoV of one of the detectors, a new UUID is assigned.

Algorithm 4.1 Validate state chain **Q**

Require: **Q**
Ensure: $\exists Q_i \rightsquigarrow Q_{i+1} \; \forall \; (Q_i, Q_{i+1}) \in \mathbf{Q}$
1: **for** $(Q_i, Q_{i+1}) \in \mathbf{Q}$ **do**
2: $Q^t \leftarrow Q_{i+1}$
3: $q_j^t = 0 \; \forall \; q_j \in Q_i, q_j^t \in Q^t : \phi(q_j, q_j^t) = 2$
4: **for** $m \in [1, \ldots, |Q_i|]$ **do**
5: **for** $n \in [m+1, \ldots, |Q^t|]$ **do**
6: $(q_m^i, q_n^i) \in Q_i$
7: $(q_m^t, q_n^t) \in Q^t$
8: **if** $|q_m^i| + |q_n^i| = 1 \wedge |q_m^t| + |q_n^t| = 1$ **then**
9: **if** $\max(\{|q_m^i - q_m^t|, |q_n^i - q_n^t|\}) > 0$ **then**
10: $q_m^t = 0, q_n^t = 0$
11: **end if**
12: **end if**
13: **end for**
14: **end for**
15: **if** $Q^t \neq Q_{i+1}$ **then**
16: $\mathbf{Q} = [Q_1, \ldots, Q_i, Q^t, Q_{i+1}, \ldots, Q_N]$
17: **end if**
18: **end for**

are interpreted as $\{-1, 0, 1\}$. In Algorithm 4.1 line 2 this new transitional state Q^t is created which is a copy of the state Q_{i+1} which will be inserted in between $Q_i \rightarrow Q_{i+1}$ if necessary, i.e. $Q^t \neq Q_{i+1}$. Checking the conceptual distance between the symbols ϕ in line 3, the algorithm ensures that where $\phi(q_j, q_j^t) = 2$, q_j^t is set to 0 removing $- \leftrightarrow +$ transitions. The two loops starting in lines 4 and 5 check ever symbol of the state Q_i against every symbol in Q^t. Since, in the resulting Matrix M where a set of four symbols $q_n^{i,t}$ and $q_m^{i,t}$ with $q^i \in Q_i$ and $q^t \in Q^t$ is compared

$$M = \begin{bmatrix} q_1^{i,t} & q_1^{i,t}, q_2^{i,t} & q_1^{i,t}, q_3^{i,t} & q_1^{i,t}, q_4^{i,t} \\ q_2^{i,t}, q_1^{i,t} & q_2^{i,t} & q_2^{i,t}, q_3^{i,t} & q_2^{i,t}, q_4^{i,t} \\ q_3^{i,t}, q_1^{i,t} & q_3^{i,t}, q_2^{i,t} & q_3^{i,t} & q_3^{i,t}, q_4^{i,t} \\ q_4^{i,t}, q_1^{i,t} & q_4^{i,t}, q_2^{i,t} & q_4^{i,t}, q_3^{i,t} & q_4^{i,t} \end{bmatrix} \tag{4.1}$$

the order of the elements is irrelevant $q_n^{i,t}, q_m^{i,t} \Leftrightarrow q_m^{i,t}, q_n^{i,t}$ and only has to be checked once. This process is sped up by only considering one half of the matrix. Additionally, the diagonal is excluded due to the check for transitions $- \leftrightarrow +$ in line 3.

$$M \approx \begin{bmatrix} & q_1^{i,t}, q_2^{i,t} & q_1^{i,t}, q_3^{i,t} & q_1^{i,t}, q_4^{i,t} \\ & & q_2^{i,t}, q_3^{i,t} & q_2^{i,t}, q_4^{i,t} \\ & & & q_3^{i,t}, q_4^{i,t} \\ & & & \end{bmatrix} \tag{4.2}$$

The first check in line 8 of Algorithm 4.1 creates the sum of the absolute values of $q^i_{n,m}$ and $q^t_{n,m}$ which, if equal to 1, symbolises that one of the symbols changes from $\{-,+\}$ to 0 and vice versa which is the first indication that Definition 3.1 might be violated. Note, $- \leftrightarrow +$ are already mitigated by replacing the symbol q^t with 0. An example case for an illegal transition would be $(0,-) \rightarrow (-,0)$ but this could also represent the legal case of $(0,-) \rightarrow (0,-)$. In order to find the illegal cases, line 9 in Algorithm 4.1 finds the maximum of the absolute values $\max(\{|q^i_m - q^t_m|, |q^i_n - q^t_n|\})$ which in the illegal case $(0,-) \rightarrow (-,0) \Leftrightarrow (0,-1) \rightarrow (-1,0)$ would be $\max(\{|(-1) - 0|, |0 - (-1)|\}) = 1$ or for $(0,+) \rightarrow (+,0) \Leftrightarrow (0,1) \rightarrow (1,0)$ would be $\max(\{|1 - 0|, |0 - 1|\}) = 1$ and for every legal case that remains after line 8, e.g. $(0,-) \rightarrow (0,-) \Leftrightarrow (0,-1) \rightarrow (0,-1)$, would be $\max(\{|0 - 0|, |(-1) - (-1)|\}) = 0$. If such an illegal transition is found both symbols in the artificial transitional state are set to the 0 state $q^t_n = q^t_m = 0$ because every symbol can transition to 0 at every point in time. An easy example is $(--0+) \not\rightsquigarrow (-0--)$ which would be turned into $(--0+) \rightsquigarrow (-000) \rightsquigarrow (-0--)$. Lastly, all states $Q_{i+1} \in \mathbf{Q}$ are removed if $\exists Q_i : Q_i = Q_{i+1} \; \forall \; (Q_i, Q_{i+1}) \in \mathbf{Q}$ to create a state chain $\mathbf{Q}$ that complies with Definition 3.2.

All of these functionalities are optional and can be dynamically reconfigured during runtime to cater to different user needs. If, for example, a QTC state for each timestamp is required as in the work by Duckworth et al. (2016), the validation and collapsing of the state chain can be turned off. Additionally, the module also allows to specify a quantisation factor that defines how far one of the agents has to move from one time step to the other to be considered a non-zero-state and a smoothing time over which the position of human and robot are averaged as described in Section 3.5 of the previous chapter. This has proven necessary to remove noise from the centre points oscillating around the connecting line or one of the perpendicular lines of the double cross.

The final result of this module is a continuous and valid QTC state chain per person in the vicinity of the robot that represents the entire trace of interaction of the human

(a) *Four people are tracked. Two of them only via the upper body detector, one only via the leg detector, and one via the combination of both using NNJPDA.*

(b) *Five people are tracked. The upper body detector only picks up two due to occlusion or "incorrect" body posture. The other three are tracked via the leg detector input.*

Figure 4.8: *As a proof of concept, several people moving around an office environment were tracked, showing that the multisensor tracking compensates for false negatives of the detectors.*

and robot in QTC_{BC} and can be associated with the person using the UUID. This is used in the subsequent evaluation to classify the interaction type based on pre-trained HMMs using the data of the "Bristol experiment" (see Section 3.5.2). In addition to this, the system also creates a QTC_C state chain for said humans encoding the interaction of the human with the robot's goal. This is used for online classification and will be picked up on in Chapter 6.

4.3 Evaluation

As mentioned in the beginning of this chapter, the majority of the presented components have been evaluated in the publications describing them. For the evaluation of the detectors, please refer to Mitzel & Leibe's (2012) and Arras et al.'s (2007) work, respectively. An exhaustive evaluation of the tracker can be found in Bellotto & Hu's (2010*b*) work and more recently by Linder et al. (2016).

Since this chapter presents the integration of all these components into a state-of-the-art robot platform and the widely used Robot Operating System (ROS), it is presented in a short proof of concept experiment to show that the QTC based model presented in Chapter 3 is suitable to work on data generated by the robot's sensors. To show that the tracking framework is able to cope with groups of people, *Linda* was deployed in an open

office environment, observing people in the kitchen area. Screenshots of the live system can be seen in Figures 4.1, 4.6, and 4.8. Please refer to Appendix B for links to videos.

To evaluate the QTC models, the robot was driving along a corridor encountering an oncoming human, engaging in a *pass-by* interaction. The robot was driving on a straight line towards its goal while the human was avoiding the robot to either the left or right side. Following the Oz of Wizard study design (see Section 2.4.1), this was repeated 7 times for each side by the author of this work. During the experiment, the human interaction partner tried to match the speed of the robot to which end they executed a number of test trials to get acquainted with its movement speed prior to the recording of state chains. These resulting state chains were classified into passing on the left vs. passing on the right using the dataset presented in Section 3.5.2, featuring two human interactants. The HMMs for classification were trained using the trajectories of both humans recorded via a motion capture system to not only show that the model is able to deal with sensor data but also to use models created from a different set-up and different sensor to classify it.

4.3.1 Results

Figure 4.9 shows the recorded QTC state chains in the HMM-based representation from Section 3.4. The figure is clearly divided into two possible paths, passing on the left and passing on the right as can be seen from the $-$ or $+$ for q_3 and q_4 in the tuple $(q_1\, q_2\, q_3\, q_4)$ of QTC_C and shows visual similarities to Figure 3.9 which encodes the same kind of interaction in a different context. To give an example of one of the interactions: the two most probable paths for the two conditions both start with the agents approaching each other $(-\,-\,0\;0)$ and then diverge into left or right. The most probable path for passing on the left is: $(-\,-\,-\,-)$ both approach each other and go to the left, $(0\;0\,-\,-)$ both are shoulder to shoulder on their left side, $(+\,+\,-\,-)$ both agents move away after passing still being on the left of the connecting line, $(+\,+\,0\;0)$ both move apart.

As stated above, the recorded encounters are classified using models trained from data collected during the Bristol experiment using motion capture. The classifiers were trained for three different conditions, i.e. starting the circumvention early, late or in between where each is $500ms$ apart. Each of the six models (three per side) was trained with 162 to 178 (for passing on the right) and 183 to 189 (for passing on the left) QTC_C state

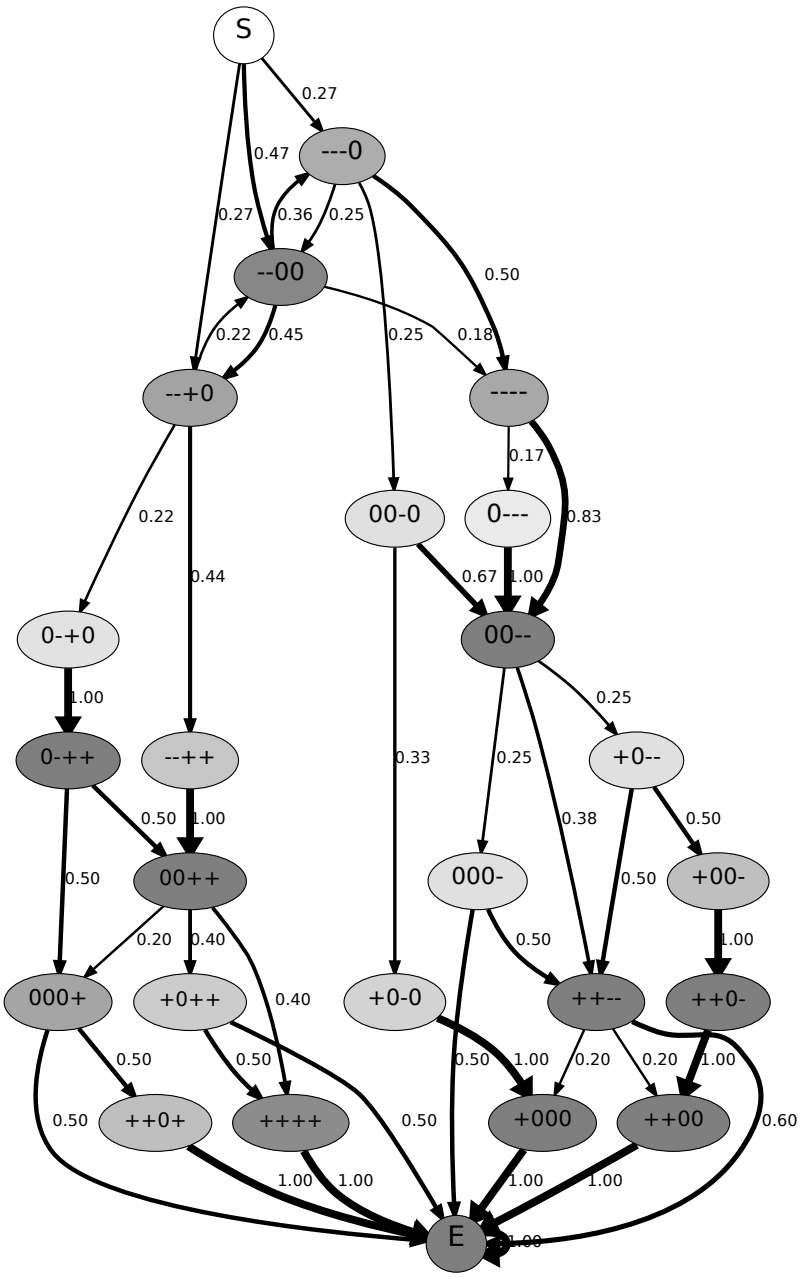


Figure 4.9: *Visualisation of the HMM trained from the recorded QTC state chains. Reminder: a − in the 3rd and/or 4th position of the tuple indicates circumvention on the left and a + in the respective positions represents circumvention on the right. Transitions with a probability below* 0.15 *have been pruned for visualisation purposes. The colour of the nodes represents the a-priori probability of that specific state to be present in any observed chain (from white = 0.0, e.g. "S", to dark grey = 1.0, e.g. "E"), i.e. its observation probability. This figure looks rather similar to Figure 3.9 which was created using recordings of a motion capture system, showing the similarity between these actions visually.*

chains, respectively. Using these models and the state chains generated using sensor data via the presented tracking framework, classification rates from 78.57% to 85.71% were achieved on the dataset using input generated from the integrated systems instead of external motion capture. These results are not as good as the classifications rates of up to 100% from Table 3.2b showing the "Bristol Experiment" results, but are well above the null-hypothesis $H_0 = 50\%$ for a two class example. Hence, the two classes can be reliably classified.

4.4 Discussion

This chapter presents components that have already been proposed for human detection and tracking and the overall tracking system has been evaluated by Linder et al. (2016). However, for the first time, this system has been brought together for the online generation of QTC state chains. The achieved classification results show that even when using a HMM model trained on data obtained by different sensors and in a different environment, not even using a robot, the chosen QTC representation of the interaction, QTC_C in this case, is able to abstract from all of this and to still reliably classify new encounters. Moreover, the robot is not able to observe humans after they passed it and are outside of its FoV, whereas the motion capture system provided full observability. Hence, the state chains generated using the presented system were incomplete with regards to the used HMM model but could still be reliably classified. This, however, might have been the reason for the lower classification rates compared to the results in Chapter 3. Nevertheless, this shows that QTC in general is suitable to represent HRSI using on-board sensors only. Additionally, the introduced system allows to use the robot in HRSI experiments using the incremental online generation of QTC state chains for real-time classification and behaviour generation approaches. This represents an important corner stone of the remainder of this thesis, allowing evaluation of the system on a mobile robotic platform such as *Linda* interacting with the participants autonomously.

Limitations and Lessons Learned The described systems is used throughout the STRANDS project and over the years a few observations about peculiarities and limitations of the system have been made. Like all robotic systems, this approach is very

susceptible to sensor noise and limited fields of view when it comes to detectors. The upper body detector itself, for example, has shortcomings when it comes to detecting sitting people due to the defined ROI and the template which represents an upright upper body in Figure 4.4 being optimised for walking/standing people. Figure 4.8b shows one of the people in the images is not detected due to an "incorrect" body posture. The solution to most of these issues is provided by the Bayesian tracker, fusing the detectors to compensate for the fact that one of them might produce false negatives or a person is not in its field of view, and via smoothing the trajectories using a Kalman Filter and a constant velocity model. However, this becomes an issue when both detectors are not able to pick-up the people around the robot which can easily happen when they are sitting in a wheelchair like in the *Haus der Barmherzigkeit* environment (see Section 1.3.1). Many approaches have been tried to overcome this, like lowering the lower limit for upper body detections or developing a laser based wheelchair detector (Beyer et al. 2016) but none of them were showing any improvement at the time of writing this thesis. The leg detector, on the other hand, is prone to false positives due to chairs and table legs which is countered by a simple filter that removes all detections that correspond to obstacles in the known map, but still causes the creation of fake humans and therefore influences the actual performance of every human-aware navigation approach. This can be addressed using so-called no detection zones in this filter where areas with high false positive rates can be blacked out to not allow for any tracks to be created. This, however, requires labour intensive manual annotation of the environment.

Another feature that is missing is the re-identification of previously seen persons to allow for the loss of a track and generate more complete QTC state chains, but similar to the problem with detecting wheelchairs, at the time of writing this thesis no adequate solution has been found that would produce reliable results given the hardware and environmental constraints. One could also argue that it is not necessary to re-identify people when they re-enter the FoV as each interaction should be treated separately. However, this missing component makes it impossible to create different models for different groups of people (e.g. staff and patients in a care home) even though it would be possible to use different models for different people depending on the amount of training data.

Concluding from the above statements, the presented tracking pipeline has its limitations, but is fast and reliable enough to generate the qualitative state chains necessary for online classification and behaviour generation.

4.5 Summary

This chapter introduces the means of ***creating QTC state chains from the on-board sensors of a robot using on-board processing only [Obj. 3.2]***. Tailored to *Linda*, an upper body detector using the head mounted RGB-D camera and a laser based leg detector were chosen and in the case of the upper body detector implemented into ROS. These detections however have to be fused to generate a single detection for each human and subsequent detections have to be combined into tracks to enable the generation of QTC states that need at least two consecutive positions of the human to be created. This has been solved using a Bayesian tracking framework with NN or NNJPDA data association using a particle, Extended, or Unscented Kalman Filter for prediction. For the very linear prediction problem at hand a simple Kalman Filter would have sufficed but there is no trade-off computation wise in using a different version that is also able to deal with non-linear data.

The output of the tracking framework and the AMCL based self-localisation of the robot is then used to create *valid* QTC state chains that do not contain illegal transitions or equal adjacent states and therefore conform with the definition of legal state chains. These ***state chains are generate incrementally and online [Obj. 3.1]*** and can be used to classify the observed interaction type using pre-trained HMMs or for online classification or behaviour generation as described in the following. Hence, the conducted experiment showed that ***QTC based probabilistic models abstract from the environment, the sensors used, the agents involved, and the observers perspective [Obj. 1.1]*** well enough to use a model trained with data from the "Bristol experiment" to classify data generated by the robot's sensors.

To summarise, the ***autonomy [Obj. 3]*** objectives of ***working in real-time [Obj. 3.1]*** only ***relying on on-board sensors and processing [Obj. 3.2]*** have been addressed in this chapter with the perception and state generation showing that this part of the system is able to achieve said objectives. Additionally, reinforcing the results of Chapter 3,

the experiment directly addressed the ***robust qualitative interaction models [Obj. 1]*** objectives by showing that the HMM trained on a data set ***abstracts from the metric environment representation [Obj. 1.1]***, only ***represents the qualitative character of motions [Obj. 1.2]*** and, therefore, has the ***ability to generalise [Obj. 1.4]*** well enough to be used for the classification of the collected robot data.

On a side note, the whole system described in this chapter is very modular which means that detectors can easily be added or removed from the tracker embracing the distributed nature of ROS systems. Moreover, the QTC generation is part of a larger library that is able to produce numerous QSRs and the HMM generation and classification is able to deal with any kind of QSR. This means that the system can either be used as a whole or in parts as in for example Lightbody et al.'s (2015) work.

—With regard to robots, in the early days of robots people said, 'Oh, let's build a robot' and what's the first thought? You make a robot look like a human and do human things. That's so 1950s. We are so past that.

Neil deGrasse Tyson, Astrophysicist/Author

5

Constraint based HRSI Behaviour Generation

After formulating the model and showing that it can be used on data produced by the robot's sensors, the next step towards an autonomous system is behaviour generation using the underlying model. When it comes to modelling and classifying activities, QSRs like QTC are a convenient way of abstracting from all the low-level behaviour that is not relevant to capture the essence of the interaction, for example it does not matter if the robot avoids a human at $0.55^{m}/_{s}$ or $0.54^{m}/_{s}$. When trying to generate behaviour for a mobile robot, on the other hand, explicit knowledge about this low-level movement is crucial. However, the exact repetition of the executed movement commands might not be possible, or indeed necessary, as long as the executed movement generates the correct qualitative state. Of course, this implies that the qualitative description models all the parameters that are important for the criteria one wants to optimise. In the presented case of human-aware navigation and human-robot joint motion, this comes down to creating legible trajectories for the robot that are also perceived as safe and, therefore, comfortable. Hence, this chapter addresses the problem of how to translate a, e.g. QTC_{BC} state like

$(--++)$, into velocities that can be sent to the wheels of the robot. Assuming that q_1 and q_3 describe the movement of the human and are known, q_2 and q_4, i.e. $(-+)$ in the given example, have to be translated into actual command velocities, which make the robot approach the human while also moving to the right of the connecting line between them.

Current state-of-the-art navigation approaches mostly rely on a combination of global planning using Dijkstra or A* and local planning (e.g. Fox et al. 1997) to achieve robust navigation in the face of static and dynamic obstacles. In the implementation of the ROS navigation stack that is not only used on *Linda* but also in the entire STRANDS project and is the default navigation stack for each ROS robot, the global path planning is done via Dijkstra using a global costmap. This costmap contains all the static obstacles that were present during the creation of the map and the obstacles currently observed by the sensors, as lethal obstacles (as described in Section 5.2.2). Due to the robot being represented by its centre point and a footprint, these obstacles are inflated by the radius of the circumscribed footprint to assure that the robot never ends up inside an obstacle. The costs are then used as movement costs in the Dijkstra algorithm and hence create a path that tries to keep as much distance to obstacles as possible while still optimising for distance travelled. This form of path planning, however, is slow and costly (up to several seconds depending on the size of the map, the distance of the goal, and the computational power) because it considers the whole map which can quickly amount to several million grid cells. Hence, constant replanning to avoid dynamic obstacles is not feasible.

For the current ROS navigation stack to achieve computationally cheap dynamic obstacle avoidance (running at $20hz$ by default), a smaller version of this costmap, which only represents a square of a few m^2 around the robot, is used in a sampling based approach. This so-called Dynamic Window Approach (DWA) local planner by Fox et al. (1997) uses this metric costmap and translates it into the velocity space to allow for dynamic sampling of future trajectories as can be seen in Figure 5.1. This velocity space is further restricted by the so-called dynamic window which describes only the velocities that can be reached in a predefined time frame Δt, given the acceleration limits of the robot.[29] This Δt is referred to as *sim time* and defines how far the planner should plan

[29]These limits depend on the robot's motors and are assumed constant. For *Linda* the limits are $x = \pm 0.8^{m}/_{s^2}$ and $\theta = \pm 3.14^{rad}/_{s^2}$.

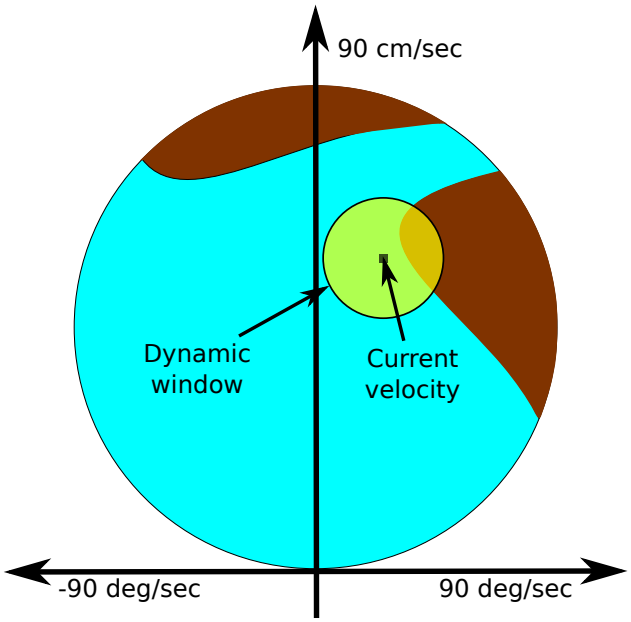


Figure 5.1: *The velocity space representation of a metric local costmap of a corridor as it is used by the Dynamic Window Approach (DWA) local planner. This shows velocities in x, y, θ direction. The blue area represents admissible velocities and the brown area represents velocities at which the robot would collide with an obstacle. The yellow circle discribes the dynamic window which contains velocities that can be reached in a pre-defined time Δt given the acceleration limits of the robot.*

ahead. A predefined number of velocity samples in each direction possible is uniformly distributed across the window and then scored based on several different critique functions, e.g. goal-directedness, global path following, etc. which all assign a cost value to each sample. The sample with the lowest costs is then chosen to be executed till Δt is up.

In contrast to the metric local costmap, QTC describes movement as velocities already where the symbols (q_2, q_4) describe an interval $\xi = [\overrightarrow{\nu}_{\min}, \overrightarrow{\nu}_{\max}]$ constraining the possible directions of the robot's velocity vector $\overrightarrow{\nu_r} \in \xi$ which means that this representation can be used as an additional critique function without having to transform it to metric space first. Therefore, this chapter introduces so-called *Velocity Costmaps* which describe the constraints ξ and are used in conjunction with Fox et al.'s (1997) DWA local planner to achieve safe, legible, and sociable human-aware navigation. This way, the local planner takes care of the generation of the command velocities to be send to the robot's motors where the Velocity Costmaps ensure that only trajectories that produce the correct QTC state are generated.

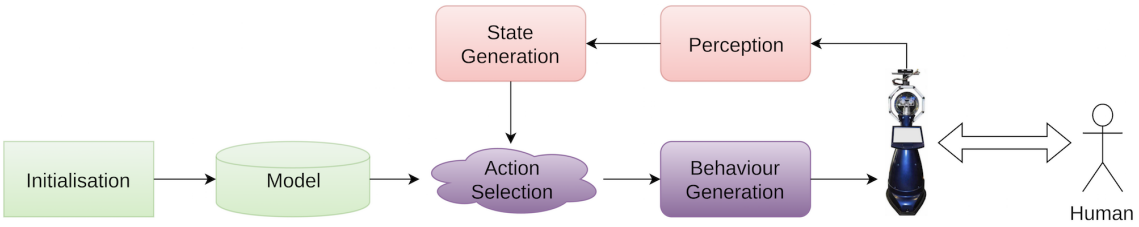


Figure 5.2: *This chapter's contribution towards the system shown in Figure 1.1 building on the partial system in Figure 4.2. The interaction of human and robot is observed using the robot's on-board sensors and automatically transformed into valid QTC state chains as described in Chapter 4 and used together with the model of the observed previous interactions to generate behaviour for the robot. This approach relies on a-priori knowledge of the interaction type, e.g. pass-by or path crossing, due to the missing classification component and uses hand crafted rules for the preliminary action selection process which is replaced in the following chapter. The novel contributions to the system are highlighted in purple where only the behaviour generation is part of the final system. For the first time the loop of sensing and acting has been closed in a purely reactive manner.*

Chapter Contribution The main contribution of this chapter, therefore, is the generation of movement commands for a mobile robot from QSRs, representing a novel approach to human-aware navigation. Using the high-level QTC based representation from Chapter 3, low-level command velocities that can be send to the robot's wheels are generated using so-called *Velocity Costmaps* which restrict the sample space of the DWA local planner to generate trajectories that produce the desired QTC state. These Velocity Costmaps introduced in this chapter, on the one hand, produce ***trajectories that are safe and also perceived as safe by the human [Obj. 2.2]*** interaction partner and, on the other hand, are still ***task efficient by minimising the travel time towards the goal [Obj. 2.3]***. These costmaps are based on QTC states that are derived from either hand crafted rules as done in this chapter or a learned joint probability table as shown in Chapter 6 which allows to incorporate human judgement via Learning from Demonstration (LfD). By using the DWA local planner which uses a search space based on the acceleration limits of the robot, the generated movements are not only safe and task efficient but are also ***tailored to the used hardware [Obj. 3.3]*** and thereby the planner only produces velocities that are possible to achieve in the given time. The contribution of this chapter to the overall proposed system can be seen in Figure 5.2 which builds on Figure 4.2 and extends it with the preliminary action selection module used for the experiments presented in this chapter and later on replaced by an autonomous system and the behaviour generation.

In the following, the approach based on hand crafted rules is evaluated in simulation and in a real-world experiment using the robot *Linda*, showing how to incorporate knowledge about HRSI encoded in QTC_{BC} into a concise model for trajectory sampling in velocity space. This does not make use of any learned models but only uses a conceptual model of the interaction.

The presented approach has been described in a conference paper that has been published at the *IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), 2016*. Please refer to Appendix A.3 for the author's contributions to this work.

5.1 The Dynamic Window Approach Local Planner

In order to describe the behaviour generation using *Velocity Costmaps* it is important to introduce the concepts underlying the Dynamic Window Approach (DWA) local planner and, therefore, make the actual generation of command velocities clearer. In its original formulation by Fox et al. (1997), the DWA planner functioned as a global planner for a synchro-drive robot. In its current implementation in the ROS navigation stack, however, the DWA became a local planner that moves between goals that are placed along the path generated by the global planner. This leads to a more goal directed and more informed robot movement then just sampling velocities. Nevertheless, its capabilities of avoiding dynamic obstacles in close vicinity to the robot and the computationally cheap sampling using a model directly derived from the motion parameters of the robot made the DWA planner an essential component of almost every mobile robot that uses ROS. Hence, despite it being first presented in (1997), it is constantly updated by the ROS community and therefore can still be considered state-of-the-art in robotic navigation for exactly these reasons. This section introduces the most important principles underlying the DWA local planner during the time of the creation of this thesis.

The ROS version of the DWA planner is able to sample velocities in x, y and θ which allows to create trajectories for holonomic robots. *Linda* on the other hand is a non-holonomic robot, i.e. it has a differential drive, which does not allow it to move along the y-axis. This restricts the sample space to 2 dimensions which are equal to a polar coordinate system with the robot at its centre and $\theta = 0$ facing forwards. For this reason,

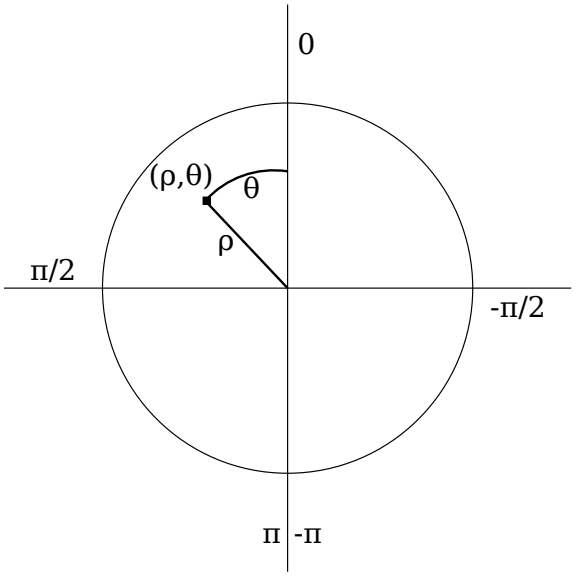


Figure 5.3: *The polar coordinate and velocity space for a non-holonomic robot with the robot at its centre. ρ describes the distance from the centre in meters and θ the distance from 0 in radians.*

when speaking of velocities, it refers to (ρ, θ) where ρ is the forward velocity or distance from the centre of the polar coordinate system and θ the angular velocity or the angular difference from 0 as can be seen in Figure 5.3. Hence, (ρ, θ) also describe a discrete point in the polar coordinate space which is later on used to look up the allowed velocities in the *Velocity Costmaps*. Due to this restriction to 2 dimensions the following descriptions of the planning approach will be restricted to 2D but can of course easily be extended to 3 dimensions.

As mentioned above, the DWA has a certain planning horizon Δt for which it samples velocities. Given the purpose of reaching a certain goal in the environment, it has to create a sequence of n velocities (ρ_i, θ_i) for $[\Delta t_0, \Delta t_n]$ until it reaches the goal. This would mean that this approach quickly becomes intractable for small Δt and/or large n. Even if the DWA is not used as a global planner any more, this still holds true even for partial goals along the global path. To prevent this sample space explosion, the DWA only considers the first time interval Δt_0 and assumes a constant speed, i.e. zero acceleration, for the following $[\Delta t_1, \Delta t_n]$ as defined by Fox et al. (1997). After each time interval this search is continued for the next interval until the goal is reached.

5.1.1 Dynamic Window

Despite all the optimisation already undertaken, e.g. only sampling for the next Δt and only working in 2 dimensions, this process can still be optimised further. The dynamic window (Fox et al. 1997) of the DWA defines a small subspace of all the possible velocities based on the acceleration limits of the robot and the so-called admissible velocities. As defined by Fox et al. (1997), admissible velocities refers to velocities that allow the robot to stop before it reaches an obstacle. Assuming that for a velocity (ρ, θ) the term $dist(\rho, \theta)$ represents the distance to the closest obstacle and let $\dot{\rho}_b$ and $\dot{\theta}_b$ be the accelerations for breakage, i.e. maximum deceleration until reaching 0 velocity, then the set of admissible velocities $\mathbf{V}_a$ is defined as

$$\mathbf{V}_a = \left\{ (\rho, \theta) : \rho \leq \sqrt{2 \cdot dist(\rho, \theta) \cdot \dot{\rho}_b} \wedge \theta \leq \sqrt{2 \cdot dist(\rho, \theta) \cdot \dot{\theta}_b} \right\} \tag{5.1}$$

Hence, $\mathbf{V}_a$ is the set of velocities that allow the robot to break before colliding with the closest obstacle.

Additionally, in order to take the limited acceleration of the motors of the robot into account and to reduce the overall search space, the actual dynamic window is defined as $\mathbf{V}_d$ which is the set of velocities that are reachable in Δt given the accelerations $\dot{\rho}$ and $\dot{\theta}$ and the actual velocity of the robot (ρ_a, θ_a).

$$\mathbf{V}_d = \left\{ (\rho, \theta) : \rho \in \left[\rho_a - \dot{\rho} \cdot \Delta t, \rho_a + \dot{\rho} \cdot \Delta t \right] \wedge \theta \in \left[\theta_a - \dot{\theta} \cdot \Delta t, \theta_a + \dot{\theta} \cdot \Delta t \right] \right\} \tag{5.2}$$

For *Linda* these acceleration limits are $\dot{\rho} = \pm 0.8^{m}/_{s^2}$ and $\dot{\theta} = \pm 3.14^{rad}/_{s^2}$.

Thus, the resulting search space $\mathbf{V}_r$ can be defined as the intersections of the entire space $\mathbf{V}_s$, the admissible velocities $\mathbf{V}_a$, and the dynamic window $\mathbf{V}_d$

$$\mathbf{V}_r = \mathbf{V}_s \cap \mathbf{V}_a \cap \mathbf{V}_d \tag{5.3}$$

To put this into a graphical context, Figure 5.1 shows the entirety of $\mathbf{V}_s$ in the outer bubble, the admissible velocities $\mathbf{V}_a$ are represented by the blue area, and the dynamic window $\mathbf{V}_d$ as the yellow area. The final search space $\mathbf{V}_r$ is represented by the blue area

overlayed by the yellow area. The brown area overlayed by the yellow area is not part of this search space.

To optimise the sampling of velocities even further, not the entire search space $\mathbf{V}_r$ is sampled but only a few representative sample points are chosen which are distributed uniformly over $\mathbf{V}_r$. In *Linda*'s case the number of samples to explore the velocity space in direction of ρ is 3 and in direction of θ is 20. Hence, the set of trajectories sampled is defined as

$$\mathfrak{T} = \{(\rho_i, \theta_j) : \rho_i \in \mathbf{V}_r, \theta_j \in \mathbf{V}_r \wedge 1 \leq i \leq n, 1 \leq j \leq m\} \tag{5.4}$$

where $n = 3$ is the number of samples in ρ direction and $m = 20$ the number of samples in θ direction. These trajectories $\mathfrak{t}_i \in \mathfrak{T}$ are the basis for the following sampling step.

5.1.2 Trajectory Sampling

In order to achieve goal directed movement of the robot, the DWA scores the sampled trajectories $\mathfrak{t}_i \in \mathfrak{T}$ based on several distinct so-called critique functions. Hence, for each sampled trajectory $\mathfrak{t}_i$ the sum of all these critique functions is computed

$$\gamma_i(\mathfrak{t}_i) = \sum_{j=1}^{N} \omega_j c_j(\rho_i, \theta_i) \tag{5.5}$$

where $c_j \in \boldsymbol{\mathcal{C}}$ represents a critique function out of the set of critique functions $\boldsymbol{\mathcal{C}}$, $\omega_j \in \boldsymbol{\omega}$ represents an associated weight out of the set of weights $\boldsymbol{\omega}$, and $N = |\boldsymbol{\mathcal{C}}| = |\boldsymbol{\omega}|$.

The optimal trajectory

$$\mathfrak{t}^\star = \arg\min_{\mathfrak{t}_i \in \mathfrak{T}} \gamma_i(\mathfrak{t}_i) \tag{5.6}$$

is then chosen for execution until Δt is up and the next sampling step is executed.

At the time of writing this thesis, the set of critique functions $\boldsymbol{\mathcal{C}}$ contains the following scoring mechanisms.

Path Distance scores the trajectory samples $\mathfrak{t}_i \in \mathfrak{T}$ based on their Euclidean distance to the path generated by the global planner. Hence, the higher the weight on this critique function, the more closely the robot sticks to the global path. Depending on the algorithm

used to find this global path this can be more or less desirable. Given the use of Dijkstra's algorithm as described above, this results in a more energy efficient robot navigation by following this shortest path.

Path Align similar to Path Distance, but calculates scores based on the orientation alone to assure that the robot faces in the direction of future travel along the path.

Goal Distance scores the trajectory samples $\mathfrak{t}_i \in \mathfrak{T}$ based on their Euclidean distance to the next intermediate goal along the global path. As discussed earlier, DWA is not used as a global planner any more as originally proposed by Fox et al. (1997), but as a local planner that moves between intermediate goals that have been created along the path calculated by the global path planner. Hence, the weighting for this critique function determines the goal directedness of the chosen trajectory $\mathfrak{t}$. Usually, the path distance score receives a lower weighting then the goal distance score to assure that the robot travels towards the goal while sticking to the path if possible. The other effect of this scoring function is to ensure maximum velocity where possible, due to the fact that the velocity samples that represent higher velocities are closer to the goal when transformed into Euclidean space.

Goal Align similar to Goal Distance, but calculates scores based on the orientation alone to assure that the robot faces in the direction of future travel towards the goal.

These scoring functions are essential to the robot's navigation, as without it, it would just move into free space and stop. In the original work by Fox et al. (1997), the velocity space $\mathbf{V}_r$ is also smoothed to achieve larger side clearance when passing obstacles and to achieve straighter forward trajectories and rounder arcs when moving around obstacles. In its current implementation, however, the DWA planner relies on following the global path which, using Dijkstra on a global costmap, achieves sufficient side clearance and straight/round trajectories where possible. The following section introduces a new critique function into the set of critique functions $\boldsymbol{\mathcal{C}}$ which is based on QTC. Thereby, the DWA planner will still be responsible for generating the movement commands but will be restricted to only allow $\mathfrak{t}_i \in \mathfrak{T}$ which achieve a low score using this new critique function.

5.2 Velocity Costmaps

In order to use QTC in a critique function for the DWA local planner, the mentioned *Velocity Costmaps* are generated which are occupancy map representations of the costs in velocity space based on the desired QTC state (see Figure 5.6 and 5.7). This section introduces the concept of velocity costmaps, its grounding in QTC, and how they are combined with the DWA planner to produce robot motion.

5.2.1 Action Selection

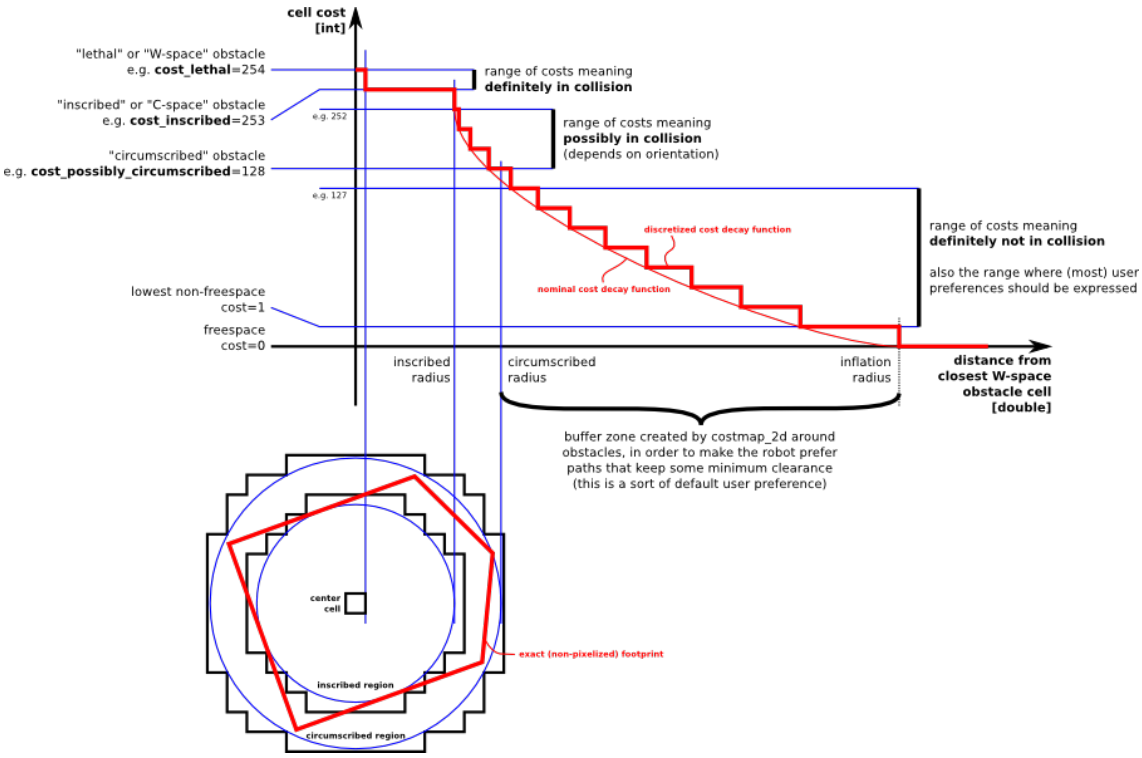
Disregarding the problem of activity recognition in real-time on a mobile robot for now, the first hurdle to overcome for behaviour generation is finding the best action for the robot to perform at any given time t. Assuming that for every QTC_{BC} state $(q_1\ q_2\ q_3\ q_4)$ the two symbols $(q_1\ q_3)$ that describe the human motion are known, the problem of finding the best action for the robot $(q_2\ q_4)$ to execute can be transformed into a simple look-up

$$\boldsymbol{\Omega} \rightarrow \boldsymbol{\mathcal{A}} \tag{5.7}$$

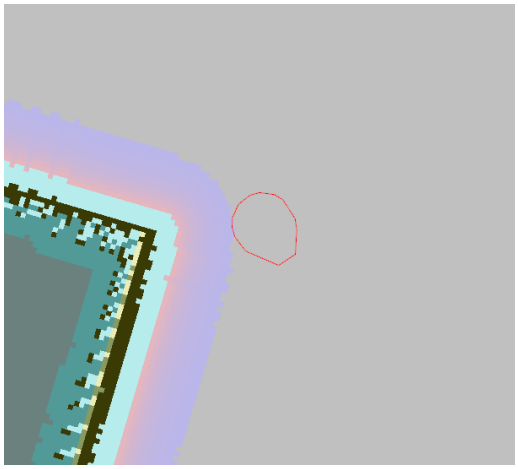
where $\boldsymbol{\Omega}$ represents the set of all possible states of the human and $\boldsymbol{\mathcal{A}}$ represents the set of all possible actions of the robot. For the experiment presented in Section 5.3, the type of interaction will be given to remove uncertainty from classification and, therefore, $O \in \boldsymbol{\Omega}$ can easily be extracted from the current QTC_{BC} state of human and robot. Moreover, the set of possible robot actions $\boldsymbol{\mathcal{A}}$ can be model by hand by defining rules $O_j \rightarrow S_i$ for the most common mappings in Equation 5.7. These rules are not exhaustive but by simply using the last known mapping for uncommon mappings that have not been defined these edge cases can be circumvented. Chapter 6 shows how these mappings can be learned and how the current interaction type can be classified but to show that this approach of behaviour generation yields trajectories which approximate the correct QTC state, the uncertainty of learned mappings and online classification has been removed.

5.2.2 Costmap Generation

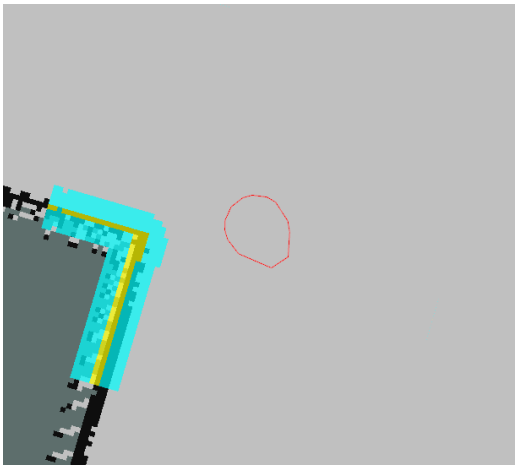
As described in Equation 5.5, each trajectory $\mathfrak{t}_i \in \boldsymbol{\mathfrak{T}}$ is assigned a cost value $\gamma_i(\mathfrak{t}_i)$ based on the sum of several independent critique functions $c_i \in \boldsymbol{\mathcal{C}}$ and their weights $\omega_i \in \boldsymbol{\omega}$.



(a) *Compared to the cost values used for Occupancy Maps* $[0, 100]$*, the costs in costmaps are* $[0, 255]$ *where* 255 *is reserved for* unknown *grid cells and* 254 *for* lethal *costs. These lethal costs are asigned at the exact position of obstacles and are then inflated. The first inflation step sets the costs surrounding lethal obstacles to* 253 *based on the* inscribed robot footprint *because the robot would be in collision with these obstacles if the center of the robot would be moved into this area. The next inflation range* $[128, 252]$ *is based on the* circumscribed robot radius *which represents a possible collision if the robot would assume a certain rotation. The cost range* $[1, 127]$ *is used as a smooth transition to the circumscribed costs based on an exponential decay function.* 0 *is used for the remainder of the cells representing free space. Image taken from:* Robot Operating System wiki: Costmap 2D.



(b) *The global costmap showing the costs generated by Figure 5.4a and the robot footprint in red.*



(c) *The local costmap only using the inscribed costs from Figure 5.4a in a small rectangle around the robot.*

Figure 5.4: *The ROS cost generation function and an example of a global and local costmap.*

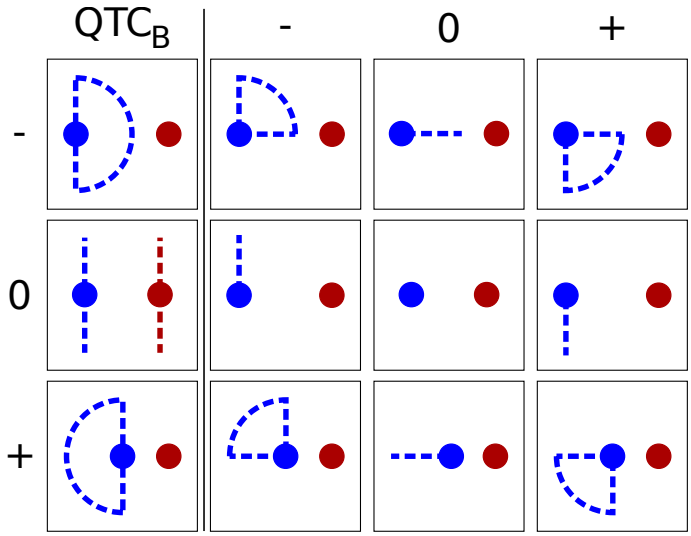


Figure 5.5: *The velocity costmap prototypes with the robot in blue and the human in red. The area enclosed by the partial blue circle or the blue line represents the low cost area ξ, everything outside is assigned the lethal cost value. The human can have any possible QTC state (except for QTC_B,0 where special behaviour is required). The row denotes the distance constraint and the column denotes the side contstraint. These zones are directly inspired by the original definition of QTC by Van de Weghe (2004).*

Subsequently, the trajectory with the lowest score is chosen in Equation 5.6 and executed until the *sim time* Δt is up after which the whole sampling process will be repeated. To ensure human aware navigation, *Velocity Costmaps* are included into $\boldsymbol{\mathcal{C}}$. The ROS implementation of the DWA local planner works on so-called occupancy maps that describe metric space as a grid-based representation where each pixel represents the costs of moving the centre of the robot to that pixel. Each pixel can assume the value $p_{x,y} \in [0, 100]$ where 0 represents free space and 100 represents a lethal obstacle. Normally, these occupancy maps are based on the underlying global metric map and represent obstacles in the static Simultaneous Localisation And Mapping (SLAM) map based on work by Grisetti et al. (2005) and (2007) or obstacles observed via the laser scanner. These obstacles are assigned lethal costs and are then inflated by the robot radius to make sure it never ends up inside an obstacle as can be seen in Figure 5.4. The local costmap shown in Figure 5.4c is then transferred into the velocity space for sampling. If during this sampling process any of the critique functions produces lethal costs, this trajectory is immediately discarded regardless of the specified weight of the critique function that produced them. As stated in the introduction of this chapter, QTC is a representation of velocity which makes this transformation from Cartesian to velocity space unnecessary. Therefore, the generated *Velocity*

Table 5.1: *The δ and α values to compute the Velocity Costmaps inspired by Figure 5.5. α is used on either side of λ resulting in a free space of $\pi/2$ if alpha = $\pi/4$. $\pi/32$ has ad-hoc been defined to represent movement on a straight line and proved suitable in subsequent tests.*

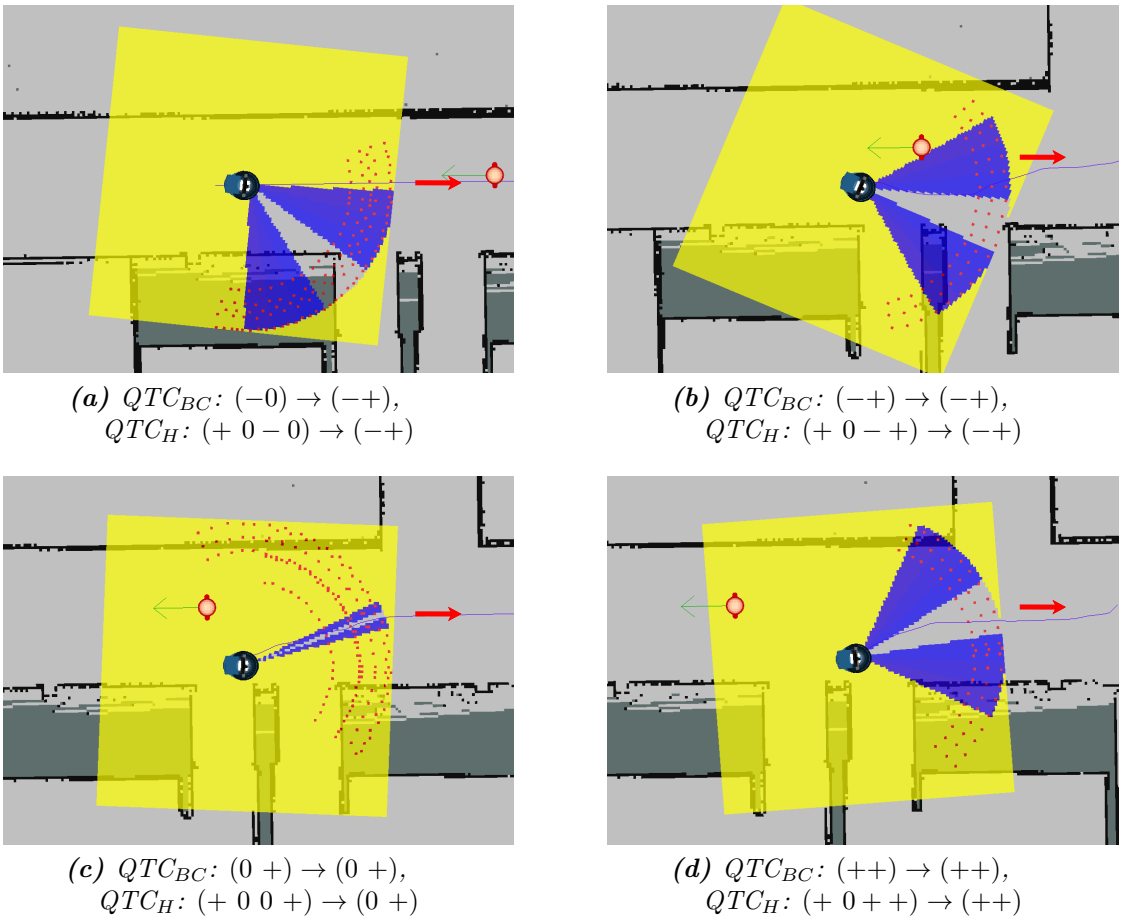
	QTC_B	−	0	+	
−	0	$-\frac{\pi}{4}$	0	$\frac{\pi}{4}$	δ
	$\frac{\pi}{2}$	$\frac{\pi}{4}$	$\frac{\pi}{32}$	$\frac{\pi}{4}$	α
0	$\pm\frac{\pi}{2}$	$-\frac{\pi}{2}$	0	$\frac{\pi}{2}$	δ
	$\frac{\pi}{32}$	$\frac{\pi}{32}$	0	$\frac{\pi}{32}$	α
+	π	$-\frac{3\pi}{4}$	π	$\frac{3\pi}{4}$	δ
	$\frac{\pi}{2}$	$\frac{\pi}{4}$	$\frac{\pi}{32}$	$\frac{\pi}{4}$	α

Costmap is a Cartesian representation of the polar space described by the desired QTC state and can directly be used for trajectory sampling. Figure 5.5 shows the prototypes used to create these occupancy maps for the QTC states.

Given the current position of the human (ρ_h, θ_h) relative to the robot, the angle $\lambda = \theta_h + \delta$ is computed where δ depends on the desired QTC state of the robot. If the polar representation (ρ_p, θ_p) of the pixel $p_{x,y}$ lies within the allowed area of $\xi_\alpha = \{\theta_p : \lambda - \alpha \leq \theta_p \leq \lambda + \alpha\}$ and $\xi_\varrho = \{\rho_p : \varrho_{\min} \leq \rho_p \leq \varrho_{\max}\}$ where α an ϱ, like δ, also depend on the desired QTC state (see Table 5.1), for each pixel $p_{x,y}$ the following cost value is computed

$$cost\,(p_{x,y}) = \boldsymbol{a}_{||\theta_p|-|\lambda||\cdot|\boldsymbol{a}|};\ \boldsymbol{a} = \{a_i : a_i \in \mathbb{N}\} \tag{5.8}$$

with $\boldsymbol{a}$ being a strictly increasing set of low costs. By initialising the occupancy map with the lethal cost value of 100 for all pixels $p_{x,y}$, given $p_{x,y} \notin \xi$, every trajectory sample $\mathfrak{t}_i \in \mathfrak{T}$ that does not fall within the allowed area $\mathfrak{t}_i \notin \xi$ will have lethal costs in the subsequent sampling process and therefore not be considered for execution regardless of the used weight ω_j in Equation 5.5. Looking at Fig. 5.6a as an example, given the desired QTC state of $(- +)$ approaching and moving to the right, the angle is computed as $\lambda = \theta_h + \frac{\pi}{4}$ which results in the allowed sample space of $\xi_\alpha = [\lambda - \frac{\pi}{4}, \lambda + \frac{\pi}{4}]$. Assuming that the human is directly in front of the robot $\theta_h = 0.0$, these two values are $\lambda = \frac{\pi}{4}$ and $\xi_\alpha = [0, \frac{\pi}{2}]$ as it is shown in the top right corner of Fig. 5.5. The low cost areas are set to $\boldsymbol{a} = \{0, 5, 10, 15\}$ in all cases where $q_4 \neq \emptyset$ (see blue areas in Fig. 5.6 and 5.7) to increase the avoidance manoeuvre by assigning lower costs to samples in the centre of this region. The resulting costs are then weighted and summed with the remainder of the



(a) *QTC_{BC}:* $(-0) \rightarrow (-+)$, *QTC_H:* $(+\ 0 - 0) \rightarrow (-+)$

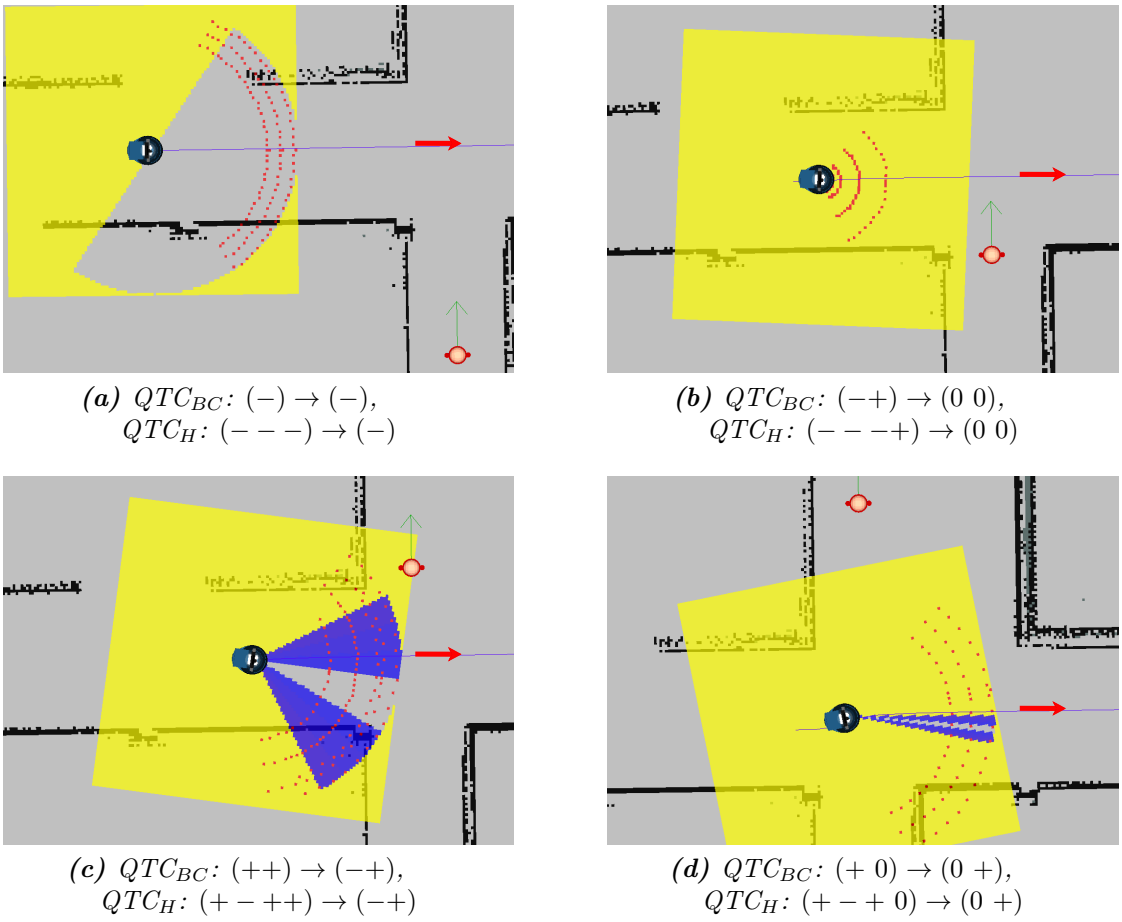
(b) *QTC_{BC}:* $(-+) \rightarrow (-+)$, *QTC_H:* $(+\ 0 - +) \rightarrow (-+)$

(c) *QTC_{BC}:* $(0\ +) \rightarrow (0\ +)$, *$QTC_H$:* $(+\ 0\ 0\ +) \rightarrow (0\ +)$

(d) *QTC_{BC}:* $(++) \rightarrow (++)$, *QTC_H:* $(+\ 0 + +) \rightarrow (++)$

Figure 5.6: *Example of a pass-by interaction. Blue figure: robot, red figure: human. The partial circles (with radius* $\max(\rho)$*) inside the yellow square represent a Cartesian representation of the polar space used for the Velocity Costmap (see Figures 5.5 and 5.8a). Blue: low cost areas* $\{5, 10, 15\}$ *to increase avoidance manoeuvre (see Equation 5.8), yellow: lethal costs of* 100*, free space:* 0 *costs, red dots: generated samples* $\mathfrak{t}_i \in \mathfrak{T}$*. Captions represent the mapping* $\boldsymbol{\Omega} \rightarrow \boldsymbol{\mathcal{A}}$ *of observed human state to learned robot state in* QTC_{BC} *and* QTC_H *(see Chapter 6). The red arrow points to the robot's goal which will be of significance in Chapter 6.*

critique functions in Equation 5.5. Given this representation, it is also possible to restrict the minimum and maximum speed of the robot using ϱ in addition to the angular speed but this is currently only used for the QTC_B state (0) to allow the robot and human to travel in the same direction with equal velocity. Hence, ϱ is not shown in Table 5.1 as it is only used in this case where it is set to $\varrho = \rho_h \pm \nabla$ with $\nabla = 0.05^{m}/_{s}$. In all other cases it ranges from $0^{m}/_{s}$ to $0.55^{m}/_{s}$ the maximum translational velocity of the robot. However, it could be used when generating behaviour for the non-simplified version of QTC_B using the tuple $(q_1\ q_2\ q_\nu)$. Similarly, λ could be adjusted according to q_α from the full QTC_C tuple $(q_1\ q_2\ q_\nu\ q_3\ q_4\ q_\alpha)$, but all this will remain future work.



(a) QTC_{BC}: $(-) \rightarrow (-)$, QTC_H: $(- - -) \rightarrow (-)$

(b) QTC_{BC}: $(-+) \rightarrow (0\ 0)$, QTC_H: $(- - -+) \rightarrow (0\ 0)$

(c) QTC_{BC}: $(++) \rightarrow (-+)$, QTC_H: $(+ - ++) \rightarrow (-+)$

(d) QTC_{BC}: $(+\ 0) \rightarrow (0\ +)$, QTC_H: $(+ - +\ 0) \rightarrow (0\ +)$

Figure 5.7: *Example of a path crossing sequence where "$\rightarrow$" represents the mapping from observed state to generated constraint. The transition from QTC_B 5.7a to QTC_C 5.7b causes a state change in the robot even though the human's state is unchanged. See Figure 5.6 for a detailed explanation of the symbols and see Figures 5.5 and 5.8a for the theoretical background.*

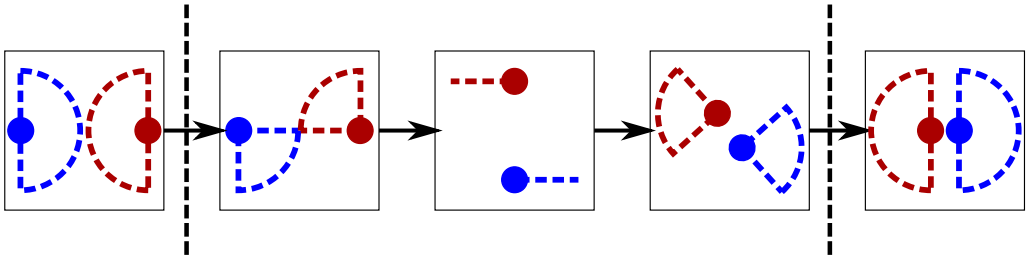
Finally, to overcome the issue of what behaviour to choose when these costmaps do not allow for any movement towards the goal, e.g. only allowing backwards movement even though the robot cannot move backwards or not allowing any movement at all to have the robot stop and wait until the human has passed or a Velocity Costmap is generated that allows goal directed movement again, $\gamma_i(\mathbf{t}_i) = 0$ costs are assigned to trajectories that have no translational movement $\mathbf{t}_i = (0, \theta)$. Hence, the robot is always allowed to stop and to turn on the spot as $\theta \in \boldsymbol{\theta}$ can have any possible value.

5.3 Experiment and Evaluation

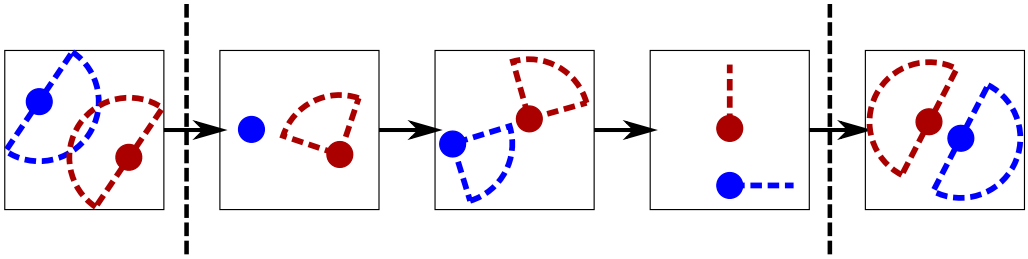
To evaluated the functionality and soundness of the proposed velocity costmaps for behaviour generation, an Oz of Wizard (see Section 2.4.1) experiment in simulation and as a proof of concept using *Linda* were conducted. As mentioned earlier, this set-up assumes that the interaction type is given before the start of the interaction and uses hand-crafted predefined rules which can be seen in Equation 5.9 for the pass-by interaction. $(\emptyset, \emptyset)$ represents the special case of undefined robot behaviour and will be replaced by the last valid QTC state for costmap generation. These rules represent prototypical conceptual pass-by and path crossing encounters and are not based on any data but on geometric reasoning about the given interaction types. The resulting rules are used for a look-up of the "correct" robot behaviour and are shown in Figure 5.8.

$$f(q_1, q_3) = \begin{cases} (-, \emptyset) & \text{if } q_1 = - \wedge q_3 = \emptyset \\ (-, +) & \text{if } q_1 = - \wedge q_3 \neq \emptyset \\ (0, +) & \text{if } q_1 = 0 \wedge q_3 = + \\ (+, +) & \text{if } q_1 = + \wedge q_3 \neq \emptyset \\ (+, \emptyset) & \text{if } q_1 = + \wedge q_3 = \emptyset \\ (\emptyset, \emptyset) & \text{otherwise} \end{cases} \tag{5.9}$$

For the simulation experiment, an office environment of $\sim 5,000m^2$ resembling one of the University of Lincoln's buildings was constructed, using its main corridor for the interaction between human and robot. The STRANDS project follows a mixed navigation approach using a combination of a metric and a topological map as described by Pulido Fentanes et al. (2015) and shown in Section 1.3.1, thus, a topological edge along a $12m$ long and $2.6m$ wide straight stretch and another $15m$ long edge passing a 4-way crossing (see Figures 5.6 and 5.7) was created in this corridor. No obstacles except walls and the physical model of the simulated human are present in these parts of the environment. The real-world environment comprised a $8m \times 8m$ area of the L-CAS office that was cleared of all obstacles except the human interaction partner, using a $5.5m$ long topological edge passing through the centre of the free area (see Figure 5.9). The two participants were the author of this work and another PhD student working on the STRANDS project



(a) *A list of all the QTC states used to generate the robot behaviour in the pass-by interaction. Equation 5.9 shows the rules to generate this behaviour. The major difference here is that the second square shows the human going to the right of the connecting line where in the actual rule it only states that the human needs to approach and the state has to be QTC_C. Hence the robot would always go to the right as this is the only action it knows. In the conducted experiment, the human just walks straight regardless. See Figure 5.6 for the resulting interaction.*



(b) *A list of all the QTC states used to generate the robot behaviour in the path crossing interaction. The robot travels from left to right while the human walks from bottom to top. See Figure 5.7 for the resulting interaction.*

Figure 5.8: *Prototypical interactions encoded in Velocity Costmaps with the observed human state in red and the generated robot state in blue. The dashed line represents instants in time where the distance threshold d_s is crossed, triggering a behaviour change in the robot by switching from QTC_B to QTC_C or vice-versa. The arrows represent possible transitions but for the actual implementation transition probabilities were disregarded and a simple look-up shown in Equation 5.9 was used based on Equation 5.7.*

which means that both were well acquainted with the robot, the goal of the study, and the lab environment. Before the interaction, the second participant was told to always walk in a straight line towards a physical marker on the ground matching the speed of the robot (the first participant showed the same behaviour). No compensation was paid to either of the participants. After a short training phase of 1-2 trials to be able to match the speed of the robot, both participants interacted with the robot in all conditions in a within participant study design to achieve comparable results by minimising the effect of differences in behaviours between participants.

The robot's behaviours encoded in the hand-crafted rules were to avoid people to the right in pass-by encounters (see Figures 5.6 and 5.8a) and to stop and wait in path

crossing situations (see Figures 5.7 and 5.8b). In the subsequent evaluation, the robot showed 4 different behaviours based on the chosen avoidance cost model for comparison: *i)* the vanilla DWA planner, *ii)* a Gaussian Cost model on the local map (G-Local), *iii)* a Gaussian Cost model on the global map (G-Global), and *iv)* the Velocity Costmap approach (Vel-Maps). The weights $\omega_i \in \boldsymbol{\omega}$ of the used critique functions $c_i \in \boldsymbol{\mathcal{C}}$ were: *Velocity Costmaps:* 30, *Goal Align:* 10, *Path Align:* 10, *Goal Distance:* 24, *Path Distance:* 10, *Obstacles:* 0.01 (only lethal obstacles) and 30 when using G-Local which proofed to work the best using trial and error.[30] The QTC_{BC} distance threshold was set to $d_s = 4.0m$.[31] This parameter set gives the highest value to the Velocity Costmaps and a rather small value to the path distance and path align critique functions. This is necessary to allow the robot to diverge from the global path which represents a more or less straight line in the given set-up by weighting the Velocity Costmap costs higher than the path following costs. All the parameters were the same in simulation and on the real robot.[32] The main difference between the experiments was the full observability of the human in simulation compared to the perception pipeline described in Chapter 4 which only tracks the human in an area of up to $7m$ and $224°$ in front of the robot.

In both experiments the robot was reset to its original starting position and traversed the edge in the same direction towards the same goal using one of four planner variants. The simulated and real human also always started from the same position and moved towards the goal. The simulated human received a constant velocity command of $\rho = 0.55^{m}/_{s}, \theta = 0.0^{rad}/_{s}$ which corresponds to the robots maximum linear velocity. For the pass-by scenario both robot and human moved on a straight line towards each other, whereas during the path crossing the human's position was offset by $90°$ to create perpendicular trajectories (see Figures 5.6, 5.7, and 5.10). In both cases, if the robot did not initiate an avoidance behaviour, robot and human would collide half way through traversing the edge. The same conditions were recreated in the real-world experiment. Participants walked on a straight line towards a marker on the other side of the room.

[30] Previously unmentioned because it is not part of the original concept of the DWA planner, the weight for non-lethal costs can also be set. The higher the costs the less likely it is to move close to obstacles.

[31] This was the distance at which the robot would start its avoidance manoeuvre.

[32] For reproducibility, the remaining important DWA parameters used were: *vx_samples*: 3, *vth_samples*: 20, *max_trans_vel*: 0.55, *max_vel_x*: 0.55, *max_rot_vel*: 1.0, *acc_lim_x*: 1.0, *acc_lim_theta*: 3.2, *sim_time*: 0.8, *sim_granularity*: 0.025, *angular_sim_granularity*: 0.1, *forward_point_distance*: 0.325, *scaling_speed*: 0.25, *max_scaling_factor*: 0.2.

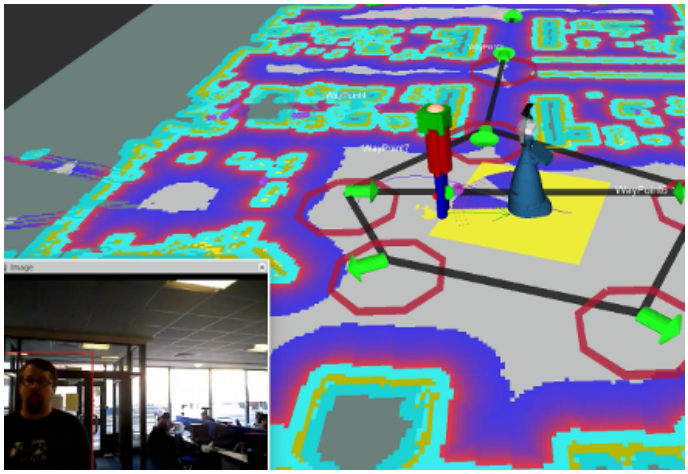


Figure 5.9: *Using Velocity Costmaps based on QTC descriptors in the L-CAS office environment with* Linda *the mobile robot. The bottom left shows the RGB image of the head mounted camera with the upper body detector output. The rest of the image shows the metric map, the global costmap, the Velocity Costmap (yellow square), the topological map (green arrows surrounded by red "influence areas" and black edges), the robot model, and the people perception output.*

The starting positions for pass-by were slightly offset to the right to account for a later detection of the human, but would still lead to a collision if the robot would not initiate avoidance. The participants were instructed to walk with a constant speed towards their goal, matching the velocity of the robot. If they collided with the robot (physical collisions are mitigated by the emergency bumpers around *Linda*), or had to stop in close proximity ($\sim 20 - 30cm$) to the robot or step aside to avoid one, it was reported as a collision by the participant. The interaction was started by the participant via a button on a remote control.

For each of the four conditions, 50 trials were recorded in simulation leading to a total of 200 interactions each for pass-by and path crossing. In the proof of concept experiment using the real robot, two participants generated 64 pass-by and 61 path crossing situations in total for all 4 conditions combined (with a minimum of 15 each). The safety of the trajectory was evaluated using the number of collisions, the perceived safety by analysing the minimum distance kept to the human, and the efficiency of the executed trajectory in terms of distance travelled, mean speed, and the duration.

Table 5.2: *Percentage of trajectories colliding with the human*

	Pass-by		Path crossing	
	Simulation	Robot	Simulation	Robot
DWA	100%	53.3%	100%	86.7%
G-Global	0%	22.2%	100%	75.0%
G-Local	100%	33.3%	0%	100%
Vel-Maps	**0%**	**12.5%**	**0%**	**13.3%**

Table 5.3: *Mean values for simulated scenarios: Min Distance(MD), Mean Spead(MS), Travel Time(TT), Distance Travelled(DT). Results with (****) achieved* $p < 0.0001$ *comparing the two distributions which produced the given mean using a two-tailed unpaired t-test.*

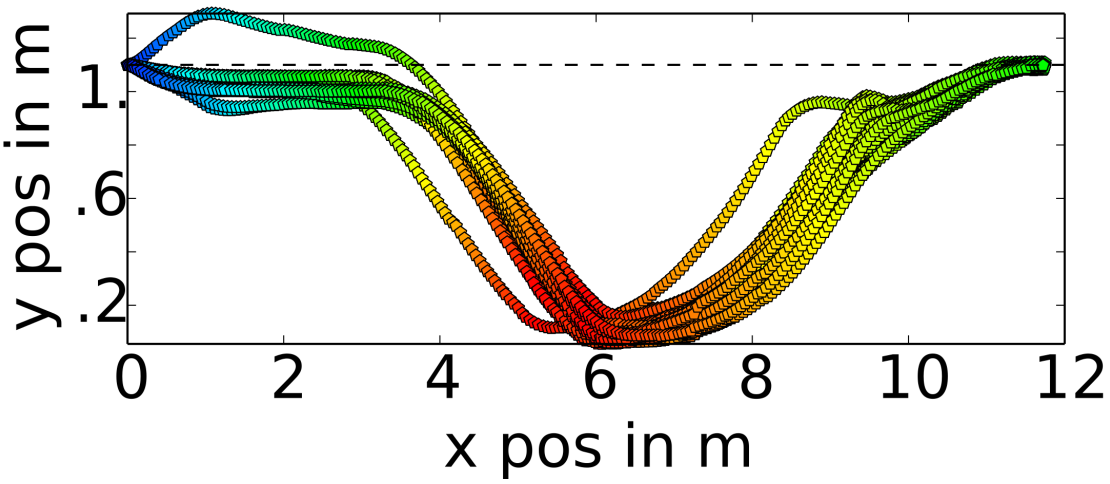
	Pass-by			Crossing		
	Vel-Maps		G-Global	Vel-Maps		G-Local
MD (m)	1.06	****	0.92	2.98	****	1.45
MS ($\frac{m}{s}$)	0.52	****	0.53	0.43	****	0.46
TT (s)	23.09	****	22.51	34.11	****	32.03
DT (m)	12.05	****	11.93	14.78	****	14.81

5.3.1 Results

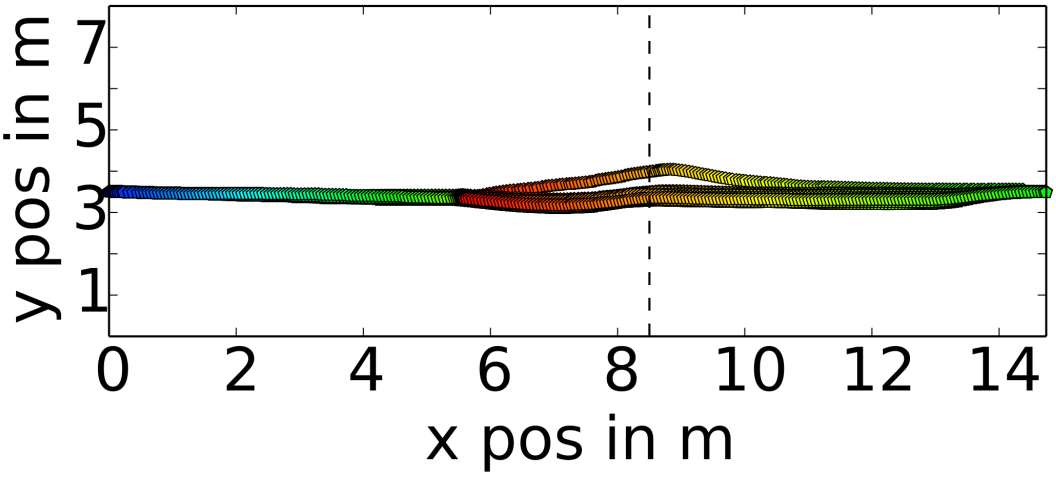
As can be seen from Table 5.2, a high percentage of the generated trajectories led to the robot colliding with the human, where collisions is to be taken in the literal sense or when the human had to explicitly prevent them. Thus, only G-Global and the Vel-Maps were compared for pass-by and G-Local and the Vel-Maps for path crossing in simulation. All the results of both experiments were generated using an unpaired t-test where *(****)* in Table 5.3 indicates that the difference between the two mean values is highly significant with $p < 0.0001$.

Simulation Results

In the pass-by scenario the main difference in results can be seen in the mean minimum distance between robot and human, denoted *Min Distance (MD)* in Table 5.3. The absolute difference between the two means is $14cm$ which also results in a higher travel time, and distance using the Vel-Maps. The absolute difference for the latter, however, is negligible. In the path crossing scenario, the difference in the *Min Distance (MD)* amounts to $1.53m$, the difference in distance travelled is only $3cm$ which implies that both cost functions created straight trajectories (see Figure 5.10b) like it was encoded in the rule.



(a) Pass-by shows an avoidance movement half way through the interaction when encountering the human. The smooth transition between the colours visualises that the robot travelled with close to constant velocity creating a uniform distribution of samples across time.



(b) Path crossing shows an abrupt transition from green to red, visualising where the robot stopped and waited for the human to pass.

Figure 5.10: *The generated trajectories using Velocity Costmaps in simulation; the black dashed line represents the human trajectory. The robot travelled from left to right and its trajectory is colour coded from blue via red to green to visualise time passed. Hence, abrupt colour changes visualise the robot being stationary for a certain time window.*

Real-World Results

To summarise the proof of concept experiment using *Linda*, only the most compelling results are listed in the following. In the path crossing scenario, all approaches but the Vel-Maps resulted in a very high number of collisions. The mean minimum distance to the human for the proposed approach was $0.76m \pm 0.42m$ in path crossing which is considerably lower than in simulation with full observability of the human, but still the highest of all the 4 conditions as can be easily inferred from the number of collisions. For the pass-by scenario the Vel-Maps also achieved the lowest number of collisions. Comparing the two most successful conditions, based on their number of collisions, a mean minimum distances of $0.56m$ for the Vel-Maps and $0.53m$ for G-Global with a p value of $p = 0.46$ and therefore no statistical significance was achieved. For G-Local, which performed much better on the real robot than in simulation regarding the collisions, a mean minimum distance of $0.47m$ was measured which with a p value of $p = 0.062$ also comes short of statistical significance when compared to Vel-Maps. Neither mean speed, travelled distance, nor duration showed any significant differences between any of the four conditions.

5.4 Discussion

The experiments showed that the QTC-based Velocity Costmaps approach to human-aware navigation resulted in collision free trajectories in almost all of the cases and shows the behaviour that was encoded via the given rules, i.e. avoiding to the right (see Figures 5.8a and 5.10a) or stopping to let the human pass (see Figures 5.8b and 5.10b). The late detection of the human and thereby reduced observability was one of the major downfalls of the real-world experiment. All the conditions suffered equally from this but the Velocity Costmaps were still able to cope in most of the cases. Given perfect observability in the simulated trials, only the Velocity Costmaps showed the ability to prevent collisions in both scenarios. The G-Global cost model achieved comparable results in the pass-by scenario, relying on the costly global path planner (Dijkstra) to avoid the human, using the local DWA planner only to follow that path and to not collide with walls. The poor performance of the two local obstacle avoidance strategies, i.e. vanilla DWA and G-Local, stems form the DWA planner getting stuck in a local cost maxima and stopping the robot to prevent a collision, despite there not being any additional constraints on the standard

DWA. Hence, the bad performance of the DWA in the experiment can be explained by the relatively small planning horizon that in the pass-by encounter has it driving towards the human until it is trapped by the human continuing her approach and in the path crossing by its inability to predict the human motion, thus, driving straight till the human is directly in front of the robot. Given that the human assumed a constant velocity, this did not prevent collisions in simulation but would have using a real robot, at least in most of the cases as can be seen from the robot trials in Table 5.2. Getting stuck in a local cost maxima and stopping is also the reason why the G-Local cost models performed well in the path crossing scenario as it would have the robot stop to let the human pass. The G-Global cost model, however, resulted in the global planner to try and pass in front of the human, leading to collisions because of the relentless motion model of the human used in simulation.

Looking at the mean minimum distance between human and robot in simulation, one can see that there is not much difference between the Velocity Costmaps and G-Global in the pass-by scenario which can be attributed to the size of the corridor itself. The human-robot distance was measured from the centre point of each agent and the human walked in the middle of the $2.6m$ wide corridor which theoretically leaves $1.3m$ on either side. In reality this is not achievable without colliding with the wall. The fact that the Velocity Costmaps approach kept a greater distance is due to the relatively high weight for the human-awareness. In the path crossing scenario, however, the Velocity Costmaps approach deliberately restricted the sample space of the DWA to only allow 0 velocities (i.e. stopping the robot to let the human pass) by assigning lethal costs throughout preventing any admissable forward trajectories (see Figure 5.7b) at a much greater distance than all the reactive planners which is an indication for the power and descriptiveness of encoding these kind of interactions in QTC_{BC}. The robot trials showed that the simulation results are a good indicator for the behaviour shown in real life as all algorithms showed performance comparable to simulation, suffering from the limitations of the human tracker. The lower collision rates could be attributed to the human walking slower than the simulated one and being influenced by their sense of self-preservation.

Limitations The biggest limitation of this approach mentioned by other researchers is the use of "outdated" approaches like the DWA planner by Fox et al. which is from (1997)

and Gaussian cost models as implemented by Lu et al. (2014). Both these approaches, however, are the default local planner and human-aware navigation of ROS which is the most commonly used middleware for research robots and also found its way into industrial applications with ROS Industrial.[33] Hence, their argument is not void but can be countered by the fact that a wide variety of institutions are still using this software and just because it is based on a publication from (1997) does not mean that it is outdated and not state-of-the-art.

The other point of criticism might be the low number of participants for the real-world study. The experiment at hand, however, only aimed at showing that the Velocity Costmaps are able to produce trajectories that create the desired QTC states. Hence, the trials using *Linda* were only intended to show that the results are comparable to a deployment on a real robot. In the next chapter, the system as a whole, including the Velocity Costmaps, is evaluated in a more comprehensive user study.

5.5 Summary

This chapter presented an overview of the navigation approaches currently used in the ROS systems, on *Linda*, and throughout the STRANDS project. Navigation is based on a metric map created via SLAM, a global costmap on which the global path planner computes an optimal path which could also include HRSI principles as can be seen from Chapter 2, and a local costmap used by the DWA local planner to avoid dynamic obstacles or obstacles that are not in the metric map. This local planner uses several critique functions to score trajectories in a sampling based approach and selects the one with the lowest costs. To generate only trajectories that create the desired QTC state, *Velocity Costmaps* are added to the set of critique functions which assign lethal costs to all the sampled trajectories that would not produce the desired QTC state. Thus, this mechanism is used to achieve human-aware navigation based on the QTC_{BC} model described in Chapter 3 by restricting the sample space of the DWA local planner.

Two experiments, in simulation and the real-world were conducted to show, that by encoding high-level knowledge of the unfolding of a possible interaction, the system is able to cope with a wider variety of possible situations and to make a more informed choice

[33] `http://rosindustrial.org/`

based on the intent of the human. They also show that this comes at ***no extra cost, comparing speed, and travel distance [Obj. 2.3]*** with a standard Gaussian cost approach. This indicates that the presented ***Velocity Costmaps are able to handle the trade-off between safety of the human interaction partner [Obj. 2.2] and finding a fast and energy efficient path [Obj. 2.3]***. More importantly, these Velocity Costmaps in combination with the DWA planner are able to transform high-level QTC states into low-level movement commands which will either recreate the desired QTC state or have the robot stop and wait until a QTC state is found that can be used to achieve movement towards the robot's goal or the human is gone.

Regarding the objectives concerning ***task efficient and comfortable behaviour generation [Obj. 2]***, this chapter addresses the creation of ***safe movement [Obj. 2.2]*** which prevents collisions in the vast majority of cases compared to a standard state-of-the-art approach while at the same time keeps the travel time and distance at a minimum, thereby, creating ***task efficient movement [Obj. 2.3]***. Since it builds on the QTC based model it inherits all its properties of ***abstracting from the actual environment [Obj. 1.1]*** by using the DWA local planner to handle the obstacle avoidance and generate goal directed movement if possible. Since the DWA takes the robot's acceleration limits into account, the generated movement commands are ***tailored to the hardware used [Obj. 3.3]*** and action selection is ***purely based on the robot's sensors and on-board processing [Obj. 3.2]***. Due to its reactive nature, ***action selection is fast and flexible [Obj. 2.4]*** and ***works in real-time [Obj. 3.1]***.

The main limitation of this approach is that the rules to find the best robot action given the observation of the human are hand coded and that the interaction type has to be specified in advance. This, however, will be solved in the next chapter which also shows how to incorporate human judgement in the action selection via LfD, thereby, influencing legibility and sociability in addition to the perceived safety and comfort. Another limitation is that this approach currently only influences the trajectory but not the speed or acceleration of the robot which are important metrics when it comes to human-aware navigation. Nevertheless, the speed could be set using ϱ and the angle using λ as mentioned earlier and the relative speed and angle compared to the human could be encoded by using the full, non-simplified, version of QTC. This, however, will remain future work.

—Extinction is the rule. Survival is the exception.

Carl Sagan, Astronomer/Astrophysicist

6

HRSI State Prediction and Action Selection

Addressing the remaining objectives of creating legible and sociable behaviour that is perceived as safe and therefore comfortable while at the same time creating goal directed and task efficient movements using a fast and flexible action selection process that is able to adapt to changes in the behaviour of the human, a dynamic belief generation that predicts the current best state and classifies the interaction type based on incremental updates of the QTC state chains has to be created. Therefore, the system introduced in this chapter has to overcome the problems stated in Chapter 5, namely learning the interaction models from observation, classifying the current interaction type online based on these models instead of assuming it to be known a-priori, and learning the previously hand-crafted rules for action selection from demonstration to include human judgement on the legibility and sociability of the action. To this end, the currently used QTC model has do be adapted in order to facilitate decision making by excluding the robot state from the classification of the interaction type.

As mentioned in Chapter 3, a Hidden Markov Model (HMM) is well suited to overcome the issue of discrete sensor measurements and the resulting missing states and yields very good classification results. However, a HMM needs the entire state sequence for classification which would result in the robot only being able to determine the type of interaction after the interaction is complete. To build a system that is able to work online using incremental updates of the QTC state chain, the activity model has to be adapted and a new classification method is necessary. The training on the other hand will still follow the same approach as for the HMM, but will be adapted by removing the hidden layer to generate a Markov Model (MM). To overcome the limitation of removing the hidden layer and dealing with unobserved state transitions, this MM is used as the basis of a particle filter for Qualitative Spatial Relations (QSR). This enables the classifier to deal with missing observations in between states and is able to recover from false classifications or switch behaviour in case of behaviour change by the human interaction partner.

As mentioned above, in order to classify the current interaction, the robot movement has to be disregarded because it is not known a-priori, but is to be determined after classification. Hence, the QTC states have to be split into robot symbols and human symbols which makes the representation too ambiguous for classification. To overcome this problem, the online belief generation uses a combination of QTC_C describing the interaction between the human and the robot's goal and the QTC_{BC} state chains for human and robot to generate a new QTC like state that is meaningful enough to unambiguously classify the action of the human without considering the robot's action in the classification task. The HMM based approach described in Chapter 3 and used in Chapter 4, on the other hand, used the full QTC_{BC} state of human and robot, which uses the actions of human and robot equally to determine the interaction type. Hence, this chapter combines the QTC state generation described in Section 4.2 with said particle filter using QTC like states to generate Velocity Costmaps introduced in Chapter 5. Thus, a fully autonomous system which can be seen in Figure 6.2 for integration into the robot *Linda* is presented.

In addition to the online classification, the action selection rules from Chapter 5 can also be learned from demonstration by a lay user, as mentioned above, which is shown in this chapter. Learning from Demonstration (LfD) is a popular principle in robotics, creating policies from example state to action mappings as defined by Argall et al. (2009). In HRSI there are only very few examples that make use of this approach as mentioned

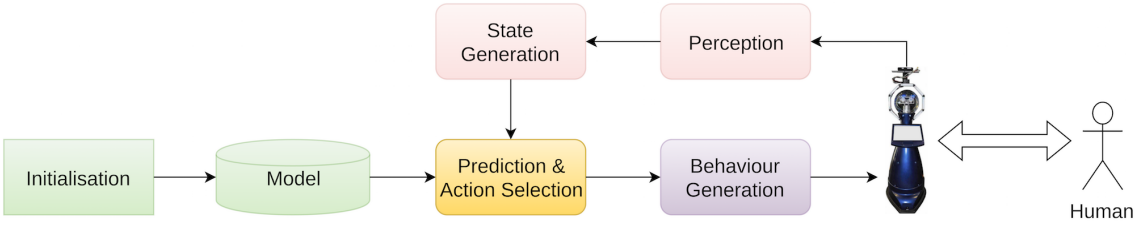


Figure 6.1: *This chapter's contribution to the system shown in Figure 1.1 building on the partial system shown in Figure 5.2. In comparison to Figure 5.2, the manually crafted action selection is now replaced by a fully autonomous state prediction and action selection process that is based on models learned from interaction and not on hand crafted rules of conceptual models. The fully autonomous system of perception, state prediction using learned models, action selection using learned state to action mappings, and behaviour generation is presented. The novel contribution is highlighted in yellow.*

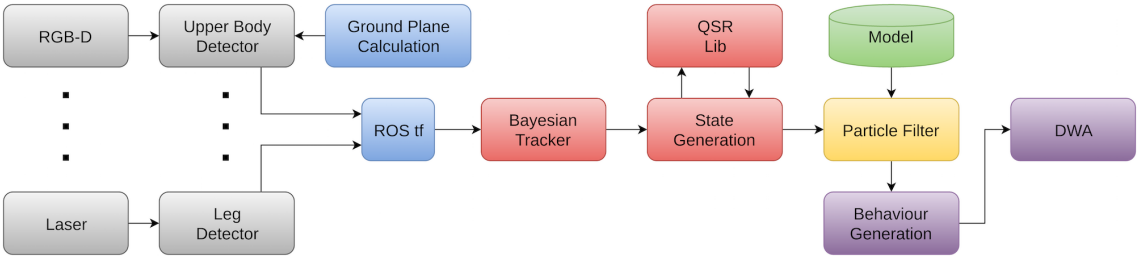


Figure 6.2: *The complete perception, activity recognition, and behaviour generation pipeline (extending Figure 4.3). The colours correspond to the modules in Figure 1.1 and 6.1. From left to right: grey: sensors and detectors used on* Linda *at the time of writing this thesis, blue: helper functions, red: tracking and QSR generation, yellow: state prediction and action selection, green: activity model, purple: Velocity Costmap and low-level movement command generation using the DWA local planner. The source code to all these components is freely available, please refer to Appendix B.*

in Chapter 2 where Lichtenthäler et al.'s (2013) work, where a naïve participant teleoperated the robot to record the preferred trajectories in path-crossing situations, is closely related to the experiment design in this chapter. The LfD approach which makes use of the knowledge and experience of the human demonstrator to generate safe, legible and comfortable trajectories is employed in this chapter to generate the action selection policy, while at the same time learning interaction models for classification from observation of the human interaction partner.

Chapter Contributions To summarise, the main contribution of this chapter is the particle filter for Qualitative Spatial Relations (QSR) which builds on a prediction and observation model that is generic enough to allow the use with any kind of QSR and is explained here based on QTC. These newly introduced QTC models use a conglomerate of

different QTC states using different variants of the calculus to create a representation that is meaningful enough to ***unambiguously distinguish different types of interactions [Obj. 1]*** without relying on the robot's state for classification, therefore, ***facilitating decision processes [Obj. 1.6]*** by modelling the robot state in conjunction with the human state while at the same time being able to separate the two into observation and action. These models are learned from demonstration during the training phase where the robot is remote controlled by either a participant or an experimenter while another experimenter or participant interacts with it. The best action of the robot is selected from a conditional probability table that describes the joint probability for each possible action given the current belief. Compared to Chapter 5, where these were hand-crafted rules, the conditional probability tables can be learned from demonstration using an "Inverse Oz of Wizard" experiment set-up (as described in Section 2.4.1) similar to the work presented by Lichtenthäler et al. (2013) to incorporate human judgement into the behaviour model which is shown in the experiment section of this chapter. A conceptual overview of this chapters contribution towards the final system can be seen in Figure 6.1.

This final method chapter addresses the objectives of creating ***a fast and flexible action selection process [Obj. 2.4]*** by using a particle filter that is able to change the current classification of the interaction type based on the humans behaviour in ***real-time [Obj. 3.1]*** and therefore always represents the belief that fits the current observed state of the world best. This builds on the perception pipeline introduced in Chapter 4 and is, therefore, able to ***work in real-time [Obj. 3.1] only relying on the robot's sensors and on-board processing [Obj. 3.2]***. The classification itself is based on a conglomerate of QTC states that were introduced in Chapter 3 and inherits all the properties of this qualitative representation like the ability to ***abstract from the metric world [Obj. 1.1]*** and only ***represent the qualitative character of motion [Obj. 1.2]*** while also ***representing distance as a relevant attribute of HRSI [Obj. 1.3]*** which allows it to ***generalise over a vast number of situations and environments [Obj. 1.4]*** and in itself is a ***well-founded, concise and tractable model [Obj. 1.5]***. Especially the tractability of this model is shown in the following using it to classify interactions online. Finally, the action selection is implemented as simple look-up like in Chapter 5 but uses a model learned from demonstration which creates ***legible and sociable [Obj. 2.1], and***

safe [Obj. 2.2] movement by inheriting the ***safety [Obj. 2.2] and task efficiency [Obj. 2.3]*** of the Velocity Costmaps in combination with the DWA local planner.

The whole system is evaluated in a two-part user study where the first part represents the learning phase in which the robot is remote controlled by a participant while interacting with the experimenter, recording the generated QTC states for the activity model and the conditional probability table for action selection. The second part evaluates the learned models and action selection policies using a separate set of participants and a fully autonomous robot that has no prior knowledge about the interaction type.

6.1 Particle Filter based Activity Recognition using QSRs

Particle filters are a popular concept in modern robotics and are used for all kinds of estimation problems. The people perception in Chapter 4, for example, could have used a particle filter instead of a Kalman filter. The self localisation of the majority of mobile robots including *Linda* uses the Adaptive Monte-Carlo Localisation (AMCL) approach as described by Thrun et al. (2005). This uses a particle filter to represent the beliefs about the position of the robot. In general, particle filters are used to approximate a continuous function, like the movement of people or the robot, using discrete update and prediction steps based on a prediction and an observation model. The prediction model for AMCL, for example, models the odometric error of the robot and the observation model scores the predicted position and orientation based on the laser measurements taken. The predicted particles are weighted based on the observation and are then re-sampled in a Monte-Carlo based sampling process. Due to the pseudo random nature of this approach it is able to deal with missing observations and also able to recover from false classifications given that a certain amount of particles are kept alive despite their low weight. In general, the more particles are used the more precise the estimation of the underlying continuous function becomes but the more computationally expensive this approach gets. Hence, particle filters represent a trade-off between precision and computational cost.

Even though a QSR state chain is not a continuous process but inherently discrete, due to particle filters themselves following a discrete prediction and update procedure they are well suited to represent these incremental updates in the state chain, allowing for online

classification of the interaction type. The following presents a particle filter for QSRs able to deal with missing observations and to recover from wrong beliefs.

6.1.1 Prediction Model

The prediction model for a particle filter describes a conditional probability distribution $P(X_{t+1}|X_1, \ldots, X_t)$ that predicts the probability of state X_{t+1} given all previous states. To simplify this problem, many particle filters, including the one presented here, follow the Markov assumption that it is sufficient to only take the current state into consideration to predict the next state, thus turning the conditional probability into $P(X_{t+1}|X_t)$, increasing the tractability of the approach. For discrete states in any given QSR, that requires a transition probability of each possible state to each possible state including self-transitions, e.g. $A \rightarrow B$ with $A, B \in \mathbf{\Omega}$ where $\mathbf{\Omega}$ represents the entire set of possible states that can be observed by the system.

From above description, the use of a Markov chain seems to be the most appropriate method of representing these transition probabilities. Let $\mathbf{Y}$ be a set of random variables $\mathbf{Y} = (X_t)_{t \in \mathbb{N}}$ with $X_t \in \mathbf{\Omega}$ where $\mathbf{\Omega}$ describes the finite set of possible states of the observed interaction, then $\mathbf{Y}$ is a discrete Markov chain if $P(X_{t+1} = \Omega_{jt+1}|X_t = \Omega_{jt}, X_{t-1} = \Omega_{jt-1}, \ldots, X_1 = \Omega_{j1}) = P(X_{t+1} = \Omega_{jt+1}|X_t = \Omega_{jt})$. All these conditional probabilities can then be expressed as a matrix $\mathcal{M}$ of size $|\mathbf{\Omega}| \times |\mathbf{\Omega}|$ which is the so-called Markov Model (MM). This MM $\mathcal{M}_{i,j}$ is used as the prediction model for the particle filter by looking up the transition probabilities $s_i \rightarrow s_j$ with $s_i, s_j \in \mathbf{\Omega}$ describing the probability of s_i transitioning to s_j.

The model $\mathcal{M}$ is created using the same approach as for the HMM in Section 3.4. The two matrices T for the transitions and E for the emissions are both of the size $|\mathbf{\Omega}| \times |\mathbf{\Omega}|$ and the transition matrix is initialised based on the legal transitions. The emission matrix E, in contrast to the previous approach, however, is initialised to the identity $\mathcal{I}$ which only allows a state to emit itself as a symbol, e.g. $(+ - \emptyset\,\emptyset)$ can only emit $(+ - \emptyset\,\emptyset)$. Thereby, in the subsequent training process, the Baum-Welch algorithm is forced to only update the transitions based on the training data, effectively creating a Markov Model by omitting the emissions. Hence, as in Chapters 3 and 4, the state transitions are modelled as a first-order Markov chain, while now uncertainty in the observations is accounted for

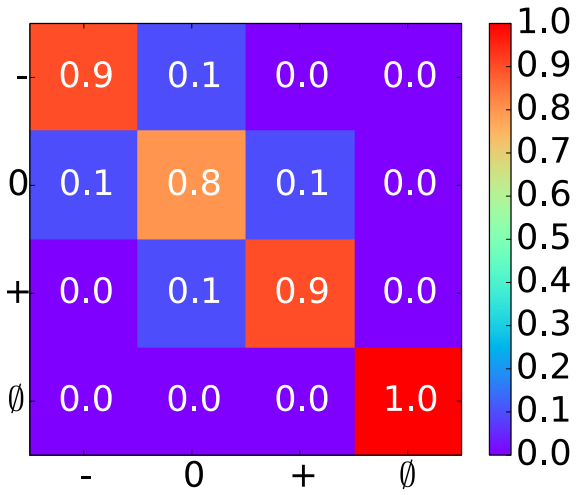


Figure 6.3: *The QTC confusion matrix. Allowing for a conceptual distance $\phi = 1$ between the symbols $-, 0, +$ with $p = 0.1$. Since $\phi \triangleq 3 \; \forall \; \emptyset \rightarrow \{-, 0, +\}$, the $\emptyset$ can only be observed as itself.*

during the update step of the particle filter using the observation model as described in Section 6.1.2 instead of the emission probabilities of the HMM. The resulting prediction model can then be used to obtain the transition probabilities from one state to all other states $P_{\mathcal{M}}(X_{t+1} = \Omega_{jt+1} | X_t = \Omega_{jt})$ via a fast and simple look-up which is used in a Monte-Carlo approach to generate a new generation of particles from the old ones. For every state $X_t = \Omega_{jt}$ the joint probability table for transitioning into any other state $X_{t+1} = \Omega_{jt+1}$ is used to pseudo randomly pick a new particle where the probability $P_{\mathcal{M}}(X_{t+1} | X_t)$ serves as the weight for this sampling process. Hence, the prediction is based on the learned probabilities of the type of interaction that $\mathcal{M}$ models.

6.1.2 Observation Model

In a particle filter, the observation model is used to score the prediction based on the current observation of the state of the world. For AMCL this is easily done by a Gaussian distribution around the measured position of the robot, assigning the particles a weight according to this Gaussian. When looking at QSRs, this task of scoring the state based on the similarity to another state becomes conceptually more complicated. Technically, this observation model is also just a $|\mathbf{\Omega}| \times |\mathbf{\Omega}|$ matrix containing a conditional probability distribution $P(Y_t = \Omega_{t,i} | X_t = \Omega_{t,j})$ describing the probability of observing $\Omega_{t,i}$ given the prediction $\Omega_{t,j}$.

To obtain this observation model from training data, the emission matrix of the HMM described in Section 3.4 could be used, but in order to train the transitions and emissions at the same time and get a meaningful result for both, large amounts of training data are required. Since none of the studies conducted came close to the required numbers, a more deterministic and scalable approach has been developed. Figure 6.3 shows a confusion matrix for QTC based on the conceptual distance of a single symbol ϕ introduced in Section 3.3. This confusion matrix assigns the highest weight to $\phi = 0$ and a weight of 0.1 to $\phi = 1$. For all other values $\phi \geq 2$ a zero probability of observation is assumed. These numbers represent trial and error results rewarding the correct particles and killing the remainder to achieve quick convergence. By not allowing for illegal transitions $\phi = 2$, this model becomes rather restrictive given the discrete nature of the observations, but this is countered by creating only valid QTC state chains as described in Section 4.2 and other mechanisms inside the filter itself described later on. This confusion matrix is then used to generate the probability

$$P(Y_t = \Omega_{t,i} | X_t = \Omega_{t,j}) = \prod_{q^k_{t,i} \in \Omega_{t,i}, q^k_{t,j} \in \Omega_{t,j}} P(q^k_{t,i} | q^k_{t,j}) \tag{6.1}$$

with $\Omega_{t,i}, \Omega_{t,j} \in \mathbf{\Omega}$ where $P(q^k_{t,i}|q^k_{t,j})$ can be inferred from Figure 6.3. These products are then arranged in an $|\mathbf{\Omega}| \times |\mathbf{\Omega}|$ matrix $\mathcal{O}$ which is subsequently normalised so each row sums to 1. In a simple look-up, each predicted particle can now be assigned a weight based on its conceptual distance to the observed state which influences the likelihood of it surviving the Monte-Carlo based re-sampling step. Hence, the closer the particle to the observation $\Phi(\Omega_{t,i}, \Omega_{t,j}) \rightarrow 0$, the more likely it will survive and multiply since particles are drawn with replacement.

6.1.3 Particle Filter

The key idea of every Sampling Importance Resampling (SIR) particle filter is to approximate the posterior probability density function $P(X_t|Y_{1:t})$ by a set of random variables and associated weights to compute estimates of the state of the world (Arulampalam et al. 2002). To simplify the algorithm, as mentioned above, the system is assumed to be Markovian which means that the current state of the system only depends on the pre-

vious state and the current observation. Given this assumption, the previously defined probability density functions $P(X_t|X_{t-1})$ representing the prediction model and $P(Y_t|X_t)$ representing the observation model with $X \in \{(\Omega, \mathcal{M}) : \Omega \in \mathbf{\Omega}, \mathcal{M} \in \boldsymbol{\mathcal{M}}\}$ and $Y \in \mathbf{\Omega}$ are used where $\mathbf{\Omega}$ is the set of all possible states and $\boldsymbol{\mathcal{M}}$ is the set of all trained interaction models, e.g. for pass-by and path crossing. This results in

$$P\left(X_t|Y_{1:t}\right) = \sum_{i=1}^{N_p} \omega_t^i \delta\left(X_t - X_t^i\right) \tag{6.2}$$

where X_t^i represents the value of the i-th particle, N_p being the number of particles, and $\delta(\cdot)$ representing the Dirac delta measure (Arulampalam et al. 2002). The weights ω^i are normalised to sum to 1 with $\omega^i \geq 0$. For the observation $\mathcal{M} \in X$ is ignored.

It should be noted, however, that a particle $X \in \{(\Omega, \mathcal{M}) : \Omega \in \mathbf{\Omega}, \mathcal{M} \in \boldsymbol{\mathcal{M}}\}$ consists of the current state it represents and the model $\mathcal{M} \in \boldsymbol{\mathcal{M}}$ it was generated by. Thus, while the observation model is the same for every particle, each particle has a separate prediction model. This allows for the classification of the current interaction type.

Algorithm 6.1 Particle Filter

Require: $\mathbf{\Omega}, \boldsymbol{\mathcal{M}}, \mathcal{O}, N_p$
Ensure: $N_p \geq |\boldsymbol{\mathcal{M}}| \cdot |\mathbf{\Omega}|$
  $\mathbf{X} = \{(\Omega, \mathcal{M}) : \Omega \in \mathbf{\Omega}, \mathcal{M} \in \boldsymbol{\mathcal{M}}\}$
  $\mathbf{x}_0 = U(\mathbf{X})$ with $|\mathbf{x}_0| = N_p$
  while *true* **do**
    equalise weights $\omega_t^i \in \boldsymbol{\omega}_t$
    for all $x_t^i \in \mathbf{x}_t$ **do**
      predict new particle: $x_p^i \sim P_m(x_{t+1}^i|x_t^i)$
      update weight: $\omega_{t+1}^i = P_{\mathcal{O}}(y_{t+1}|x_{t+1}^i)$ with $y_{t+1} \in \mathbf{\Omega}$
    end for
    sample N_p times from $\boldsymbol{\omega}_{t+1} \rightarrow \mathbf{j}$
    $\mathbf{x}_{t+1} = \mathbf{x}_p^{\mathbf{j}}$
  end while

Algorithm 6.1 shows the functionality of the particle filter which requires the set of all possible states $\mathbf{\Omega}$, the set of models $\boldsymbol{\mathcal{M}}$, the observation model $\mathcal{O}$, and the number of particles N_p which has to be greater than the number of states times the number of models. The set of all particles $\mathbf{X}$ describes all possible states in all models and the uniform distribution U makes sure that at least one instance of each particle is created initially. Normally, a uniform distribution does not guarantee that every state in every

model is assigned a particle which is why this has been programmatically ensured. Hence, the particles are evenly spread amongst all states in all models until every state has at least one particle after which the remaining particles are drawn randomly. Thus, the list of all of particles $\mathbf{x}_0$ contains N_p particles where each particle $X \in \mathbf{X}$ can have multiple instances. The subsequent loop represents the prediction, update, and re-sampling step:

1. Equalise the set of weights $\boldsymbol{\omega}$.
2. Generate particles based on the previous state of the system $P_m(x^i_{t+1}|x^i_t)$. Given $x^i_t = (\Omega_t,)$, Ω_{t+1} is sampled from the set of possible successors of Ω_t using the transition probabilities in model $\mathcal{M}$ to create $x_{t+1} = (\Omega_{t+1}, \mathcal{M})$.
3. Reweigh particles based on current observation $y_t \in \boldsymbol{\Omega}$ and the observation model $\mathcal{O}$: $\omega^i_{t+1} = P_{\mathcal{O}}(y_{t+1}|x^i_{t+1})$.
4. Sample the list of indices $\mathbf{j}$ based on the weights $\boldsymbol{\omega}_{t+1}$ such that $|\mathbf{j}| = N_p$.
5. Create new particle generation $\mathbf{x}_{t+1}$ using the list of predicted particles $\mathbf{x}_p$ and the list of indices $\mathbf{j}$: $\mathbf{x}_{t+1} = \mathbf{x}^{\mathbf{j}}_p$.

The result of this filtering is the so-called belief $bel(\mathbf{x}_t)$ at any give time t. Hence, the current state of the world at time t is expressed by the sum of particles generated by the filter which allows for incremental updates of this belief which represents the most likely state of the human Ω_t and the model $\mathcal{M}$ it was generated by. Let $\boldsymbol{\mathcal{M}}$ be a set of prediction models with $\mathcal{M}_m \in \boldsymbol{\mathcal{M}}$, then each model will be represented by a similar number of particles after the initialisation of the filter. Since the model will be used in the prediction step for each particle, the model that predicts the next state most accurately will generate the majority of surviving particles. Hence, classification of the interaction and belief generation for the current state is achieved by forming the mode of the individual components of the list of particles $\mathbf{x}_t$ currently held by the filter. As a reminder, the list of particles at time t is defined as $\mathbf{x}_t = [\{(\Omega_t, \mathcal{M}_m) : \Omega_t \in \boldsymbol{\Omega}, \mathcal{M}_m \in \boldsymbol{\mathcal{M}}\}]$ where $\Omega_t \in \boldsymbol{\Omega}$ describes the state this particle represents and $\mathcal{M}_m \in \boldsymbol{\mathcal{M}}$ the model it was generated by.[34] Note, $\mathbf{x}_t$ is represented as a list of particles and not a set which means that duplicate entries are possible which is one of the basic requirements of a particle filter. Let $\boldsymbol{\Omega}_t$ be the list

[34]The model $\mathcal{M}_m \in \boldsymbol{\mathcal{M}}$ is not dependent on the time t as it does not change over time.

of states extracted from the list of particles $\mathbf{x}_t$ and $\boldsymbol{\mathcal{M}}_m$ be the list of models generated similarly, then the current best state is calculated as $\Omega^\star = \text{mode}(\boldsymbol{\Omega}_t)$ and the best model as $\mathcal{M}^\star = \text{mode}(\boldsymbol{\mathcal{M}}_m)$ where mode is a function to find the most common element of a list equivalent with the bin with the most entries of a histogram. In case of a multimodal distribution of particles or models a winner is chosen randomly among the set of most common states and models. No constraints are used for the generation of the lists $\boldsymbol{\Omega}_t$ and $\boldsymbol{\mathcal{M}}_m$ which means that each predicted state and each predicted model contribute equally to the vote regardless of the weight of the particle. Given this method, the belief $bel(\mathbf{x}_t)$ converges to one of the models $\mathcal{M}_m$ in a few sampling generations which can lead to over-fitting resulting in the inability to recover from false classifications or the inability to change the robot's behaviour if the human's behaviour changes because once all particles representing a certain model die, some states might never be predicted and there is no way of reviving a dead model given the current definition of the filter.

Algorithm 6.2 Particle Filter with starvation prevention

Require: $\boldsymbol{\Omega}, \boldsymbol{\mathcal{M}}, \mathcal{O}, N_p, \varsigma$
Ensure: $N_p \geq |\boldsymbol{\mathcal{M}}| \cdot |\boldsymbol{\Omega}|$
  $\mathbf{X} = \{(\Omega, \mathcal{M}) : \Omega \in \boldsymbol{\Omega}, \mathcal{M} \in \boldsymbol{\mathcal{M}}\}$
  $\mathbf{x}_0 = U(\mathbf{X})$ with $|\mathbf{x}_0| = N_p$
  while *true* **do**
    equalise weights $\omega_t^i \in \boldsymbol{\omega}_t$
    for all $x_t^i \in \mathbf{x}_t$ **do**
      predict new particle: $x_p^i \sim P_m(x_{t+1}^i | x_t^i)$
      update weight: $\omega_{t+1}^i = P_{\mathcal{O}}(y_{t+1} | x_{t+1}^i)$ with $y_{t+1} \in \boldsymbol{\Omega}$
    end for
    sample $N_p - (N_p \cdot \varsigma)$ times from $\boldsymbol{\omega}_{t+1} \rightarrow \mathbf{j}$
    sample $N_p \cdot \varsigma$ times from $U(\mathbf{X}) \rightarrow \mathbf{x}^u$
    $\mathbf{x}_{t+1} = \mathbf{x}_p^{\mathbf{j}} \cup \mathbf{x}^u$
  end while

To overcome this issue of over-fitting and to be able to change the classification once the behaviour of the human changes, Algorithm 6.2 shows how to prevent models from dying. By defining a so-called starvation factor $0 \leq \varsigma \leq 1$, in the re-sampling step, only $N_p - (N_p \cdot \varsigma)$ particles are re-sampled based on the weights $\boldsymbol{\omega}_t$ and the remainder is drawn from the set of all possible particles $\mathbf{X}$ which is the distribution over all particles in all models using a uniform distribution.[35] This starvation factor was set to $\varsigma = 0.1$ during the

[35]This uniform distribution is different from the one used for initialisation as it does not ensure that each particle is represented at least once but is purely random.

experiments which has shown to provide enough uniformly distributed particles to "revive" a model if necessary and also enough particles sampled according to $\boldsymbol{\omega}$ to generate reliable classifications. Belief generation including state prediction and model classification is achieved the same way as described above and the resulting best state $x^\star$ and best model $\mathcal{M}^\star$ can then be used for action selection.

6.2 QTC Models for Decision Processes

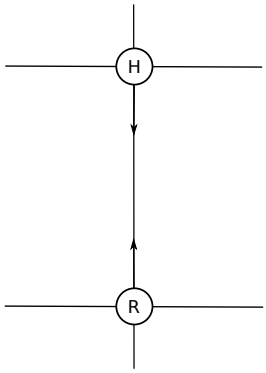
Looking at the models introduced in Chapter 3, the biggest problem is that the robot is part of the QTC state and, therefore, the classification of the interaction type is based on both the human and the robot. Figures 6.4 and 6.5 show two interactions, i.e. path crossing and pass-by, which could be produced by the rules introduced in Chapter 5. The resulting QTC_C state chains are

$$\begin{aligned}(-_h\ -_r\ 0_h\,0_r) &\rightsquigarrow (-_h\ -_r\ +_h\,+_r)\\ &\rightsquigarrow (0_h\,0_r\ +_h\ +_r)\\ &\rightsquigarrow (+_h\ +_r\ +_h\,+_r)\\ &\rightsquigarrow (+_h\ +_r\ 0_h\,0_r)\end{aligned}$$

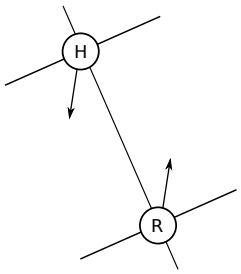
for the pass-by interaction and

$$\begin{aligned}(-_h\ -_r\ +_h\,-_r) &\rightsquigarrow (-_h\,0_r\ +_h\ 0_r)\\ &\rightsquigarrow (0_h\,0_r\ +_h\ 0_r)\\ &\rightsquigarrow (+_h\ -_r\ +_h\,+_r)\\ &\rightsquigarrow (+_h\,0_r\,0_h\,+_r)\\ &\rightsquigarrow (+_h\ +_r\ -_h\,+_r)\end{aligned}$$

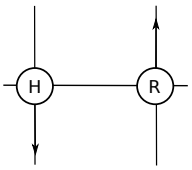
for the path crossing interaction. As in the figures, the symbols are indexed with h for the symbols describing the human movement and r for the robot, respectively. As a quick reminder, the first two symbols represent the distance constraint, i.e. approach $-$, repel



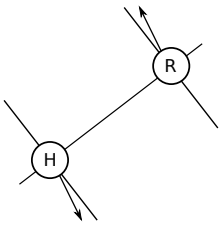
(a) $(-_h\ -_r\ 0_h\ 0_r)$*: Both agents approach each other on a straight line.*



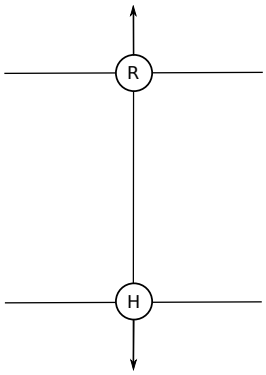
(b) $(-_h\ -_r\ +_h\ +_r)$*: Human and robot both move to the right while approaching.*



(c) $(0_h\ 0_r\ +_h\ +_r)$*: Human and robot are "shoulder to shoulder" at the closest point during the interaction and only move along the perpendicular lines.*

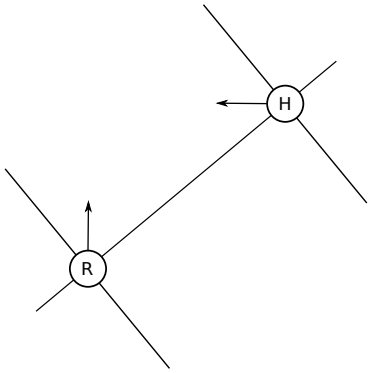


(d) $(+_h\ +_r\ +_h\ +_r)$*: Human and robot passed each other and are correcting their trajectories towards their goals.*

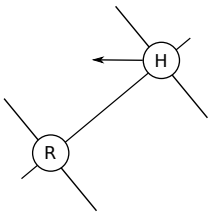


(e) $(+_h\ +_r\ 0_h\ 0_r)$*: Human and robot move away from each other on a straight line after the pass-by interaction is over.*

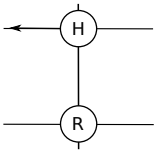
Figure 6.4: *A pass-by scenario as it would be created by the rules shown in Figure 5.8a encoded in QTC_C. The respective QTC_C states are shown in the captions of the sub-figures and are indexed with h for the symbols describing the human movement and with r for the robot. The actual movement of the two agents form t_{n-1} to t_n is indicated by the arrows. From Figure 6.4a to 6.4b the QTC_{BC} distance threshold d_s is crossed and the avoidance behaviour of the robot is triggered.*



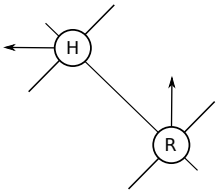
(a) $(-_h\ -_r\ +_h\ -_r)$*: Both agents approach each other.*



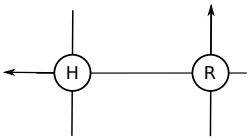
(b) $(-_h\ 0_r\ +_h\ 0_r)$*: Robot stops to let the human pass, human keeps approaching.*



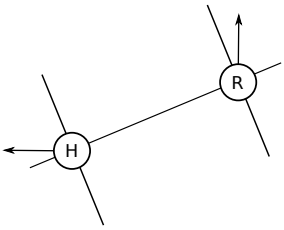
(c) $(0_h\ 0_r\ +_h\ 0_r)$*: Robot still stationary, human is directly in front of the robot at the closest point.*



(d) $(+_h\ -_r\ +_h\ +_r)$*: Human moves away from the robot and the robot starts moving towards its goal again approaching the human.*



(e) $(+_h\ 0_r\ 0_h\ +_r)$*: Human and robot are at the same hight, human moves away and the robot moves along the perpendicular line.*



(f) $(+_h\ +_r\ -_h\ +_r)$*: Both agents move away from each other.*

Figure 6.5: *A path crossing scenario as it would be created by the rules shown in Figure 5.8b encoded in QTC_C. The respective QTC_C states are shown in the captions of the sub-figures and are indexed with h for the symbols describing the human movement and with r for the robot. The actual movement of the two agents form t_{n-1} to t_n is indicated by the arrows. From Figure 6.5a to 6.5b the QTC_{BC} distance threshold d_s is crossed and the stopping and waiting behaviour of the robot is triggered.*

+, or stationary 0, and the last two represent the side constraint, i.e. left −, right +, or along the connecting line 0.

The two state sequences for pass-by and path crossing are significantly different from each other and can be reliably classified using a HMM. However, as mentioned above, these state chains include the robot behaviour as well as the human behaviour. To create any kind of decision process, the observed state of the human has to be separated from the action of the robot that triggers a state change. Thus, the two symbols describing the robot behaviour cannot be included in the classification of the interaction type because no a-priori knowledge about the interaction and hence the robot's actions to be chosen has to be assumed. Looking at the two state chains without the robot symbols:

$$(-_h\, 0_h) \rightsquigarrow (-_h\, +_h) \rightsquigarrow (0_h\, +_h) \rightsquigarrow (+_h\, +_h) \rightsquigarrow (+_h\, 0_h)$$
$$(-_h\, +_h) \rightsquigarrow (-_h\, +_h) \rightsquigarrow (0_h\, +_h) \rightsquigarrow (+_h\, +_h) \rightsquigarrow (+_h\, 0_h) \rightsquigarrow (+_h\, -_h)$$

it is easily visible that there is no difference apart from the side constraint in the first state and the extra last state for the path crossing which could also be noise in the observation. If these state chains would be expressed in QTC_{BC}, the distinguishing symbols would be removed completely depending on the actual value of d_s:

$$(-_h\, \emptyset_h) \rightsquigarrow (-_h\, +_h) \rightsquigarrow (0_h\, +_h) \rightsquigarrow (+_h\, +_h) \rightsquigarrow (+_h\, \emptyset_h)$$
$$(-_h\, \emptyset_h) \rightsquigarrow (-_h\, +_h) \rightsquigarrow (0_h\, +_h) \rightsquigarrow (+_h\, +_h) \rightsquigarrow (+_h\, \emptyset_h)$$

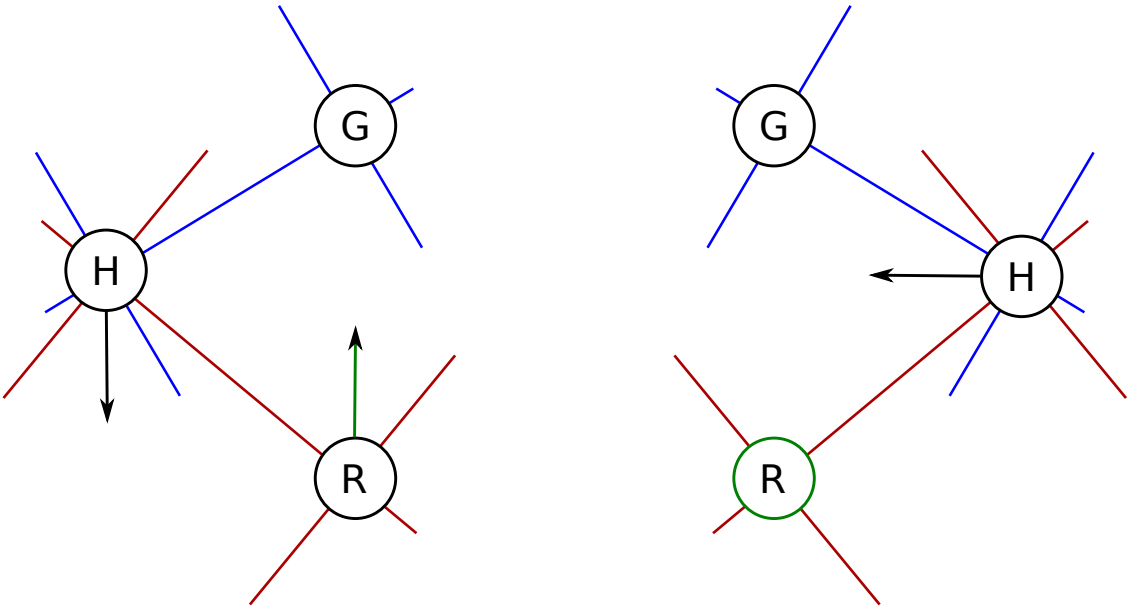
Apart from above issue, in Chapter 5 the interaction type was assumed to be known and the rules were hand crafted which implicitly encodes the very important assumption that the robot's intention is known. Thus, the intention of the human was used to determine which specific action the robot should choose but the robot's intention was ignored which means that for example it would always be forced to participate in a pass-by interaction even though it wants to move in the same direction as the human and not pass her. As a result, in order to be able to classify a certain encounter, the intention of the robot is necessary, e.g. it is important to know if the robot wants to move in the same or

the opposite direction as the human as both would result in two completely different interactions, i.e. pass-by or overtaking/following, while the intention of the human to move along the corridor in a certain direction would be unchanged. To overcome these two limitations, the robot's and the human's intention have to be expressed in a combined QTC state which can then be used for classification.

6.2.1 Combining QTC sequences

It is apparent from above problem description that the QTC symbols for the human from the QTC representation of human and robot movement alone are not sufficient to classify the interaction and that the robot's intention is not included any more. To overcome the issue of classification, QTC states could be created for the relation between the human and all kinds of features in the environment like flower pots, corners, or any other landmark in addition to the QTC state for human and robot. This however, severely limits the generalisability of the learned model because these landmarks would have to be at similar locations in all areas where this kind of interaction could take place or training data for every location where these interactions could happen would be required. Moreover, during a long-term deployment of several month, these landmarks could also move which would make classification impossible. Such a landmark, however, would solve this classification problem or at least simplify it significantly. If there are no physical landmarks that can be used, a virtual landmark has to be generated but the position of this landmark has to be chosen carefully and it should also correlate with the type of interaction and the desired robot behaviour.

The virtual landmark that was chosen for the presented approach is the robot's short term navigation goal, i.e. the next waypoint in its topological map on route to the final goal. By using this waypoint, the created representation also takes the short term intention of the robot into account, e.g. travelling in the same or opposite direction of the human as stated above. Additionally, for similar interactions the robot's goal will also be in similar locations, mostly on a straight line in front of the robot depending on the granularity of the topological map assuming that every significant change of direction in travel would only happen at waypoints and not in between as it has been implemented in both real-world scenarios of the STRANDS project. Figures 5.6 and 5.7 show the robot's goal, or



(a) *Resulting QTC states for this pass-by encounter:*
$H \leftrightarrow G$: $(+^{hg}0^{hg}+^{hg}0^{hg})$
$H \leftrightarrow R$: $(-^{hr}-^{hr}+^{hr}+^{hr})$
QTC_H *to action mapping:*
$(+^{hg}+^{hg}-^{hr}+^{hr}) \rightarrow (-^{hr}+^{hr})$

(b) *Resulting QTC states for this path crossing encounter:*
$H \leftrightarrow G$: $(-^{hg}0^{hg}-^{hg}0^{hg})$
$H \leftrightarrow R$: $(-^{hr}0^{hr}+^{hr}0^{hr})$
QTC_H *to action mapping:*
$(-^{hg}-^{hg}-^{hr}+^{hr}) \rightarrow (0^{hr}0^{hr})$

Figure 6.6: *Two examples of how to encode the human's movement using QTC_H and how to generate the state to action mapping.* H *represents the human,* R *the robot, and* G *the robot's goal. The blue double-cross is used for the relation between the human and the robot's goal and the red for the relation between the human and the robot. The human's movement from t_{n-1} to t_n is indicated by the black arrow and the robot's movement by the green arrow or by the green colour of it's circle when stationary. The captions of the sub-figures show the resulting QTC states where symbols from human and robot are indexed hr and the human symbols are printed in red and the symbols for the robot in green. The symbols for the relation of human and the robot's goal are indexed hg and the symbols describing the human movement are printed in blue.*

the direction of the goal using a read arrow pointing towards it. Thus, using the robot's goal as a virtual landmark facilitates classification without using any environment features to not compromise the abstraction from the metric world and guarantees to always be in a similar location depending on the topology of the map while explicitly encoding the intention of the robot.

As described in Section 4.2, the online QTC generation produces states in QTC_{BC} for *human* (h) and *robot* (r) and in QTC_C for *human* (h) and the *robot's goal* (g). Thus, the result of running the people perception and online QTC generation are the state chains $(q_1^{hr}\ q_2^{hr}\ q_3^{hr}\ q_4^{hr})$ for human and robot and $(q_1^{hg}\ q_2^{hg}\ q_3^{hg}\ q_4^{hg})$ for the human and the robot's

goal where $(q_1^*\ q_3^*)$ represent the movement of the human and $(q_2^*\ q_4^*)$ of the robot or the robot's goal. Since the robot's goal does not move during the interaction, $(q_2^{hg}\ q_4^{hg})$ are disregarded. Figure 6.6 shows two example states from two different interactions, i.e. pass-by in Figure 6.6a and path crossing in Figure 6.6b, and how the separate QTC states are put together to create the state of the human, which from here on will be called Qualitative Trajectory Calculus – Human (QTC_H), and the state of the robot. The robot state can either be the observed state during the learning phase or the state created by the live system. When only looking at the red symbols describing the human's movement in relation to the robot, the two interactions are equal but the required robot behaviour in green is completely different. Hence, the blue symbols for the relative movement of the human in relation to the robot's goal are added to disambiguate the interactions and include the robot's intention.

Given this representation it is rather straight forward to create the set of human observations $\boldsymbol{\Omega}$, the set of robot actions $\boldsymbol{\mathcal{A}}$, and the joint probability table $P(\boldsymbol{\Omega}, \boldsymbol{\mathcal{A}})$ by simply combining all the remaining 6 symbols for every time step t. Thus, each observed state $\Omega \in \boldsymbol{\Omega}$ consist of the 4-Tuple $(q_1^{hg}\ q_3^{hg}\ q_1^{hr}\ q_3^{hr})$ with $q_1^{hg}, q_3^{hg}, q_1^{hr} \in \{-, 0, +\}$ and $q_3^{hr} \in \{-, 0, +, \emptyset\}$ and each action $\mathcal{A} \in \boldsymbol{\mathcal{A}}$ consist of $(q_2^{hr}\ q_4^{hr})$ with $q_2^{hr} \in \{-, 0, +\}$ and $q_4^{hr} \in \{-, 0, +, \emptyset\}$. This results in a state space of $\boldsymbol{\Omega} \times \boldsymbol{\mathcal{A}}$ with $|\boldsymbol{\Omega}| = 108$ and $|\boldsymbol{\mathcal{A}}| = 12$, but only a small subset of these are actually observed during any given interaction. Figures 5.6 and 5.7 show a simulation example of a pass-by and a path crossing interaction encoded in QTC_{BC} and QTC_H, respectively. The resulting joint probability table can then be used to generate the model $\mathcal{M}$ for classification and state prediction as described in Section 6.1.1, the observation model $\mathcal{O}$ as described in Section 6.1.2, and to create the action selection policies as described blow.

6.2.2 Action Selection Policy Creation

In order for the robot to make an informed decision about its next best action, a policy has to be created that for each predicted state Ω generates an action $\mathcal{A}$ for the robot to execute. The set of all these state to action mappings from Equation 5.7 has previously been hand coded, but can now be learned from observation/demonstration using the new

representation of QTC_H for Ω and the joint probability table $P(\boldsymbol{\Omega}, \boldsymbol{\mathcal{A}})$ created above with which

$$P(\mathcal{A}_t|\Omega_t) = \frac{P(\mathcal{A}_t, \Omega_t)}{P(\Omega_t)} \tag{6.3}$$

can be calculated for every time t. Thus, $P(\mathcal{A}_t|\Omega_t)$ represents the probability of an action $\mathcal{A}_t \in \boldsymbol{\mathcal{A}}$ given the current state $\Omega_t \in \boldsymbol{\Omega}$. The resulting probabilities are then stored in a conditional probability table $\mathcal{P}(\boldsymbol{\mathcal{A}}|\boldsymbol{\Omega})$ for the easy look up of $\boldsymbol{\Omega} \rightarrow \boldsymbol{\mathcal{A}}$ at runtime. Thus, this table simply contains rows for every observed human state $o \in \mathbf{o}$ with $\mathbf{o} \subseteq \boldsymbol{\Omega}$ stating the joint probabilities for every observed action $a \in \mathbf{a}$ with $\mathbf{a} \subseteq \boldsymbol{\mathcal{A}}$.

To acquire $P(\boldsymbol{\Omega}, \boldsymbol{\mathcal{A}})$, a Wizard of Oz set-up is used where the robot is remote controlled during the interaction with the human interaction partner and the generated QTC state chains are recorded. Subsequently, these state chains have to be labelled based on the interaction type and are then clustered. For each interaction type a separate model $\mathcal{M}_m$ is generated from the set of observed states $\mathbf{o}_m \subseteq \boldsymbol{\Omega}$ to create the set of models $\boldsymbol{\mathcal{M}}$ for the particle filter. Similarly, to generate the set of conditional probability tables $\boldsymbol{\mathcal{P}}$, for each interaction type a conditional probability table $\mathcal{P}_m$ is generated from the set of observed actions $\mathbf{a}_m \subseteq \boldsymbol{\mathcal{A}}$ using $P(a_m^i|o_m^i)$ with $a_m^i \in \mathbf{a}_m$ and $o_m^i \in \mathbf{o}_m$.

Given this $\mathcal{P}_m \in \boldsymbol{\mathcal{P}}$, it is very straight forward to get the joint probability for all possible actions $a^i \in \mathbf{a}^\Omega$ where $\mathbf{a}^\Omega \subseteq \mathbf{a}_m \subseteq \boldsymbol{\mathcal{A}}$ is the set of all observed robot actions for the interaction type m for the given observation Ω. With this list of probabilities $P(a^i|\Omega)$ one can follow different strategies for action selection. A Monte-Carlo-based sampling using the probabilities as weights leads to inconsistent and illegible behaviour due to its chaotic nature. Thus, the system follows the greedy approach for policy creation

$$a^\star = \underset{a^i \in \mathbf{a}^\Omega}{\arg\max}\, P(a^i|\Omega) \tag{6.4}$$

The optimal policy $\pi^\star$, therefore, is just a mapping of the currently observed state Ω to the arg max of the corresponding entry in $\mathcal{P}_m$. This could also be turned into a ϵ-greedy approach which would allow for exploration of different actions instead of just exploiting the previously acquired knowledge. For the experiment described in the following, however, the greedy approach is used.

6.3 The System in Summary

To summarise before going into detail about the conducted experiment, the final system for human-aware navigation requires a set of interaction models $\boldsymbol{\mathcal{M}}$ and a set of action selection conditional probability tables $\boldsymbol{\mathcal{P}}$, which are acquired in a Wizard of Oz set-up that can either use LfD or follows the traditional Wizard of Oz principle. During this model acquisition, the QTC states of human and robot, and human and the robot's goal are recorded and then rearranged to create this set of interaction models $\boldsymbol{\mathcal{M}}$ that describe the human behaviour and the set of conditional probability tables $\boldsymbol{\mathcal{P}}(\boldsymbol{\mathcal{A}}|\boldsymbol{\Omega})$ describing the probability of a robot action given a human observation. The set of models $\boldsymbol{\mathcal{M}}$ and the set of states $\boldsymbol{\Omega}$ is used in the particle filter to predict the current state of the world $\Omega \in \boldsymbol{\Omega}$ and to classify the current interaction type $\mathcal{M} \in \boldsymbol{\mathcal{M}}$. This process works in real-time with incrementally updated QTC state chains and produces the current best state $\Omega^\star$ and the best interaction model $\mathcal{M}^\star$. Both $\Omega^\star$ and $\mathcal{M}^\star$, therefore, form the belief of the current state of the world. Given $\mathcal{M}^\star$ the conditional probability table $\mathcal{P}_{\mathcal{M}^\star} \in \boldsymbol{\mathcal{P}}$ is used to determine the next best action $\mathcal{P}_{\mathcal{M}^\star}(\mathcal{A}|\Omega^\star)$ using Equation 6.4. Given this resulting best action $a^\star \in \boldsymbol{\mathcal{A}}$, the Velocity Costmap for $a^\star$ is generated using Table 5.1 to calculate the constraints ξ_α and ξ_ϱ and Equation 5.8 to find the low cost areas and then sent to the DWA local planner. This whole process of belief generation and velocity costmap generation has a complexity of $O(n)$ and, therefore, scales linearly with the number of particles. For the following experiment $N_p = 1000$ particles were used which results in a total computation time of $\sim 0.001s$. The major limiting factor is the QTC state generation that smoothes the observed trajectories by calculating the average over an interval of $0.3s$.

This fully autonomous system is tested in the following to on the one hand evaluate its legibility, sociability, and safety and on the other hand its task efficiency regarding distance and time travelled.

6.4 Experiment

To evaluate the new QTC based representation, the particle filter for online classification and belief generation, the velocity costmaps in a more realistic scenario, and the action selection, the experiment consisted of two distinct parts using two separate sets of par-

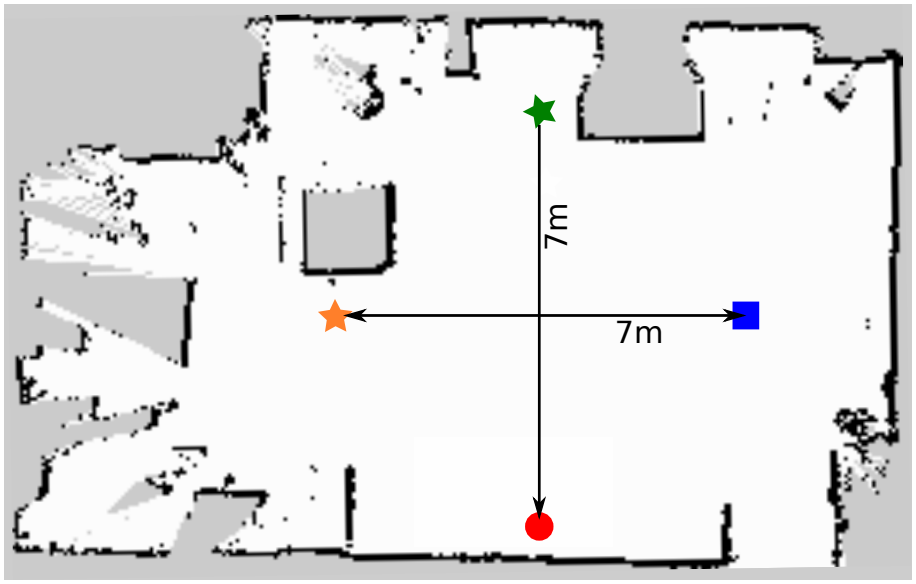


Figure 6.7: *The SLAM map of the area of the experiment. Green star: human starting position for path crossing, orange star: human starting position for pass-by and robot goal in both conditions, blue square: robot starting position in both conditions and human goal for passby, red circle: human goal for path crossing.*

ticipants. The first part was dedicated to the model acquisition using naïve participants using Learning from Demonstration (LfD) and the second part was the evaluation of the learned behaviour, the interaction classification, state prediction and action selection, and the behaviour generation. Both experiments were conducted using *Linda* the robot in a lab like environment with clearly marked start and goal positions for human and robot (see Figure 6.7). To track humans and generate QTC states, the system described in Chapter 4 was used only relying on the robot's sensors to create more realistic conditions.

6.4.1 Model Teaching

For the acquisition of $P(\boldsymbol{\Omega}, \boldsymbol{\mathcal{A}})$ to generate $\boldsymbol{\mathcal{M}}$ and $\boldsymbol{\mathcal{P}}$, QTC states generated using an "Inverse Oz of Wizard" set-up as described by Lichtenthäler et al. (2013) were recorded. This approach follows the "Oz of Wizard" design by Steinfeld et al. (2009) but replaces the simulated human with one of the experimenters and has the participant control the robot as described in Section 2.4.1. To this end, a Logitech F710 wireless gamepad was used to control *Linda*, with the rotational movement of the robot mapped onto the horizontal axis of the left control stick, and the translational movement onto the vertical axis, respectively. Since the QTC model described in Chapter 3 cannot express speed, the com-

manded translational speed was thresholded to be binary with either 0 or $0.55m/s$. Thus, the robot could either stop and wait or actively avoid the human at maximum speed; no gradual slow down or similar approaches were possible. The only recordings made were the generated QTC states and the trajectories of human and robot.

Five participants (4 male, 1 female) were recruited who all worked in the School of Computer Science of the University of Lincoln either as PhD students or Researchers. The age of the participants at the time of the experiment was 23, 27, 28, 31, and 33 and 3 were European and 2 two of Asian decent which means that due to their cultural background, their definition of proxemics (Hall 1969) might differ. On a scale from 1 (very low) to 5 (very high) they rated their experience in Computer Science with a mean of 4.6 and their experience in Robotics with a mean of 4.0. Due to the robot being stationed in the open plan L-CAS office, they had all interacted with *Linda* before in one way or another but were naïve to the system described in this thesis. The participants were informed about the goal of the study and were instructed to (while the experimenter is interacting with it) remote-control the robot in a way they would like *Linda* to behave if they were in the place of the experimenter. They were also informed about the speed constraint, i.e. the binary speed described earlier, and the reason behind it. In an initial *training phase*, every participant was given as much time as they wanted to practice the control of the robot before the start of the recording. To this end, the experimenter interacted with the robot as they would during the experiment while the participant was trying to control the robot to their liking. Once the participant expressed their confidence in being able to perform the task in a way they would like the robot to perform, the experiment started. None of the training trials were used for evaluation. No compensation was paid before or after the experiment.

During the test trials and the actual experiment, as described by Lichtenthäler et al. (2013), the experimenter played the role of the human interacting with the robot. The starting points and goals for human and robot were marked on the floor and can be seen in Figure 6.7. After each trial, the robot drove to its starting position autonomously to ensure the same initial state for every trial. The experimenter walked towards the goal along a straight line with a constant speed, matching the maximum speed of the robot. Two different scenarios were recorded, i.e. *pass-by* (p) and *path crossing* (c). Before the start of the training phase, the participants were primed to always avoid the human by going

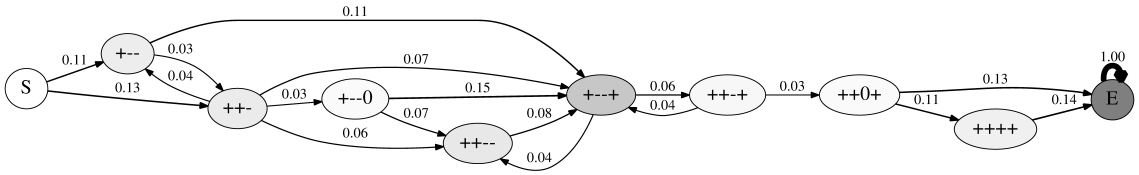


Figure 6.8: *The learned Markov Model $\mathcal{M}_p \in \boldsymbol{\mathcal{M}}$ for the pass-by scenario. The human moves away from the goal and towards the robot, passes on the right, and finally moves away from goal and robot. The numbers represent transition probabilities and everything below* 0.03 *has been pruned for visability. Grey level: a-priori probability of occurance of the state from 0 (white) to 1 (dark grey). The two leftmost states only have 3 symbols because the $\emptyset$ of QTC_{BC} is not represented in this visualisation. There is no QTC_B phase towards the end of the interaction because the human is not observed once she passed the robot. Hence, $d(k,l) < d_s$ until the human is not observed any more and the interaction is "complete".*

to the right in the pass-by interaction and to stop and wait until the human had passed in the path crossing – the latter being inspired by the findings of Kruse et al. (2012) and Lichtenthäler et al. (2013) – but were not given any additional constraints regarding, e.g. distance or magnitude of avoidance. Each participant recorded 7 interactions per scenario summing up to a total of $5 \cdot 2 \cdot 7 = 70$ runs. If during the experiment the participant felt that they controlled the robot in manner they did not intend, the trial was excluded and repeated until the total number of 7 was reached.

The recorded QTC states were used to create the set of models $\boldsymbol{\mathcal{M}} = \{\mathcal{M}_c, \mathcal{M}_p\}$ and conditional probability tables $\boldsymbol{\mathcal{P}} = \{\mathcal{P}_c, \mathcal{P}_p\}$ for crossing c and pass-by p as described above. Figure 6.8 shows the Markov Model $\mathcal{M}_p$ generated from the recorded data.

6.4.2 Evaluation Experiment

A group of 12 participants with a mean age of 19.5 and standard deviation of 2.8 was recruited, which exclusively consisted of male Computer Science students. With one exception, none of them had interacted with a robot before and on a scale from 1 (very low) to 5 (very high) they rated their experience with robots with 1.5. All participants were naïve to the specific goal of the study and the robot platform used.

To evaluate the legibility, sociability, saftey, exprienced comfort, and task efficiency of the approach, it was compared to the commonly used Gaussian cost model (e.g. Sisbot et al. 2007) based on proxemics (Hall 1969) similar to the study presented in Chapter 5, for the same reason of it being one of the standard ROS approaches to human-aware navigation. These costs, like the Velocity Costmaps also introduced in Chapter 5, were

added to the DWA local planner using the local costmap described in Section 5.2.2. Each participant faced all conditions in a within participant design to be able to compare the results for all conditions without the influence of interpersonal differences in observations or preferences regarding robotic navigation. The conditions were randomised for each participant, and the participants were unaware of which condition they were facing. The different interactions, i.e. pass-by and path crossing, were randomized between participants, but always conducted in blocks, meaning that a participant would either start with 10 iterations of the path crossing and then finish with 10 iterations of the pass-by or vice-versa but they were never interleaved. After each single trial in both scenarios, a questionnaire using a Likert scale from 1 (fully disagree) to 5 (fully agree) was completed, asking for:

Q1 I felt safe when I encountered the robot in this situation.

Q2 I felt comfortable when I encountered the robot in this situation.

Q3 I was able to follow my intended path with no disruption by the robot in this situation.

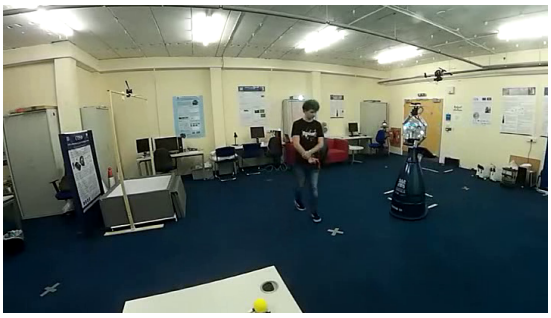
Q4 I did not have to slow down to let the robot pass.

Q5 The robot behaved appropriately in this situation.

Additionally, participants were given the chance to mention anything out of the ordinary that had occurred using a closed-ended question also allowing to specify "other" (see Figure 6.10). Question Q4 and its results were inverted to simplify the presentation. During the experiment, participants were asked: "I had to slow down to let the robot pass" with 1 (fully disagree) to 5 (fully agree) to not interleave positive and negative questions. Regarding the appropriateness of the behaviour in question Q5, the participants were asked to put themselves into a situation where they would have encountered the robot in the wild (e.g. as a tour guide in a museum) and if they would have thought that it would be appropriate for the robot to behave the same way they just encountered. Hence, this question investigated the sociability of the executed behaviour. The used questionnaire can be found in Appendix C.1.

Before the experiment, participants were introduced to the robot and the lab environment. The experimenter explained and demonstrated at which marker the participants

(a) *Pass-by interaction using Velocity Costmaps and the particle filter. The robot moves out of the way of the human.*



(b) *Crossing interaction using Velocity Costmaps and the particle filter. At this point the robot is stationary.*

Figure 6.9: *Examples of pass-by and crossing for the same participant. The robot had no a-priori knowledge of the interaction type and used the particle filter with the learned models $\mathcal{M}$ for the belief classification and $\mathcal{P}$ for action selection.*

would start and towards which marker they should walk for each of the two conditions, i.e. pass-by and path crossing. The general goal of the study, i.e. the investigation of a mobile robot's movement behaviour on the experienced safety and comfort of a human interaction partner was disclosed to the participants before the experiment but the underlying conditions and working principles were only explained after the trails were complete. For the interaction, participants were given a wireless controller with which they could start the interaction or stop the robot in case they feared it would come too close. The robot itself drove autonomously from its starting position to the goal using one of the described avoidance strategies. Participants were asked to start walking when they pressed the button, trying to match the robot's speed, but to assume a constant velocity if possible. Before the start of each of the conditions, i.e. pass-by and path crossing, participants were asked to use the controller to start the robot while they stood outside of the experimental area to observe the robot's speed and get a feeling for when to start walking. No additional training phase was used. As a compensation for their time, participants could enter a raffle for one of two Amazon vouchers with a value of 10 GBP which were randomly distributed after the experiment was finished.

In the path crossing scenario, participants were asked to follow a straight line if the robot permits. In the pass-by scenario, on the other hand, participants were told to partake in the avoidance manoeuvre in a way they thought would be appropriate given the behaviour of the robot. This could either be waiting for the robot to move out of

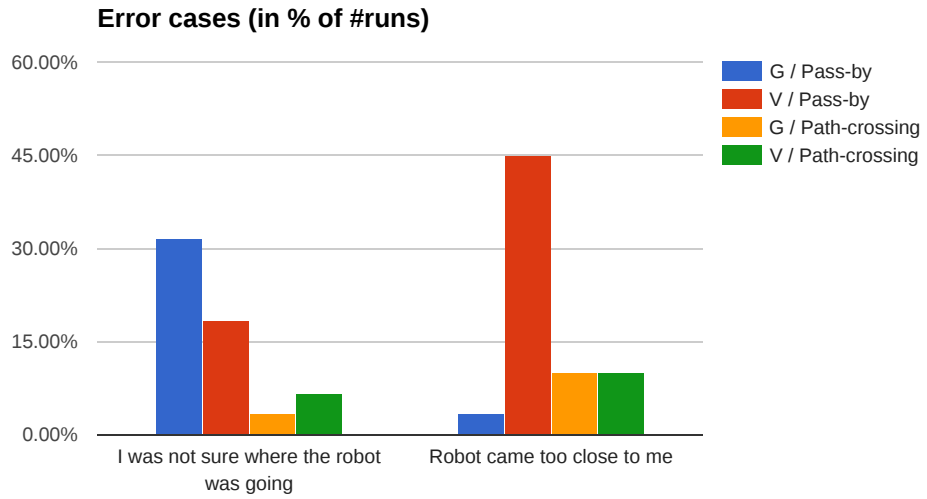


Figure 6.10: *The errors specified by the participants after each trial. Errors that occured less than* 10% *of the total amount of runs are not shown. The ommited errors were:* robot made physical contact, robot stopped too early and abrupt, robot stuttered, *and* robot did not reach the goal.

the way or moving around the robot themselves. Both scenarios were repeated 10 times (5 times per condition) resulting in $12 \cdot 5 = 60$ trials for each condition in each scenario summing up to a total of $2 \cdot (2 \cdot 60) = 240$ runs for both scenarios and conditions. Figure 6.9 shows one of the participants interacting with the robot in the particle filter based Velocity Costmap condition.

For all conditions, the robot had no a-priori knowledge of the interaction type, which is irrelevant for the Gaussian cost models, but for the particle filter condition entails the online classification of the current belief $bel(\mathbf{X})$ which includes the interaction type $\mathcal{M} \in \boldsymbol{\mathcal{M}}$ and the current state of the world $\Omega \in \boldsymbol{\Omega}$. To determine the robot's next action, greedy action selection using $\mathcal{P}_{\mathcal{M}}(\mathcal{A}|\Omega)$ following Equation 6.4 from the conditional probability table $\boldsymbol{\mathcal{P}}$ was used. Hence, the action $a = (q_2, q_4)$ to be executed was transformed into the corresponding Velocity Costmap and then sent to the DWA planner as described in Chapter 5. As a result, the planner would only allow trajectories $\mathfrak{t} \in \mathfrak{T}$ that create the desired QTC state. The weights $\omega_i \in \boldsymbol{\omega}$ were the same as described in Section 5.3. To create equal conditions for both approaches, compared to the experiment in Section 5.3, the Gaussian costmodel was only used on the local costmap and not on the global one because the Velocity Costmaps work only with the DWA planner as an informed reactive system as well.

Table 6.1: *Questionnaire: Mean Likert scale responses, 95% confidence intervals, and p values. Bold values are mentioned in text.*

	G pass-by	p	V pass-by	G crossing	p	V crossing
Q1	**4.90±.09**	$****$	**4.55±.14**	4.92±.08	–	4.82±.13
Q2	4.45±.20	–	4.43±.18	4.72±.14	–	4.67±.15
Q3	**3.63±.34**	–	**3.85±.27**	4.97±.05	–	4.92±.13
Q4	3.85±.38	–	3.88±.33	4.98±.03	–	4.92±.08
Q5	**4.00±.29**	–	**4.00±.27**	4.80±.14	–	4.77±.16

6.4.3 Results

For convenience, in the following the approach using Velocity Costamps and the particle filter based classification will be abbreviated to V and the Gaussian cost model will be denoted by G. After excluding trials where the human was not detected or detected too late (generated less than 5 sample generations using the particle filter) from both conditions, a total number of 107 valid crossing and 116 pass-by encounters out of the 120 per condition remained. According to convention, in the following, p-values of statistical tests signifying significant differences between two values will be denoted by one to four asterisks ($*$) for $p < 0.05$, $p < 0.01$, $p < 0.001$, and $p < 0.0001$, respectively. All values shown in this section are either the mean values of the results and their respective 95% confidence interval or the percentage of number of runs.

The results of the questionnaire in between runs can be seen in Table 6.1. With the exception of the reply to question Q1 in the pass-by scenario with $p < 0.0001$ in a two-tailed unpaired t-test, none of the results show statistical significance. To highlight the most important findings (bold values in Table 6.1), participants rated their feeling of safety significantly higher in the G pass-by condition $4.90 \pm .09$ than in the V pass-by $4.55 \pm .14$, participants could better follow their path in the V pass-by condition $3.85 \pm .27$ than in the G pass-by $3.63 \pm .34$, and they rated G pass-by and V pass-by the same for appropriateness with a $4.00 \pm .29$ for G pass-by and $4.00 \pm .27$ for V pass-by. All questions in the crossing scenario yielded virtually the same results.

According to the participants feedback in Figure 6.10, the errors that occurred the most during the trials was either the robot coming too close to the human or the participant not knowing where the robot wanted to go. To highlight the most important findings, in 31.67% of the cases in the G pass-by and 18.33% of the V pass-by encounters the

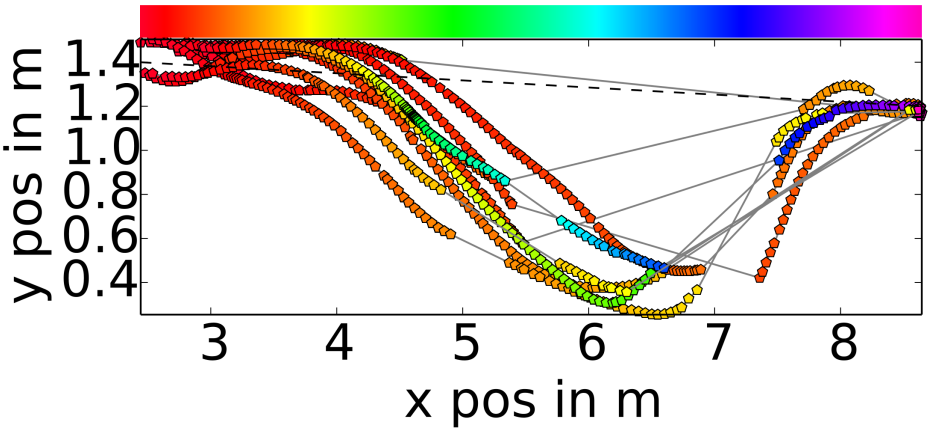
Table 6.2: *Mean values, 95% confidence intervals, and p values for minimum distance to human (M), distance travelled (D), travel time (T), and speed (S) over all runs*

	G pass-by	p	V pass-by	G crossing	p	V crossing
M	0.68±.04	$****$	0.54±.04	1.04±.05	$*$	1.18±.09
D	6.66±.24	$****$	7.18±.06	6.95±.02	–	6.96±.03
T	28.67±1.7	$****$	17.92±.63	18.10±.33	–	18.49±.45
S	0.25±.02	$****$	0.41±.01	0.39±.01	–	0.38±.01

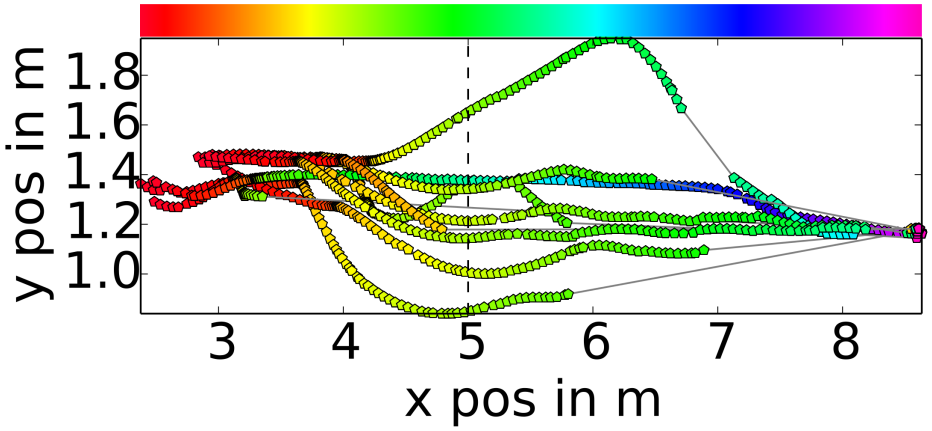
participant was not sure where the robot was intending to go. One of the participants was so confused by the behaviour of the robot in the G pass-by condition that he stood in front of the robot until its goal timed out and the interaction had to be interrupted. The feedback given was that he was not sure what to do at all and waited for the robot to initiate an action. This is reflected as the error "robot did not reach the goal" among the errors specified by the participants (see Figure 6.10). The difference between V pass-by and G pass-by, however, only reaches $p = 0.144$ (two-tailed) in a Chi squared test with 1 degree of freedom. Also, the crossing values for this error are not significantly different from each other. Looking at the robot being too close to the participant, in 45.00% of the V pass-by and only 3.33% of G pass-by this is the case with $p < 0.001$ in the same test. As for the previous error, the crossing values are not significantly different.

In addition to the questionnaire, sensor data from the robot was collected and the mean values of the minimum distance kept to the human, the distance travelled, the travel time and, the speed were evaluated. The results can be found in Table 6.2 and show a significant difference comparing G pass-by and V pass-by with $p < 0.0001$ in a two tailed unpaired t-test. V pass-by achieved a significantly greater speed and a reduced travel time even though the travelled distance was significantly higher. On the other hand, the minimum distance kept to the human is significantly lower than in the G pass-by condition. For the crossing scenario, the only significant difference exists for the minimum distance kept from the human with $p < 0.05$ where V crossing kept a larger distance to the participant. The other results for the crossing scenario are virtually the same.

To evaluate the classification rate of the particle filter with $N_p = 1000$ particles, the number of particles per model $\mathcal{M}_m \in \boldsymbol{\mathcal{M}}$ was recorded after each sampling step. Whichever model had more particles got one vote. After the interaction, the model that had the most votes was considered to be the best performing one and therefore the overall



(a) Pass-by shows an avoidance movement half way through the interaction when encountering the human.



(b) Path crossing shows an abrupt transition from red to yellow, visualising where the robot stopped and waited for the human to pass.

Figure 6.11: *Ten randomly sampled trajectories generated using Velocity Costmaps; black dashed line: approximate human trajectory. The robot travelled from left to right and its trajectory is colour coded according to the shown scale to visualise time. The positions are only shown if the human was detected at the same time because the Velocity Costmaps only then had effect. For visualisation purposes the positions are connected via line plots. This also shows one of the biggest problems of HRI and HRSI in general, the behaviour of the robot is undefined if the human is not detected. Given the DWA local planner based approach, the robot still generates inherently safe non-human-aware trajectories even without any Velocity Costmaps being published. If no Velocity Costmap is available, the last one is used for $\Delta t = 2s$, afterwards the vanilla DWA takes over.*

classification result. On average the crossing model $\mathcal{M}_c$ achieved $13.37 \pm .67$ votes in the crossing scenario and $2.88 \pm .27$ in pass-by whereas the pass-by model $\mathcal{M}_p$ achieved on average $0.79 \pm .24$ votes in the crossing scenario and 17.92 ± 1.55 in pass-by. Using a simple diagnostics test considering the correctly classified crossing encounters as true positives (TP), the correct pass-by as true negatives (TN), the false pass-by as false positives (FP), and the false crossing as false negatives (FN), a sensitivity of 97.20% with the 95% confidence interval of 92.02% to 99.42% and a specificity of 97.41% with a 95% confidence interval of 92.63% to 99.46% was achieved. With a null-hypothesis of $H_0 = 50\%$ for a two class classification problem, this shows highly reliable classification results. Additionally, as can be seen from Figure 6.11 the action selection and Velocity Costmaps produced trajectories that conform with the given priming of avoiding to the right in pass-by and stopping and waiting in the path crossing.

6.5 Discussion

The experiments indicate that using Learning from Demonstration (LfD) via an Inverse Oz of Wizard set-up, the robot is able to learn human-aware navigation behaviour that in most regards performs as well as one of the most commonly used approaches when it comes to participant experience and subjective assessment. On top of that, it was able to ***reach the goal in significantly shorter time and with a higher average speed [Obj. 2.3]*** having ***no trade-off in perceived appropriateness (sociability) [Obj. 2.1] of experienced comfort [Obj. 2]*** of the behaviour. However, participants reported that the robot came too close to them in the pass-by scenario in 45% of the cases and they felt significantly less safe even though the total difference for this question is only 0.35 on a 5-point Likert scale. When it comes to the perceived ability of following ones path, participants rated the approach higher than the standard Gaussian cost model, but it did not reach statistical significance. Looking at the legibility of the behaviour, the Gaussian cost model produced trajectories that in 31.67% of the cases were marked by the participants as *I wasn't sure where the robot is going [sic].* The presented approach, on the other hand, only left the participant in doubt of the robot's intention in 18.33% of the trials which supports the hypothesis of ***being more legible [Obj. 2.1]***. This effect, however, came short of statistical significance with $p = 0.14$. All the above mentioned outcomes

are only taken from the pass-by scenario, for crossing there was virtually no difference in participant responses. Analysis of the collected sensor data for crossing shows that the robot kept a slight but significantly larger distance to the participant while waiting for them to pass, giving them more space to manoeuvre.

Looking at the performance of the system itself, it achieved classification rates of 97% with only very few iterations of the sampling algorithm. This shows that the model is able to represent the scenarios given and reliably classify them and pick the corresponding best action. Looking at the trajectories generated in Figure 6.11, the robot performed the learned behaviour given the priming of stopping and waiting in the crossing and going to the right in the pass-by scenario. Hence, the created conditional probability tables model the robot behaviour correctly and allow to make correct decisions with regard to the learned behaviour.

Limitations The biggest limitation is the number and diversity of participants. With a more diverse group it might have been possible to achieve more distinct results, but it was impossible to recruit these for the presented evaluation given time constraints. Nevertheless, for the evaluation study, naïve participants were used who have never interacted with a robot before which already gives an indication about the generated behaviour. Coming too close to the participant was one of the major issues during the pass-by interaction which is mainly caused by the robot not having enough time to avoid the human due to late detections. The Gaussian cost model mainly had the robot stop and wait for the human to circumvent it as it was quickly trapped in a local cost maxima and therefore kept larger distance, but did not actively avoid the human. The magnitude of avoidance has therefore to be increased for future experiments. This stopping behaviour of the Gaussian cost model is also the reason why the crossing scenarios are so similar and why the participants found this behaviour less legible.

6.6 Summary

This chapter introduced state prediction and action selection for HRSI based on a particle filter for QSRs. It describes how to build the prediction and observation model and how the particle filter uses them to model the current belief of the world. Each particle

represents the current state it is in and the model that produced it. By a simple majority vote over the particles the best state and best model are determined where the best model represents the current interaction. For action selection, a joint probability table is built for each model that states the joint probabilities of all actions given a specific state. Using the current best model and the best state the joint probability of that model provides all the possible actions for the current state. The best action is selected in a greedy fashion where always the action with the highest probability is selected.

To overcome the problem of separating the current state of the human and the robot and still have a meaningful representation that is unambiguous enough to classify the current interaction type and ***facilitate decision making [Obj. 1.6]***, a new QTC like model is introduced that uses parts of the tuple describing the human and robot state in QTC_{BC} and parts of the tuple describing the interaction between the human and the robot's goal in QTC_{C}. These symbols are then merged into a new QTC variant called Qualitative Trajectory Calculus – Human (QTC_{H}). While QTC_{H} does not represent a real QTC variant per-se, it builds on established variants and inherits all their properties. Moreover and most importantly, it allows to reliably classify the current interaction type without considering the actions of the robot and, thereby, allows to separate the two for the required decision process.

In order to build all the necessary models, some form of Wizard of Oz can be used to record the QTC sequences which has been demonstrated in the experiment section using the "Inverse Oz of Wizard". These two experiments consisted of a learning phase and an evaluation phase with two separate sets of participants. The experiments showed that the particle filter is able to reliably classify the interaction type given the perception pipeline works as expected and that the velocity costmaps generated the correct behaviour given the priming of avoiding to the right and stopping and waiting during the learning phase.

To summarise, this chapter addresses the remaining objectives of creating a model that ***facilitates decision processes [Obj. 1.6]*** by being able to separate the robot state from the human state, and creating ***legible and sociable [Obj. 2.1]*** robot behaviour by incorporating human judgement into the creation of this model and, therefore, allowing the demonstrator to determine the optimal behaviour for the robot while at the same time being ***task efficient [Obj. 2.3]***. Also the ***fast and flexible action selection [Obj. 2.4]*** is implemented using the particle filter that works in real-time and is able

to recover from false classifications or adapt in case the human behaviour changes. The ***autonomy [Obj. 3]*** of the system is given by it ***using only the on-board sensors and processing power [Obj. 3.2]*** to determine the current state of the world and predict the next best action for the robot in ***real-time [Obj. 3.1]***. By using the Velocity Costmap for behaviour generation, it is ***tailored to the robot hardware [Obj. 3.3]*** and ***inherently safe [Obj. 2.2]***. However, the participants reported that they felt less safe which means that the perceived safety was reduced due to the late detections of the human and the reduced time for avoidance. This, however, according to the questionnaire did not diminish the experienced comfort and shows that the system scales well given noisy detections.

—There is no real ending. It's just the place where you stop the story.

Frank Herbert, Science Fiction Author

7

Discussion and Conclusion

Keeping in mind the aim of this work to create a Human-Robot Spatial Interaction (HRSI) approach focussing on human-aware navigation that is able to abstract from the underlying metric representations of the world, thereby, allowing to easily transfer knowledge between similar encounters with different people in different locations in an ever changing environment, we have seen one possible approach of accomplishing that using Qualitative Spatial Relations (QSR) which presents a novel approach to human-aware navigation. This work introduces the Qualitative Trajectory Calculus – Basic/Double-Cross (QTC_{BC}) which describes the movement of human and robot in relation to each other over the interval $T = [t_{n-1}, t_n]$ and has been especially designed for HRSI to not only highlight the interaction of both agents in close vicinity but to also model a discrete distance threshold d_s that is used to trigger state changes in the robot's behaviour generation. Representing the motion of human and robot in the same state allows to reliably classify different interaction types offline, given a chain of these states per interaction, using Hidden Markov Models (HMMs) which shows that this representation is well suited for activity recognition.

However, using both the human and the robot state to classify the current interaction type is also a limitation when generating behaviour for the robot as in this case it is not known a-priori and can, therefore, not be used for classification. Using only the human state of the QTC_{BC} tuple, the description becomes too ambiguous to distinguish even the most basic activities like pass-by and path crossing where in both cases the human approaches the robot but the actual interaction is entirely different. To overcome this issue and to be able to separate out the robot state for behaviour generation, the Qualitative Trajectory Calculus – Human (QTC_H) has been introduced which is not in itself a version of the Qualitative Trajectory Calculus (QTC) but a combination of QTC_{BC} for the human and the robot using the symbols describing the human's movement and QTC_C states for the human and the robot's goal. This virtual landmark and the QTC_C state between it and the human allow to once again classify different interaction types reliably and also encode the intention of the robot by using its short term goal on a topological map. Thus, from observation, the joint probability $P(\boldsymbol{\Omega}, \boldsymbol{\mathcal{A}})$ can be created which contains the probabilities for the set of human observations $\boldsymbol{\Omega}$ and the robot actions $\boldsymbol{\mathcal{A}}$ that have been observed during the Learning from Demonstration (LfD) phase.

In order to be able to perform any kind of Human-Robot Interaction (HRI) in general and HRSI in particular, the robot has to be able to observe humans in its close vicinity. Merely detecting them, however, is not sufficient for the generation of QTC states. Since QTC needs two consecutive observations of the movement of both agents to create a single state, the detections of the humans have to be consolidated into tracks. To this end, this work uses pre-existing detection and tracking approaches which have been combined in a joint framework for the detection and tracking of humans in the robot's vicinity. These tracks, together with the robot's self-localisation, are then used for the fully automatic state chain generation, producing QTC states in all described variants between the human and the robot and the human and the robot's goal. These state chains are also validated on the fly according to the legal transitions defined in QTC itself. This approach of abstracting from the underlying metric world by generating QTC state chains using the robot's on-board sensors only, is a basic requirement to create a fully autonomous robot behaviour and is the corner stone for all the experiments conducted on the robot.

By abstracting from all underlying metric representations, like speed, acceleration, angle of avoidance, etc. this QTC representation becomes very versatile and robust to

change but loses almost all information that is necessary to generate behaviour from it. The only information encoded is the direction of the robot's velocity vector $\overrightarrow{\nu}$ and a distance threshold d_s. The presented work, however, shows that explicit knowledge about the speed at which the robot circumvents the human for example is not necessary to generate human-aware trajectories. At the time of writing this thesis the Robot Operating System (ROS) uses a combination of a global and local planner which is also used on *Linda* the robot and many other research and industrial robots all around the world. This global planner produces an energy efficient path using Dijkstra's algorithm and employs the sampling based Dynamic Window Approach (DWA) local planner for obstacle avoidance and command velocity generation which control the speed of the wheels. This mentioned sampling approach looks at a range of velocity samples, so-called trajectories $\mathfrak{t} \in \mathfrak{T}$, which are scored based on several critique functions that look at the goal-directedness and the distance to obstacles. The trajectory $\mathfrak{t}_i$ with the lowest costs is then executed for a certain planning horizon Δt at which time the sampling process starts all over again. In order to create behaviour that conforms with the desired QTC state of the robot, Velocity Costmaps are introduced into the set of critique functions $\boldsymbol{\mathcal{C}}$ which assign lethal cost value to trajectories $\mathfrak{t}_i$, representing the velocity vector of the robot $\overrightarrow{\nu_i}$, that do not have the same direction as the velocity vector described by the QTC state. This way, since lethal trajectories are discarded regardless of the weighting of the critique function, the DWA planner will only produce trajectories that conform with the model or stop and wait if no valid trajectory could be found. Moreover, since every critique function $c_i \in \boldsymbol{\mathcal{C}}$ has an associated weight $\omega_i \in \boldsymbol{\omega}$, the "human-awareness" of the planner can be determined by assigning it a higher or lower weight than the goal-directedness and obstacle avoidance scoring functions which influences the magnitude of the avoidance manoeuvre. Finally, in case there is no Velocity Costmap published, because there is no human in the vicinity of the robot, the DWA planner functions as usual producing safe trajectories that might not be legible or human-aware but circumvent static and dynamic obstacles quite reliably.

The final task to accomplish to achieve autonomous robot behaviour is the online classification of the current interaction type based on incremental updates of the state chain and the online belief generation for action selection. HMMs are well suited for offline classification of the interaction type but cannot be used online because they need the entire state chain. There are approaches to circumvent this limitation but in order to

be robust to false classifications or behaviour change of the human, and never observed transitions or states, a Monte-Carlo-based approach has been implemented to solve this problem. The resulting particle filter for generic QSRs uses a prediction model based on observed human states and an observation model based on the properties of the used qualitative representation. Using incremental updates, the filter converges to a belief that includes the current state the human is in and the model $\mathcal{M}$ that predicted this state. A simple majority vote is used to create this belief and a simple look-up for action selection in the conditional probability table $\mathcal{P}_{\mathcal{M}}$ associated with this model yields the best action for the robot to take. Both model and probability table are learned from observation and demonstration respectively and therefore also encode human judgement for behaviour generation. To allow for behaviour change during the interaction if required and prevent over-fitting of the particle filter, a starvation factor is introduced that determines the amount of particles that are chosen randomly disregarding the current belief. By doing so, the filter is able to recover from false classifications or a change in the behaviour of the human while still reliably representing the current state of the world and classifying the current interaction type.

7.1 Discussion

Since all the approaches have been discussed in their respective chapters, this discussion focuses more on the overall system and its application.

There are many different ways of creating a representation of HRSI that is not dependent on the underlying metric representation, the most common example being Gaussian cost functions based on proxemics. These just assign higher costs to trajectories that lead the robot closer to the human, but the highest costs are still assigned to obstacles, thereby, creating safe an collision free behaviour. However, this approach does not encode any high-level knowledge about the interaction type and purely relies on the planner to generate the movement behaviour. Since this reactive planner is sampling-based and evaluates a new sample every Δt, this plan might change entirely from one sampling step to the other. If the human, for example, shifts a little to the left, the planner will find more open space on the right and plan a path through there, if the human shifts back to the right, the planner will then re-plan and try to pass the human on the left. This

Markovian behaviour in general enables the robot to circumvent dynamic obstacles, even if not in all situations as we have seen from the described experiments, but does not produce legible nor sociable behaviour. In order to achieve motion that is understandable by the human counterpart, the robot behaviour has to show some form of commitment to one specific behaviour, e.g. if the human or robot start the circumvention to the right, it should not switch to the left half way through the interaction just because there is a little more space. One way of achieving this is to plan ahead for several time steps and to follow this plan until replanning is made necessary due to unforeseen human behaviour. Most of these approaches also work on cost models in the environment and while generating a certain form of commitment are not able to encode more complex interactions. This work, therefore, presented the use of QTC for HRSI and a first approach of using QSRs for human-aware navigation and human-robot joint motion. The results of especially the last experiment have shown that participants considered the behaviour generated by the presented approach much less confusing and hence more legible.

Using qualitative representations to encode real-world interactions is not a novel approach and many different representations have been introduced in the past. QTC has been selected for this specific implementation, because it describes the motion of two Moving Point Objects (MPO) in relation to each other which is one of the main tasks of HRSI. Other representations such as Cardinal Directions in combination with distance could have been used, but these only represent positions and not motion and, additionally, QTC provides a much narrower state space that is well suited to described relative movement. Using only the "simplified" versions of this calculus, i.e. disregarding the relative speed and angle, might at first appear as a disadvantage but creates a much smaller state space which facilitates activity learning. Since this thesis builds on prior work where QTC has been used to analyse HRSI, this approach was adopted and refined to a state where it can be used for online classification and behaviour generation. This shows that even the simplified version of QTC is well suited to achieve human-aware navigation. Using the full version of the underlying QTC variants, however, might improve behaviour generation as it encodes more information about how this interaction should unfold but makes it less versatile and transferable. This trade-off is omnipresent in all QSR based research but has been solved in this work to a degree at which belief generation, action selection, and behaviour generation work reliably on an autonomous robot.

In order to generate behaviour, the proposed approach of combining an existing planner with a cost function that represents the human movement, follows the approach of the vast majority of the human-aware navigation approaches ever developed. The way in which this cost function is generated, however, is entirely different to any other form of HRSI behaviour generation because it does not depend on the position of the human or its interaction with the environment but their movement in relation to the robot. The robot movement itself has so far widely been disregarded by the HRSI community when it comes to human-aware navigation even though it plays a vital role, influencing the movement of the human. Since QTC describes the movement of human and robot in relation to each other, the model inherently represents this reciprocal process. The ability to generate cost functions from these high-level representations and use them in a standard reactive planning approach, represents one of the novel contributions of this work to the HRSI body of research and empowers the robot to make more informed decisions about its behaviour bridging the gap between reactive and deliberative approaches. As the experiments have shown, this behaviour is not only safer than using a standard Gaussian cost function but also creates more legible and sociable behaviour, hence increasing the experienced comfort. However, one can also see that this is not always the case and such a high-level approach might lead to behaviour that is perceived as less safe as can be seen from the last experiment where the robot came too close to the participants in a large number of trials. The Gaussian cost models would have the robot stop when it could not find a path because it was too close to the human already whereas the QTC based representation used the encoded knowledge to still be able to fulfil its task of avoiding the human even at the cost of coming too close. This presents another trade-off that had to be tackled in the presented work, task efficiency versus human-awareness. When the human was detected rather late, the robot still achieved safe and goal-directed navigation when using Velocity Costmaps but it was perceived as less human-aware whereas the stopping and waiting behaviour of the Gaussian model renders the robot utterly useless while there is a human blocking its path but was perceived as safer. For a real-world deployment, a combination of both systems could be considered or a simple emergency behaviour that just has the robot stop and wait when too close but both of these “solutions” would lead to the same problem of the robot being trapped again. Hence, a certain degree of goal-directed behaviour has to be preserved in order for the robot to fulfil its tasks. Moreover, the

behaviour of stopping and waiting might have been perceived as safer by the participants of the last experiment but has its detriments otherwise. From qualitative feedback from the STRANDS deployments in the elder care home where stop-and-wait was used, it is clear that leaving the avoidance to the human is perceived as rather rude and unfriendly behaviour, impeding the acceptance of the robot. Hence, this open question of human-awareness versus task efficiency has been the focus of many HRSI research projects, but has not been answered yet and can also not be answered in this thesis.

Using a high-level representation like QTC not only has benefits, but also comes at the detriment of "throwing away" a lot of information. This becomes apparent when looking at the classification problem that arises from removing the robot from the QTC_{BC} representation which makes it too ambiguous for any classification to succeed. By simply combining it with the QTC_C states of the human and the robot's goal this can be overcome, but using the robot's goal while encoding its intention comes at certain risks. If the goal is not in the relatively similar position for all the interactions used to create this model, then classification becomes ambiguous again. This is solved by using the topological representation and binding models to certain edges in the environment which means that more models have to be created and more training data is necessary. This, however, is countered by the fact that in the environment topological nodes are placed in every location that triggers a major change of direction for the movement of the robot. This includes corners, crossings, or the end of a corridor opening into a wider area. Due to this layout, the robot's goal is always directly in front of the robot which results in this virtual landmark being the same for all kinds of interactions, reducing the need for additional training data since the resulting model is easily transferable between all edges in the environment. To clarify, this constraint for the topological map is not the result of this work but a commonly followed approach which is exploited here.

Regarding the overall system, following from all the comments above, this work describes human-aware navigation for a mobile robot that is task efficient while at the same time creating legible, sociable, safe, and comfortable trajectories. The question if the use of QSRs is reasonable and beneficial for e.g. human-aware navigation, has therefore been answered by showing that it is able to achieve robust behaviour that is able to get the robot to its goal quicker and more efficiently using human-aware behaviour. This comes at no extra cost regarding the perceived comfort but suffers more from the fact that late hu-

man detections require the classification of the interaction type first and then initiate the correct behaviour. An approach agnostic to the actual type of interaction is much quicker to react but very limited in its actions making it less legible, sociable, and efficient.

7.2 Limitations

The general principle of a QSR is to abstract from the metric representation of the world by creating high-level qualitative descriptions that model the "essence" of an environment or interaction. During this process all the metric information is lost which on the one hand makes it a powerful tool for knowledge transfer and classification but, on the other hand, this abstraction, even though it allows to cope with changing environments and to generalise over several different encounters in a similar setting with a similar interaction, throws away a lot of information that would be required for the regeneration of these interactions or scenes. For example, the absolute speed at which the two agents circumvent each other cannot be expressed using QTC, neither can the absolute angle at which to perform the avoidance or the acceleration to determine if the robot should slow down, etc. All these, however, are crucial factors in HRSI to communicate ones intention or to generate safe and legible behaviour. By using the QTC variant that encodes relative speed and angles, this could be mitigated, but even this would only restrict the search space for the correct angle and speed as it only models who travels faster and whose angle is larger because no qualitative representation deals with absolute values. In the presented approach this has been overcome by using the DWA local planner that generates the velocities to be sent to the robot's wheels and the low cost areas described in Chapter 5 to increase the avoidance manoeuvre. In fact these areas are necessary because even the state $(-+)$ (approach and move to the right) still allows to move almost straight which would result in not very friendly behaviour. In this case, the full QTC version might be beneficial, but this would increase the state space significantly making it less generalisable and tractable. Nevertheless, using QSRs in general is a wide spread and popular principle in Artificial Intelligence and Robotics and allows for easy knowledge transfer which enables the presented system to cope with change and learn from similar encounters. Moreover, the experiment results have shown that low-level movement commands can be generated even with this very high-level information provided by QTC_H.

QTC itself was designed to represent the movement of two MPO in relation to each other. From this follows another limitation of the presented approach which is that it currently only interacts with the closest person and disregards all others which stems from the fact that only one Velocity Costmap is generated and used even though a belief is generated for all the humans in close vicinity. Hence, this approach will create trajectories that represent the interaction of the robot with the closest person while not colliding with any other person due to the DWA planner avoiding dynamic obstacles which might lead to illegible behaviour for any of the other people involved. The interaction with groups was not tackled in this work, but should be for a live deployment. There are several ways one could think of achieving this navigation. Firstly, groups of people could just be fused into a single moving point, e.g. the centre of mass or gravity, interacting with groups like the robot would interact with a single person. This would require only to change the people perception by introducing social models to identify groups and then merge them. Another idea is the generation of several Velocity Costmaps at the same time, overlaying them to restrict the sampling space even further. This, however, might have the robot stop if it encounters a large group due to the much more restricted search space and would lead to the so-called "frozen robot problem" (Trautman & Krause 2010). All of these suggestions might be viable options to overcome this limitation, but would require extensive research into the representation of group motion using QSRs that could fill another thesis. For the presented work, only the closest human is considered for interaction.

The most crucial limitation for all HRSI approaches is the perception component. As can be seen from the experiment in Chapter 5 versus Chapter 6, the observability of the human is crucial to this approach because it not only needs time to generate the belief, even though that only takes $0.3s$, but also time to initiate the avoidance. Since this thesis does not focus on the perception, but just presents a framework that has been developed to allow for any kind of interaction, a set of better detectors would have greatly increased the performance of the presented approach. However, the detectors used are state-of-the-art at the time of writing this, but should be replaced for future use which is easily possible given the described modular framework. Several mitigation strategies like filtering the tracker output based on static obstacles in the map or forbidden areas to enhance the performance of the robot in cluttered environments have been implemented, but still require manual annotation in some cases. Nevertheless, the proposed system performed reasonably well

in the conducted experiments and the approach presented was able to deal with most of the problems arising from this specific limitation.

Last but not least, as mentioned above, the trade-off between task efficient or goal-directed and human-aware movement has always to be considered when designing such systems as can be seen from the perceived safety in the experiment in Chapter 6. However, this thesis does not aim to answer the question if one is more important than the other, but the presented approach merely leaves this decision to the person teaching the robot. Following the LfD principle, the robot will perform as it was taught, assuming that the human expert provides the correct solution for the problems presented. Other approaches like learning from observation of two humans interacting would be possible and, in fact, if encoded in QTC easily transferable as can be seen from the "Bristol experiment", but using this assumes that robots are treated like humans in all regards of this interaction. Using LfD on the other hand, allows the demonstrator to teach the robot behaviours that she would deem appropriate for a robot to perform.

7.3 Conclusion

The presented work shows a novel approach of human-aware navigation using QSRs with the aim of creating a framework that allows to generalise over a wide variety of interactions, environments, and agents by abstracting from metric representations of the world and the interaction. Thereby, the resulting system becomes robust to change in the environment and allows for an easy knowledge transfer for machine learning approaches. The desired behaviour of creating legible, sociable, safe, and comfortable trajectories in the presence of humans has been achieved using an informed reactive approach that constraints the sampling space of a local planner to only allow human-aware trajectories to be generated. This behaviour generation is based on a fast and agile belief generation to classify the current interaction type, predict the current best state, and select the next best action for the robot to preform using the qualitative model in combination with a particle filter. As a final outcome, a fully autonomous human-aware navigation system for a mobile robot has been created and evaluated. In the following the objectives listed in Section 1.1 are addressed.

Robust qualitative interaction models [Obj. 1] In order to achieve robustness to change and facilitate knowledge transfer, the interactions between human and robot have been encoded in several different variants of QTC, namely QTC_{BC} in Chapter 3 which also encodes a distance threshold for the generation of comfortable behaviour and QTC_H in Chapter 6 which is not a variant of QTC per-se but a conglomerate of different states based on QTC_{BC} and QTC_C. This qualitative representation ***abstracts from the metric representation of the environment [Obj. 1.1]*** and generates a purely qualitative model that is robust to changes and, therefore, allows easy knowledge transfer. It achieves this abstraction by ***representing only the qualitative character of motions [Obj. 1.2]*** of both agents, i.e. human and robot, including changes in direction, stopping or starting to move, etc but excluding any environment features. In addition to that, it also represents the intention of the robot by encoding the interaction between the human and the robot's goal in QTC_C to on the one hand allow for unambiguous belief generation and on the other hand enable human-aware action selection that aims at fulfilling the robot's primary task. As mentioned above, QTC_{BC} also implicitly represents a distance threshold which can be based on proxemics, therefore, ***representing a relevant attribute of HRSI situations, i.e. distance [Obj. 1.3]***, but does not have to as it can almost be freely chosen or learned depending on the desired behaviour because this threshold is used to trigger avoidance by transitioning from QTC_B to QTC_C or vice versa. In order to be used for classification and reasoning for the evaluation of the suitability of the calculus, the qualitative model has been used in a probabilistic representation, namely HMMs, to classify different HRSI encounters from two data sets. This approach has proven to be robust to the discrete nature of the used data and the resulting unobserved states by using the emission layer to compensate for the illegal transitions. For online belief generation, QTC_H has been developed which ***facilitates decision making [Obj. 1.6]*** by combining QTC's inherent attribute of combining human and robot state with the ability to separate out the robot's state for interaction type classification. Hence, this model allows for belief generation without using the robot's sate while at the same time being able to learn action selection policies from demonstration by providing a clear mapping from belief to action.

Overall, this representation has shown its ***ability to generalise [Obj. 1.4]*** over a number of individuals and situations by using a pre-trained HMM from a different environment, using different sensors, and representing two humans interacting, to classify

interactions of human and robot that were observed by the robot itself in Chapter 4. Hence, acquired knowledge from previous encounters of the same or similar type can be utilised to facilitate learning and reasoning. Additionally, all presented variants of QTC, i.e. QTC_{BC} and QTC_{H}, that have been developed for this thesis are based on the original versions of the calculus which makes it a ***tractable, concise, and theoretically well-founded model [Obj. 1.5]*** which is able to be deployed on an autonomous robot.

To conclude the findings in regards to employing QSRs for the representation of and reasoning about HRSI, the experiments have shown that QTC_{BC} is well suited to represents multiple different HRSI encounters and that a probabilistic representation like HMMs are able to reliably classify them. They also indicate that QTC_{BC} itself is better suited to distinguish between certain kinds of encounters depending on the chosen distance threshold by highlighting the interaction in close vicinity. For this classification process, the HMM used compensates for unobserved transitions to mitigate the fact that discrete sensor messages produce illegal transitions according to the definition of QTC. These illegal transitions are defined using the conceptual distance of two states and the inter-symbol distance that has been developed for this thesis and is also used for the observation model of the particle filter. The experiments in Chapter 3 are conducted using external sensors and are post-processed to create the trace of the movement of the human. This short coming of using expensive external sensors and not being able to work in real-time has been mitigated in Chapter 4 where a similar experiment using the robot's sensors shows that QTC_{BC} is well suited to be used when observing the scene from a first person point of view. QTC_{H} on the other hand has been evaluated in Chapter 6 in combination with the particle filter and has shown to represent not only the movement of the human, but also the intention of the robot well enough to unambiguously classify two encounters and to be used for action selection. Moreover, it builds on QTC_{C} and QTC_{BC} and inherits all their functionality and tools.

Comfortable and task efficient behaviour generation [Obj. 2] In order to increase the acceptance of a mobile robot in populated environments the legibility, sociability, safety, and perceived comfort of the generated robot behaviour was paramount when designing the system presented in this thesis. Legibility in this case refers to the characteristics of the trajectory that allow the human interaction partner to predict the

future unfolding of the interaction and, therefore, fulfil their expectations in regards to locomotion whereas sociability refers to the adherence to social norms like avoiding to the left or right. As indicated in Chapter 6, ***legibility and sociability has been achieved encoding human judgement [Obj. 2.1]*** in the model employed to produce the robot's behaviour. To this end, the LfD principle was used where a human remote controls the robot while it interacts with a second human to determine how a robot should behave in the interaction at hand. Hence, the legibility and sociability of the generated behaviour depends entirely on the expert knowledge gathered during the learning phase. For the actual behaviour generation, the biggest problem to overcome when using a QSR like QTC of any form or shape is to compensate for the information lost during the abstraction process which has been solved by using Velocity Costmaps which have been developed for this thesis to restrict the sampling space of a reactive local planner to only allow trajectories that correspond to the desired qualitative state. Using this purely reactive approach in Chapter 5 has shown to be ***fast and flexible in regards to action selection [Obj. 2.4]*** as a reactive approach should be and also to ***produce safe behaviour [Obj. 2.2]*** which can be seen by the low number of collisions compared to the other methods used in the evaluation and by the larger distances kept. However, this approach was purely reactive and the prior knowledge used was not learned but hand-crafted expert knowledge in form of rules that define a state to action mapping. Additionally, the interaction type was assumed to be given before the start of the interaction which makes classification unnecessary but is infeasible for a deployment of the system.

To generate a truly autonomous system, a particle filter for QSRs has been developed and evaluated in Chapter 6 which is able to classify the current interaction type and predict the current best state online using incrementally updated QTC state chains. Using the system for people perception and the creation of valid QTC state chains implemented for this thesis and presented in Chapter 4, this filter is able to reliably classify different HRSI encounters while at the same time being robust to over-fitting or sudden changes in the behaviour of the human. Therefore, the overall system becomes ***fast in terms of belief generation resulting in a flexible action selection [Obj. 2.4]*** using the ***models learned from demonstration to create legible and sociable [Obj. 2.1], safe [Obj. 2.2], and hence comfortable behaviour*** as shown in the evaluation of Chapter 6. Additionally, the experiment has shown that the generated movement using

Velocity Costmaps is goal-directed and thus ***task efficient by reducing the travel time towards the robot's goal [Obj. 2.3]*** and that the used QTC_H is, therefore, well suited for the classification, state prediction, and action selection for human-aware navigation given little training data and noisy measurements.

On the other hand, the evaluation of the system has also shown that participants felt less safe when interacting with the robot as it comes too close in some of the cases. This is caused by a late detection of the human resulting in a reduced time span to execute the avoidance movement. However, the participants rated comfort and appropriateness the same as the state-of-the-art comparison algorithm while the approach presented here was able to ***produce more legible [Obj. 2.1] and task efficient trajectories [Obj. 2.3]*** that allow the robot to fulfil its primary task more efficiently while still showing human-aware behaviour.

Autonomy [Obj. 3] The created human-aware navigation framework has to be deployable to an autonomous mobile robot which comes with many different challenges regarding perception, abstraction, processing, action selection, and behaviour generation. While in Chapter 3 external sensors and post-processing was used to evaluate the qualitative description, the first step towards autonomy was made in Chapter 4 which introduces a ***perception framework for humans in the vicinity of the robot that relies only on the on-board sensors [Obj. 3.2]*** of the robot to generate tracks of their movement. These tracks are used together with the self-localisation of the robot to ***generate QTC states in different variants in real-time [Obj. 3.1]*** while ensuring that these state chains are valid according to the definitions of QTC. The experiment in Chapter 4 has also shown that these QTC state chains can be used to reliably classify different HRSI encounters in post-processing. Finally, Chapter 6 presents a solution to online ***belief generation that is able to classify the interaction type, predict the current state, and select the next best action in real-time [Obj. 3.1] only using on-board processing [Obj. 3.2]***.

The behaviour generated by this overall system, thanks to the DWA planner used in conjunction with the Velocity Costmaps, is based on the robots acceleration limits, and maximum speed which, therefore, allows the ***generation of legible and safe behaviour that is tailored to the used hardware [Obj. 3.3]***. This is especially important to

guarantee the safety of the human interaction partner by enabling the robot to stop before colliding with the human in case of error or missing detections which makes the produced behaviour inherently safe even if it is not always perceived as such as can be seen from Chapters 5 and 6.

As a result of the autonomy objective, the developed system is able to ***produce informed human-aware navigation behaviour for a mobile robot in real-time [Obj. 3.1] using only it's on-board sensors and processing [Obj. 3.2]***. Of course, this entirely depends on the robot platform used, but in the presented case, *Linda* the robot hosts the same sensors as the vast majority of current research and commercially available robots and no specialised processing hardware like GPUs was used.

As a final outcome, this work has presented a way of using QSRs for HRSI in the context of human-aware navigation. The experiments show that this novel method of encoding HRSI and behaviour generation is able to bridge the gap between the speed and flexibility of purely reactive systems with the legibility and commitment of deliberative systems while not succumbing to costly and inflexible exhaustive planning. Additionally, the system has an inherent and scalable ability to deal with uncertainty by using an observation model that is either based on the principles of the used QSR or learned data. This has shown reliable belief and behaviour generation despite relatively little training data and suboptimal performance of the people perception. To address the human's experience, the LfD principle is used to be able to express a wide variety of possible interactions and to encode expert knowledge and human judgement into the the created model which makes it legible, sociable, and thereby comfortable to use depending on the encoded behaviour. Despite the shortcomings listed in the limitations above, the presented system is, conceptually and software wise, deployable to an autonomous mobile robot in real-world environments.

7.4 Future Work

In order to address some of the limitations listed, especially the ones stemming from the use of the simplified version of QTC or from using a qualitative representation in general, the full version of QTC or a combination of different QSRs could be used. As mentioned

in Chapter 5, the angle λ that determines the free space in the Velocity Costmap could be adjusted to restrict the sampling space even further given there was information about this angle like in full QTC_C and the speed of the robot could also be restricted further by using ϱ based on the relative speed in full QTC_B. This, however, will increase the state space significantly from 81 to 305 possible states which makes the representation less general and hinders knowledge transfer. This trade-off could be investigated in a follow up project trying to increase the performance of the presented system by trying to encode more information in the qualitative states used that would be beneficial for behaviour generation, e.g. speed and acceleration. Finding this trade-off between information encoded and tractability, however, is no easy task and could fill another PhD thesis.

One important aspect of human-aware navigation that was not investigated is the behaviour around groups. While the particle filter already creates beliefs for each human in the vicinity of the robot, only the Velocity Costmap of the closest person is used. As described earlier, several Velocity Costmaps could be overlayed to restrict the sampling space further based on the interaction between the robot and several people. The resulting frozen robot problem in dense environments could be mitigated by weighting the costs in the map based on the distance to the human meaning that lethal costs for a far away human would not be lethal any more and would therefore allow the planner to violate the desired QTC state in favour of performing its task until it comes close to this person. All this, however, would require extensive evaluation and maybe even an adjustment of the underlying qualitative model to be better suitable for group interactions.

HRSI is a vast field encompassing interactions like guiding, following, approaching, or accomplishing a joint task like carrying a large object between human and robot. This work focuses on the sub-field of human-aware navigation due to the requirements of the STRANDS project, but it would be interesting to investigate how it behaves in other scenarios. In principle the learned model determines which behaviour is executed so there is no reason why this approach would not work for, e.g. following a human along a corridor, but since the intention of the robot is not known a-prior, i.e. it's goal is decided by the human while travelling, a different action encoding would have to be used and investigated. In principle, however, the presented system should be able to express a wide variety of interactions.

Immediate future work will investigate different ways of knowledge acquisition, i.e. dialogue. Motivated by a different European Project, this system or parts of it will be used to influence the robot's navigation based on information provided from natural language processing for a robot deployed in a shopping mall. It will be investigated if QTC models could be derived from verbal information and if these could be used to classify and predict the interaction.

Finally, the presented system has not been deployed in a real-world application yet, but is in a state where it could be. For the final deployment at the end of the STRANDS project it will be integrated into the overall navigation approach to ensure safe and legible navigation in the elder care home and/or the security setting.

Acronyms

AMCL Adaptive Monte-Carlo Localisation. 62, 88, 90, 107, 138, 140

BVI Bristol Vision Institute. 64

CND Conditional Neighbourhood Diagram. VII, 45, 47–49, 52, 53, 80

DoF Degrees of Freedom. 17

DWA Dynamic Window Approach. 31, 110–118, 120, 126, 130–133, 136, 138, 153, 157, 159, 162, 169, 174, 175, 180, 202, 206

FoV Field of View. 16–18, 99, 105, 106

GBP Great Britain Pound. 65, 158

HCI Human-Computer Interaction. 36

HMM Hidden Markov Model. VII, VIII, 11, 31, 33, 35, 55–58, 70, 71, 80, 84, 85, 87, 91, 99, 102–105, 107, 108, 135, 139–141, 148, 167, 169, 177, 178, 200–203, 205

HRI Human-Robot Interaction. 1, 16, 17, 25, 26, 37, 89, 95, 162, 168

HRSI Human-Robot Spatial Interaction. VII, 2–8, 10, 11, 14, 15, 21, 22, 24–26, 28, 29, 31–36, 39–42, 48, 49, 52, 55, 57, 59, 60, 67, 70, 78, 80–87, 89, 91, 105, 113, 132, 135, 137, 162, 164, 167, 168, 170–175, 177–182, 199–201

L-CAS Lincoln Centre for Autonomous Systems Research. VIII, 15, 124, 127, 155

LED Light Emitting Diode. 17

LfD Learning from Demonstration. 6, 12, 29, 36, 38, 112, 133, 136, 153, 154, 163, 168, 176, 179, 181

MM Markov Model. 135, 139

MPO Moving Point Objects. VII, 41–43, 45, 47, 51–53, 84, 171, 175

NN Nearest Neighbour. 98, 107

NNJPDA Nearest Neighbour Joint Probabilistic Data Association. 98, 102, 107

PTU Pan-Tilt Unit. 17, 18, 95

QSR Qualitative Spatial Relations. 4–6, 10–12, 34, 35, 39, 41, 51, 92, 108, 109, 112, 135, 136, 138–140, 164, 167, 170, 171, 173–176, 178, 179, 181, 202, 203, 205

QTC Qualitative Trajectory Calculus. VII, VIII, XI, 10–13, 34, 35, 41–43, 45–47, 49, 52–57, 59–61, 64, 66–68, 70, 74, 78–80, 84–93, 96, 99, 101–109, 111, 112, 117, 118, 120, 121, 124, 125, 127, 130, 132–138, 140, 141, 145, 149–156, 159, 165, 168, 169, 171–183, 200–203, 205

QTC$_B$ Qualitative Trajectory Calculus – Basic. VII, 10, 41, 43, 44, 47–50, 52, 54–57, 59, 67, 68, 74, 78–81, 83–86, 120, 122, 123, 125, 156, 177, 182, 200

QTC$_C$ Qualitative Trajectory Calculus – Double-Cross. VII, VIII, 10, 41, 44, 46–50, 52, 55–59, 67–71, 73, 74, 76–86, 89, 102, 103, 105, 122, 123, 125, 135, 145–147, 150, 165, 168, 173, 177, 178, 182, 200

QTC$_{BC}$ Qualitative Trajectory Calculus – Basic/Double-Cross. VII, 49–52, 55–57, 59, 60, 66–69, 71, 74, 77–79, 81–86, 102, 109, 113, 118, 122, 126, 131, 132, 135, 146–148, 150, 151, 156, 165, 167, 168, 173, 177, 178, 200, 201

QTC$_H$ Qualitative Trajectory Calculus – Human. 151, 152, 165, 168, 174, 177, 178, 180

RCC Region Connection Calculus. 34, 51, 52

ROI Region of Interest. 90, 93, 106

ROS Robot Operating System. VIII, 13, 19, 20, 80, 90–93, 95–99, 102, 107, 108, 110, 113, 119, 120, 132, 156, 169, 201, 202, 205

SIR Sampling Importance Resampling. 96, 141

SLAM Simultaneous Localisation And Mapping. VIII, 120, 132, 154

STRANDS Spatio-Temporal Representation and Activities for Cognitive Control in Long-Term Scenarios. 14, 16, 20, 89, 105, 110, 124, 132, 149, 173, 182, 183, 203, 205

UUID Universally Unique Identifier. 98, 99, 102

Bibliography

Aiello, J. R. & Aiello, T. D. C. (1974), 'The development of personal space: Proxemic behavior of children 6 through 16', *Human ecology* **2**(3), 177–189.

Alessandra, T. (2000), *Charisma: seven keys to developing the magnetism that leads to success*, Business Plus.

Althaus, P., Ishiguro, H., Kanda, T., Miyashita, T. & Christensen, H. I. (2004), Navigation for human-robot interaction tasks, *in* '2004 IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04.', Vol. 2, IEEE, pp. 1894–1900.

Arechavaleta, G., Laumond, J.-P., Hicheur, H. & Berthoz, A. (2008), 'On the nonholonomic nature of human locomotion', *Autonomous Robots* **25**(1-2), 25–35.

Argall, B. D., Chernova, S., Veloso, M. & Browning, B. (2009), 'A survey of robot learning from demonstration', *Robotics and Autonomous Systems* **57**(5), 469 – 483.
URL: *http://www.sciencedirect.com/science/article/pii/S0921889008001772*

Arras, K. O., Grzonka, S., Luber, M. & Burgard, W. (2008), Efficient people tracking in laser range data using a multi-hypothesis leg-tracker with adaptive occlusion probabilities, *in* 'IEEE International Conference on Robotics and Automation, 2008. ICRA 2008.', IEEE, pp. 1710–1715.

Arras, K. O., Mozos, O. M. & Burgard, W. (2007), Using boosted features for the detection of people in 2d range data, *in* 'IEEE International Conference on Robotics and Automation, 2007. ICRA 2007.', IEEE, pp. 3402–3407.

Arras, K. O., Tomatis, N. & Siegwart, R. (2003), Robox, a remarkable mobile robot for the real world, *in* 'Experimental Robotics VIII', Springer, pp. 178–187.

Arulampalam, M. S., Maskell, S., Gordon, N. & Clapp, T. (2002), 'A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking', *IEEE Transactions on Signal Processing* **50**(2), 174–188.

Avrunin, E. & Simmons, R. (2014), Socially-appropriate approach paths using human data, *in* 'The 23rd IEEE International Symposium on Robot and Human Interactive Communication, 2014 RO-MAN', IEEE, pp. 1037–1042.

Bar-Shalom, Y. & Li, X.-R. (1995), 'Multitarget-multisensor tracking: principles and techniques', *Storrs, CT: University of Connecticut* .

Bellotto, N. (2012), Robot control based on qualitative representation of human trajectories, *in* 'AAAI Spring Symposium – Designing Intelligent Robots: Reintegrating AI', Stanford, CA, USA. TR SS-12-02.

Bellotto, N., Dondrup, C. & Hanheide, M. (2015), 'bayestracking: The bayes tracking library v1.0.5'.
URL: *http://dx.doi.org/10.5281/zenodo.15825*

Bellotto, N., Hanheide, M. & Van de Weghe, N. (2013), Qualitative design and implementation of human-robot spatial interactions, *in* 'Proceedings of International Conference on Social Robotics, ICSR 2013'.

Bellotto, N. & Hu, H. (2010*a*), 'A bank of unscented kalman filters for multimodal human perception with mobile service robots', *International Journal of Social Robotics* **2**(2), 121–136.

Bellotto, N. & Hu, H. (2010*b*), 'Computationally efficient solutions for tracking people with a mobile robot: an experimental evaluation of bayesian filters', *Autonomous Robots* **28**(4), 425–438.

Bennewitz, M. (2004), 'Mobile robot navigation in dynamic environments', *PhD thesis, University of Freiburg* .

Bennewitz, M., Burgard, W., Cielniak, G. & Thrun, S. (2005), 'Learning motion patterns of people for compliant robot motion', *The International Journal of Robotics Research* **24**(1), 31–48.

Beyer, L., Hermans, A. & Leibe, B. (2016), 'Drow: Real-time deep learning based wheelchair detection in 2d range data', *arXiv preprint arXiv:1603.02636* .

Broekens, J., Heerink, M. & Rosendal, H. (2009), 'Assistive social robots in elderly care: a review', *Gerontechnology* **8**(2), 94–103.

Burgard, W., Cremers, A. B., Fox, D., Hähnel, D., Lakemeyer, G., Schulz, D., Steiner, W. & Thrun, S. (1999), 'Experiences with an interactive museum tour-guide robot', *Artificial intelligence* **114**(1), 3–55.

Buss, M., Carton, D., Gonsior, B., Kuehnlenz, K., Landsiedel, C., Mitsou, N., de Nijs, R., Zlotowski, J., Sosnowski, S., Strasser, E. et al. (2011), Towards proactive human-robot interaction in human environments, *in* 'Cognitive Infocommunications (CogInfoCom), 2011 2nd International Conference on', IEEE, pp. 1–6.

Butler, J. T. & Agah, A. (2001), 'Psychological effects of behavior patterns of a mobile personal robot', *Autonomous Robots* **10**(2), 185–202.

Carton, D., Turnwald, A., Kühnlenz, K., Wollherr, D. & Buss, M. (2012), Proactive human approach in dynamic environments, *in* '2012 IEEE/RSJ International Conference on Intelligent Robots and Systems', IEEE, pp. 3320–3321.

Carton, D., Turnwald, A., Wollherr, D. & Buss, M. (2013), Proactively approaching pedestrians with an autonomous mobile robot in urban environments, *in* 'Experimental Robotics', Springer, pp. 199–214.

Chang, C.-W., Lee, J.-H., Chao, P.-Y., Wang, C.-Y., Chen, G.-D. et al. (2010), 'Exploring the possibility of using humanoid robots as instructional tools for teaching a second language in primary school.', *Educational Technology & Society* **13**(2), 13–24.

Chung, S.-Y. & Huang, H.-P. (2010), A mobile robot that understands pedestrian spatial behaviors, *in* 'IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2010', IEEE, pp. 5861–5866.

Chung, W., Kim, S., Choi, M., Choi, J., Kim, H., Moon, C.-b. & Song, J.-B. (2009), 'Safe navigation of a mobile robot considering visibility of environment', *IEEE Transactions on Industrial Electronics* **56**(10), 3941–3950.

Clementini, E., Di Felice, P. & Hernández, D. (1997), 'Qualitative representation of positional information', *Artificial intelligence* **95**(2), 317–356.

Clodic, A., Fleury, S., Alami, R., Chatila, R., Bailly, G., Brethes, L., Cottret, M., Danes, P., Dollat, X., Elisei, F. et al. (2006), Rackham: An interactive robot-guide, *in* 'The 15th IEEE International Symposium on Robot and Human Interactive Communication, 2006. ROMAN 2006.', IEEE, pp. 502–509.

Cohn, A. G. & Renz, J. (2008), Chapter 13 Qualitative Spatial Representation and Reasoning, *in* F. van Harmelen, V. Lifschitz & B. Porter, eds, 'Handbook of Knowledge Representation', Vol. 3, Elsevier, pp. 551–596.

Coppola, C., Martinez Mozos, O., Bellotto, N. et al. (2015), 'Applying a 3d qualitative trajectory calculus to human action recognition using depth cameras'.

Dalal, N. & Triggs, B. (2005), Histograms of oriented gradients for human detection, *in* 'IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005. CVPR 2005.', Vol. 1, IEEE, pp. 886–893.

Dautenhahn, K., Walters, M., Woods, S., Koay, K. L., Nehaniv, C. L., Sisbot, A., Alami, R. & Siméon, T. (2006), How may i serve you?: a robot companion approaching a seated person in a helping context, *in* 'Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction', ACM, pp. 172–179.

De Boor, C. (1978), 'A practical guide to splines'.

Delafontaine, M. (2011), Modelling and Analysing Moving Objects and Travelling Subjects: Bridging theory and practice, PhD thesis, Department of Geography, Ghent University.

Diego, G. & Arras, K. O. (2011), Please do not disturb! minimum interference coverage for social robots, *in* 'IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2011', IEEE, pp. 1968–1973.

Dondrup, C., Bellotto, N. & Hanheide, M. (2014*a*), A probabilistic model of human-robot spatial interaction using a qualitative trajectory calculus, *in* '2014 AAAI Spring Symposium Series'.

Dondrup, C., Bellotto, N. & Hanheide, M. (2014*b*), Social distance augmented qualitative trajectory calculus for human-robot spatial interaction, *in* 'The 23rd IEEE International Symposium on Robot and Human Interactive Communication, 2014 RO-MAN', pp. 519–524.

Dondrup, C., Bellotto, N., Hanheide, M., Eder, K. & Leonards, U. (2015), 'A computational model of human-robot spatial interactions based on a qualitative trajectory calculus', *Robotics* **4**(1), 63–102.
URL: *http://www.mdpi.com/2218-6581/4/1/63*

Dondrup, C., Bellotto, N., Jovan, F. & Hanheide, M. (2015), Real-time multisensor people tracking for human-robot spatial interaction, *in* 'Workshop on Machine Learning for Social Robotics', ICRA/IEEE.

Dondrup, C. & Hanheide, M. (2016), Qualitative constraints for human-aware robot navigation using velocity costmaps, *in* 'IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)'.

Dondrup, C., Lichtenthäler, C. & Hanheide, M. (2014), Hesitation signals in human-robot head-on encounters: a pilot study, *in* 'Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction', ACM, pp. 154–155.

Duckworth, P., Gatsoulis, Y., Jovan, F., Hawes, N., Hogg, D. C. & Cohn, A. G. (2016), Unsupervised learning of qualitative motion behaviours by a mobile robot, *in* 'Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems', International Foundation for Autonomous Agents and Multiagent Systems, pp. 1043–1051.

Ducourant, T., Vieilledent, S., Kerlirzin, Y. & Berthoz, A. (2005), 'Timing and distance characteristics of interpersonal coordination during locomotion', *Neuroscience Letters* **389**(1), 6–11.

Faria, D. R., Premebida, C. & Nunes, U. (2014), A probabilistic approach for human everyday activities recognition using body motion from rgb-d images, *in* 'The 23rd IEEE International Symposium on Robot and Human Interactive Communication, 2014 RO-MAN', IEEE, pp. 732–737.

Faria, D. R., Vieira, M., Premebida, C. & Nunes, U. (2015), Probabilistic human daily activity recognition towards robot-assisted living, *in* '24th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), 2015', IEEE, pp. 582–587.

Feil-Seifer, D. & Mataric, M. (2011), People-aware navigation for goal-oriented behavior involving a human partner, *in* 'IEEE International Conference on Development and Learning (ICDL), 2011', Vol. 2, IEEE, pp. 1–6.

Feurtey, F. (2000), 'Simulating the collision avoidance behavior of pedestrians', *Master's Thesis* .

Feyrer, S. & Zell, A. (2000), Robust real-time pursuit of persons with a mobile robot using multisensor fusion, *in* '6th International Conference on Intelligent Autonomous Systems (IAS-6)', pp. 710–715.

Fink, G. A. (2008), *Markov Models for Pattern Recognition*, Springer-Verlag Berlin Heidelberg.

Fod, A., Howard, A. & Mataric, M. J. (2002), A laser-based people tracker, *in* 'IEEE International Conference on Robotics and Automation, 2002. Proceedings. ICRA'02.', Vol. 3, IEEE, pp. 3024–3029.

Fong, T., Nourbakhsh, I. & Dautenhahn, K. (2003), 'A survey of socially interactive robots', *Robotics and Autonomous Systems* **42**(3-4), 143–166.

Förster, F., Weiss, A. & Tscheligi, M. (2011), Anthropomorphic design for an interactive urban robot: the right design approach, *in* 'Proceedings of the 6th international conference on Human-robot interaction', ACM, pp. 137–138.

Fox, D., Burgard, W. & Thrun, S. (1997), 'The dynamic window approach to collision avoidance', *IEEE Robotics & Automation Magazine* **4**(1), 23–33.

Fraichard, T. (2007), A short paper about motion safety, *in* 'International Conference on Robotics and Automation, 2007 IEEE', IEEE, pp. 1140–1145.

Garrido, J. & Yu, W. (2014), Trajectory generation in joint space using modified hidden markov model, *in* 'The 23rd IEEE International Symposium on Robot and Human Interactive Communication, 2014 RO-MAN', IEEE, pp. 429–434.

Gavrila, D. M. (1999), 'The visual analysis of human movement: A survey', *Computer vision and image understanding* **73**(1), 82–98.

Gerling, K., Hebesberger, D., Dondrup, C., Körtner, T. & Hanheide, M. (2016), 'Robotereinsatz in der Langzeitpflege – Fallstudie zum Einsatz eines mobilen Roboters zur Unterstützung von Physiotherapie', *Zeitschrift fr Gerontologie und Geriatrie* . in press, article is written in English.

Grisetti, G., Stachniss, C. & Burgard, W. (2005), Improving grid-based slam with rao-blackwellized particle filters by adaptive proposals and selective resampling, *in* 'Proceedings of the 2005 IEEE International Conference on Robotics and Automation, 2005. ICRA 2005.', IEEE, pp. 2432–2437.

Grisetti, G., Stachniss, C. & Burgard, W. (2007), 'Improved techniques for grid mapping with rao-blackwellized particle filters', *Robotics, IEEE Transactions on* **23**(1), 34–46.

Gross, H.-M., Boehme, H., Schroeter, C., Müller, S., Koenig, A., Einhorn, E., Martin, C., Merten, M. & Bley, A. (2009), Toomas: interactive shopping guide robots in everyday use-final implementation and experiences from long-term field trials, *in* 'IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009. IROS 2009.', IEEE, pp. 2005–2012.

Gross, H.-M., Scheidig, A., Debes, K., Einhorn, E., Eisenbach, M., Mueller, S., Schmiedel, T., Trinh, T. Q., Weinrich, C., Wengefeld, T. et al. (2016), 'Roreas: robot coach for walking and orientation training in clinical post-stroke rehabilitationprototype implementation and evaluation in field trials', *Autonomous Robots* pp. 1–20.

Hall, E. T. (1969), *The hidden dimension*, Anchor Books New York.

Hanheide, M., Peters, A. & Bellotto, N. (2012), Analysis of human-robot spatial behaviour applying a qualitative trajectory calculus, *in* 'RO-MAN, 2012 IEEE', IEEE, pp. 689–694.

Hawes, N., Burbridge, C., Jovan, F., Kunze, L., Lacerda, B., Mudrová, L., Young, J., Wyatt, J., Hebesberger, D., Körtner, T. et al. (2016), 'The strands project: Long-term autonomy in everyday environments', *arXiv preprint arXiv:1604.04384* .

Hebesberger, D., Dondrup, C., Körtner, T., Gisinger, C. & Pripfl, J. (2016), Lessons learned from the deployment of a long-term autonomous robot as companion in physical therapy for older adults with dementia - A Mixed Methods Study, *in* '11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)"'.

Helbing, D. (1991), 'A mathematical model for the behavior of pedestrians', *Behavioral science* **36**(4), 298–310.

Hoeller, F., Schulz, D., Moors, M. & Schneider, F. E. (2007), Accompanying persons with a mobile robot using motion prediction and probabilistic roadmaps, *in* 'IEEE/RSJ International Conference on Intelligent Robots and Systems, 2007. IROS 2007.', IEEE, pp. 1260–1265.

Huang, K.-C., Li, J.-Y. & Fu, L.-C. (2010), Human-oriented navigation for service providing in home environment, *in* 'Proceedings of SICE Annual Conference 2010', IEEE, pp. 1892–1897.

Iliopoulos, K., Bellotto, N., Mavridis, N. et al. (2014), From sequence to trajectory and vice versa: solving the inverse qtc problem and coping with real-world trajectories, AAAI.

Jaffari, O. H., Mitzel, D. & Leibe, B. (2014), Real-Time RGB-D based People Detection and Tracking for Mobile Robots and Head-Worn Cameras, *in* 'IEEE International Conference on Robotics and Automation (ICRA'14)'.

Jensen, B., Tomatis, N., Mayor, L., Drygajlo, A. & Siegwart, R. (2005), 'Robots meet humans-interaction in public spaces', *Industrial Electronics, IEEE Transactions on* **52**(6), 1530–1546.

Johansson, M., Skantze, G. & Gustafson, J. (2011), 'Understanding route directions in human-robot dialogue'.

Ke, S.-R., Thuc, H. L. U., Lee, Y.-J., Hwang, J.-N., Yoo, J.-H. & Choi, K.-H. (2013), 'A review on video-based human activity recognition', *Computers* **2**(2), 88–131.

Kelley, J. F. (1984), 'An iterative design methodology for user-friendly natural language office information applications', *ACM Transactions on Information Systems (TOIS)* **2**(1), 26–41.

Kessler, J., Schroeter, C. & Gross, H.-M. (2011), Approaching a person in a socially acceptable manner using a fast marching planner, *in* 'Intelligent Robotics and Applications', Springer, pp. 368–377.

Kirby, R., Simmons, R. & Forlizzi, J. (2009), 'COMPANION: A Constraint-Optimizing Method for Person-Acceptable Navigation', *RO-MAN 2009 - The 18th IEEE International Symposium on Robot and Human Interactive Communication* pp. 607–612.

Kluge, B. & Prassler, E. (2004), Reflective navigation: Individual behaviors and group behaviors, *in* 'IEEE international conference on robotics and automation', pp. 4172–4177.

Koay, K. L., Sisbot, E. A., Syrdal, D. S., Walters, M. L., Dautenhahn, K. & Alami, R. (2007), Exploratory study of a robot approaching a person in the context of handing over an object., *in* 'AAAI spring symposium: multidisciplinary collaboration for socially assistive robotics', pp. 18–24.

Kollmitz, M., Hsiao, K., Gaa, J. & Burgard, W. (2015), Time dependent planning on a layered social cost map for human-aware robot navigation, *in* '2015 European Conference on Mobile Robots (ECMR)', IEEE, pp. 1–6.

Krajnik, T., Fentanes, J. P., Cielniak, G., Dondrup, C. & Duckett, T. (2014), Spectral analysis for long-term robotic mapping, *in* 'IEEE International Conference on Robotics and Automation (ICRA), 2014', IEEE, pp. 3706–3711.

Krajník, T., Fentanes, J. P., Hanheide, M. & Duckett, T. (2016), Persistent localization and life-long mapping in changing environments using the frequency map enhancement, *in* 'Proceedings of International Conference on Intelligent Robots and Systems'. in review; First and second authors share first authorship.

Kruse, T., Basili, P., Glasauer, S. & Kirsch, A. (2012), Legible robot navigation in the proximity of moving humans, *in* 'IEEE Workshop on Advanced Robotics and its Social Impacts (ARSO), 2012', IEEE, pp. 83–88.

Kruse, T., Kirsch, A., Sisbot, E. A. & Alami, R. (2010), Dynamic generation and execution of human aware navigation plans, *in* 'Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1', International Foundation for Autonomous Agents and Multiagent Systems, pp. 1523–1524.

Kruse, T., Pandey, A. K., Alami, R. & Kirsch, A. (2013), 'Human-aware robot navigation: A survey', *Robotics and Autonomous Systems* **61**(12), 1726–1743.
URL: *http://www.sciencedirect.com/science/article/pii/S0921889013001048*

Kuipers, B. & Byun, Y.-T. (1991), 'A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations', *Robotics and autonomous systems* **8**(1), 47–63.

Kushleyev, A. & Likhachev, M. (2009), Time-bounded lattice for efficient planning in dynamic environments, *in* 'IEEE International Conference on Robotics and Automation, 2009. ICRA'09.', IEEE, pp. 1662–1668.

Lam, C.-P., Chou, C.-T., Chiang, K.-H. & Fu, L.-C. (2011), 'Human-centered robot navigationtowards a harmoniously human–robot coexisting environment', *IEEE Transactions on Robotics* **27**(1), 99–112.

Li, X. R. & Jilkov, V. P. (2000), A survey of maneuvering target tracking: Dynamic models, *in* 'Proc. of SPIE Conf. on Signal and Data Processing of Small Targets', FL, USA, pp. 212–235.

Lichtenthäler, C. & Kirsch, A. (2016), 'Legibility of robot behavior: A literature review'.

Lichtenthäler, C., Lorenz, T. & Kirsch, A. (2012), Influence of legibility on perceived safety in a virtual human-robot path crossing task, *in* 'RO-MAN, 2012 IEEE', IEEE, pp. 676–681.

Lichtenthäler, C., Peters, A., Griffiths, S. & Kirsch, A. (2013), 'Social navigation-identifying robot navigation patterns in a path crossing scenario', *ICSR* .

Lightbody, P., Dondrup, C. & Hanheide, M. (2015), 'Make me a Sandwich ! Intrinsic Human Identification from their Course of Action', *ICSR* pp. 1–4.

Linder, T., Breuers, S., Leibe, B. & Arras, K. O. (2016), 'On multi-modal people tracking from mobile platforms in very crowded and dynamic environments', *IEEE International Conference on Robotics and Automation (ICRA)* .

Linder, T., Girrbach, F., Arras, K. O., Linder, T., Wehner, S., Arras, K. O., Palmieri, L., Arras, K. O., Okal, B., Gilbert, H. et al. (2015), Towards a robust people tracking framework for service robots in crowded, dynamic environments, *in* 'Assistance and Service Robotics Workshop (ASROB-15) at the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS) 2015', Vol. 70, Springer Berlin Heidelberg, pp. 557–572.

Liu, J. & Daneshmend, L. K. (2012), *Spatial reasoning and planning: geometry, mechanism, and motion*, Springer Science & Business Media.

Lu, D. V., Allan, D. B. & Smart, W. D. (2013), Tuning cost functions for social navigation, *in* 'Social Robotics', Springer, pp. 442–451.

Lu, D. V., Hershberger, D. & Smart, W. D. (2014), Layered costmaps for context-sensitive navigation, *in* 'IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014), 2014', IEEE, pp. 709–715.

Luber, M., Spinello, L., Silva, J. & Arras, K. O. (2012), Socially-aware robot navigation: A learning approach, *in* 'IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2012', IEEE, pp. 902–907.

Macaluso, I., Ardizzone, E., Chella, A., Cossentino, M., Gentile, A., Gradino, R., Infantino, I., Liotta, M., Rizzo, R. & Scardino, G. (2005), Experiences with cicerobot, a museum guide cognitive robot, *in* 'AI* IA 2005: Advances in Artificial Intelligence', Springer, pp. 474–482.

Martinez-Garcia, E. A., Akihisa, O. & Yuta, S. (2005), Crowding and guiding groups of humans by teams of mobile robots, *in* 'IEEE Workshop on Advanced Robotics and its Social Impacts, 2005.', IEEE, pp. 91–96.

May, A. D., Dondrup, C. & Hanheide, M. (2015), Show me your moves! Conveying navigation intention of a mobile robot to humans, *in* 'European Conference on Mobile Robots (ECMR), 2015', IEEE, pp. 1–6.

Mead, R. & Mataric, M. J. (2015), 'Robots have needs too: People adapt their proxemic preferences to improve autonomous robot recognition of human social signals', *New Frontiers in Human-Robot Interaction* p. 100.

Meena, R., Skantze, G. & Gustafson, J. (2012*a*), A chunking parser for semantic interpretation of spoken route directions in human-robot dialogue, *in* 'The 4th Swedish Language Technology Conference (SLTC 2012), Lund, October 24-26, 2012', pp. 55–56.

Meena, R., Skantze, G. & Gustafson, J. (2012*b*), A data-driven approach to understanding spoken route directions in human-robot dialogue., *in* 'INTERSPEECH', pp. 226–229.

Mitzel, D. & Leibe, B. (2012), Close-range human detection for head-mounted cameras, *in* 'British Machine Vision Conference (BMVC)'.

Morales, Y., Watanabe, A., Ferreri, F., Even, J., Ikeda, T., Shinozawa, K., Miyashita, T. & Hagita, N. (2015), Including human factors for planning comfortable paths, *in* 'IEEE International Conference on Robotics and Automation (ICRA), 2015', IEEE, pp. 6153–6159.

Mori, M. (1970), Bukimi no tani [the uncanny valley]. energy, 7 (4) 33-35.(translated by karl f. macdorman and takashi minato in 2005) within appendix b for the paper androids as an experimental apparatus: Why is there an uncanny and can we exploit it?, *in* 'Proceedings of the CogSci-2005 Workshop: Toward Social Mechanisms of Android Science', pp. 106–118.

Mumm, J. & Mutlu, B. (2011), Human-robot proxemics: physical and psychological distancing in human-robot interaction, *in* 'International Conference on Human-Robot Interaction', Lausanne, Switzerland.

Nourbakhsh, I. R., Kunz, C. & Willeke, T. (2003), The mobot museum robot installations: A five year experiment, *in* 'IEEE/RSJ International Conference on Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003', Vol. 4, IEEE, pp. 3636–3641.

Ohki, T., Nagatani, K. & Yoshida, K. (2010), Collision avoidance method for mobile robot considering motion and personal spaces of evacuees, *in* 'IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2010', IEEE, pp. 1819–1824.

Oliver, N., Horvitz, E. & Garg, A. (2002), Layered representations for human activity recognition, *in* 'Fourth IEEE International Conference on Multimodal Interfaces, 2002. Proceedings.', IEEE, pp. 3–8.

Pacchierotti, E., Christensen, H. I. & Jensfelt, P. (2005), Human-robot embodied interaction in hallway settings: a pilot user study, *in* 'IEEE International Workshop on Robot and Human Interactive Communication, 2005. ROMAN 2005.', pp. 164–171.

Pacchierotti, E., Christensen, H. I. & Jensfelt, P. (2006), Evaluation of passing distance for social robots, *in* 'The 15th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN 2006', pp. 315–320.

Pandey, A. K. & Alami, R. (2010), A framework towards a socially aware mobile robot motion in human-centered dynamic environment, *in* 'IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2010', IEEE, pp. 5855–5860.

Peters, A. (2011), Small movements as communicational cues in HRI, *in* T. Kollar & A. Weiss, eds, 'HRI 2011 - Workshop on Human-Robot Interaction Pioneers', pp. 72–73.

Peters, A., Spexard, T. P., Hanheide, M. & Weiss, P. (2011), Hey robot, get out of my way - A survey on a spatial and situational movement concept in HRI, *in* 'BMI - Well-Being', pp. 147–165.

Piyathilaka, L. & Kodagoda, S. (2015), Human activity recognition for domestic robots, *in* 'Field and Service Robotics', Springer, pp. 395–408.

Poppe, R. (2010), 'A survey on vision-based human action recognition', *Image and vision computing* **28**(6), 976–990.

Pulido Fentanes, J., Lacerda, B., Krajník, T., Hawes, N. & Hanheide, M. (2015), Now or later? predicting and maximising success of navigation actions from long-term experience, *in* 'International Conference on Robotics and Automation (ICRA)'.

Randell, D. A., Cui, Z. & Cohn, A. G. (1992), 'A spatial logic based on regions and connection.', *KR* **92**, 165–176.

Riek, L. D. (2012), 'Wizard of oz studies in hri: a systematic review and new reporting guidelines', *Journal of Human-Robot Interaction* **1**(1).

Rios-Martinez, J. (2013), Socially-aware robot navigation: combining risk assessment and social conventions, PhD thesis, PhD Thesis, University of Grenoble.

Rios-Martinez, J., Renzaglia, A., Spalanzani, A., Martinelli, A. & Laugier, C. (2012), Navigating between people: A stochastic optimization approach, *in* 'IEEE International Conference on Robotics and Automation (ICRA), 2012', IEEE, pp. 2880–2885.

Robins, B., Dautenhahn, K., Te Boekhorst, R. & Billard, A. (2005), 'Robotic assistants in therapy and education of children with autism: can a small humanoid robot help encourage social interaction skills?', *Universal Access in the Information Society* **4**(2), 105–120.

Robot Operating System wiki: Costmap 2D (n.d.), http://wiki.ros.org/costmap_2d. Accessed: 2016-04-28.

Santos, J. M., Krajnik, T., Pulido Fentanes, J. & Duckett, T. (2016), 'Lifelong information-driven exploration to complete and refine 4d spatio-temporal maps'.

Scandolo, L. & Fraichard, T. (2011), An anthropomorphic navigation scheme for dynamic scenarios, *in* 'IEEE International Conference on Robotics and Automation (ICRA), 2011', IEEE, pp. 809–814.

Schapire, R. E. & Singer, Y. (1999), 'Improved boosting algorithms using confidence-rated predictions', *Machine learning* **37**(3), 297–336.

Schiffer, S. (2015), 'Integrating qualitative reasoning and human-robot interaction for domestic service robots'.

Schulz, D., Burgard, W., Fox, D. & Cremers, A. B. (2003), 'People tracking with mobile robots using sample-based joint probabilistic data association filters', *The International Journal of Robotics Research* **22**(2), 99–116.

Simmons, R. (1996), The curvature-velocity method for local obstacle avoidance, *in* 'IEEE International Conference on Robotics and Automation, ICRA', Vol. 4, IEEE, Minneapolis, MN, pp. 3375–3382.

Sisbot, A. (2008), Towards human-aware robot motions, PhD thesis, Université Paul Sabatier-Toulouse III.

Sisbot, E., Marin-Urias, L., Alami, R. & Simeon, T. (2007), 'A Human Aware Mobile Robot Motion Planner', *IEEE Transactions on Robotics* **23**(5), 874–883.

Spinello, L. & Arras, K. O. (2011), People detection in rgb-d data, *in* 'IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2011', IEEE, pp. 3838–3843.

Steinfeld, A., Fong, T., Kaber, D., Lewis, M., Scholtz, J., Schultz, A. C. & Goodrich, M. (2006), 'Common metrics for human-robot interaction', *HRI, 2006* p. 33.

Steinfeld, A., Jenkins, O. C. & Scassellati, B. (2009), The oz of wizard: simulating the human for interaction research, *in* 'HRI, 2009', IEEE, pp. 101–107.

Sudowe, P. & Leibe, B. (2011), Efficient use of geometric constraints for sliding-window object detection in video, *in* 'Computer Vision Systems', Springer, pp. 11–20.

Sung, J., Ponce, C., Selman, B. & Saxena, A. (2011), 'Human activity detection from rgbd images.', *plan, activity, and intent recognition* **64**.

Sung, J., Ponce, C., Selman, B. & Saxena, A. (2012), Unstructured human activity detection from rgbd images, *in* 'IEEE International Conference on Robotics and Automation (ICRA), 2012', IEEE, pp. 842–849.

Svenstrup, M., Bak, T. & Andersen, H. J. (2010), Trajectory planning for robots in dynamic human environments, *in* 'IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2010', IEEE, pp. 4293–4298.

Tadokoro, S., Hayashi, M., Manabe, Y., Nakami, Y. & Takamori, T. (1995), On motion planning of mobile robots which coexist and cooperate with human, *in* 'IROS, 1995. 'Human Robot Interaction and Cooperative Robots'.', Vol. 2, IEEE, pp. 518–523.

Takayama, L. & Pantofaru, C. (2009), Influences on proxemic behaviors in human-robot interaction, *in* '2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)', IEEE, pp. 5495–5502.

Tamura, Y., Dai Le, P., Hitomi, K., Chandrasiri, N. P., Bando, T., Yamashita, A. & Asama, H. (2012), Development of pedestrian behavior model taking account of intention, *in* 'IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2012', IEEE, pp. 382–387.

Thrun, S., Beetz, M., Bennewitz, M., Burgard, W., Cremers, A. B., Dellaert, F., Fox, D., Haehnel, D., Rosenberg, C., Roy, N. et al. (2000), 'Probabilistic algorithms and the interactive museum tour-guide robot minerva', *The International Journal of Robotics Research* **19**(11), 972–999.

Thrun, S., Burgard, W. & Fox, D. (2005), *Probabilistic robotics*, MIT press.

Torta, E., Cuijpers, R. H., Juola, J. F. & van der Pol, D. (2011), Design of robust robotic proxemic behaviour, *in* 'Social Robotics', Springer, pp. 21–30.

Tranberg Hansen, S., Svenstrup, M., Andersen, H. J. & Bak, T. (2009), Adaptive human aware navigation based on motion pattern analysis, *in* 'The 18th IEEE International Symposium on Robot and Human Interactive Communication, 2009. RO-MAN 2009.', IEEE, pp. 927–932.

Trautman, P. & Krause, A. (2010), Unfreezing the robot: Navigation in dense, interacting crowds, *in* 'IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2010', IEEE, pp. 797–803.

Trautman, P., Ma, J., Murray, R. M. & Krause, A. (2013), Robot navigation in dense human crowds: the case for cooperation, *in* 'IEEE International Conference on Robotics and Automation (ICRA), 2013', IEEE, pp. 2153–2160.

Van de Weghe, N. (2004), Representing and Reasoning about Moving Objects: A Qualitative Approach, PhD thesis, Ghent University.

Van den Berg, J., Lin, M. & Manocha, D. (2008), Reciprocal velocity obstacles for real-time multi-agent navigation, *in* 'IEEE International Conference on Robotics and Automation, 2008. ICRA 2008.', IEEE, pp. 1928–1935.

Vollmer, A.-L., Lohan, K. S., Fischer, K., Nagai, Y., Pitsch, K., Fritsch, J., Rohlfing, K. J. & Wredek, B. (2009), People modify their tutoring behavior in robot-directed interaction for action learning, *in* 'IEEE 8th International Conference on Development and Learning, 2009. ICDL 2009.', IEEE, pp. 1–6.

Wagner, T. & Hübner, K. (2004), An egocentric qualitative spatial knowledge representation based on ordering information for physical robot navigation, *in* 'RoboCup 2004: Robot Soccer World Cup VIII', Springer, pp. 134–149.

Walia, G. S. & Kapoor, R. (2014), 'Human detection in video and images - a state-of-the-art survey', *International Journal of Pattern Recognition and Artificial Intelligence* **28**(03).

Walters, M. L., Oskoei, M. A., Syrdal, D. S. & Dautenhahn, K. (2011), A Long-Term Human-Robot Proxemic Study, *in* 'RO-MAN 2011 - The 20th IEEE International Symposium on Robot and Human Interactive Communication', pp. 137–142.

Wang, K., Wang, X., Lin, L., Wang, M. & Zuo, W. (2014), 3d human activity recognition with reconfigurable convolutional neural networks, *in* 'Proceedings of the ACM International Conference on Multimedia', ACM, pp. 97–106.

Weiss, A., Bernhaupt, R., Tscheligi, M., Wollherr, D., Kuhnlenz, K. & Buss, M. (2008), A methodological variation for acceptance evaluation of human-robot interaction in public places, *in* 'The 17th IEEE International Symposium on Robot and Human Interactive Communication, 2008. RO-MAN 2008.', IEEE, pp. 713–718.

Weiss, A., Mirnig, N., Buchner, R., Förster, F. & Tscheligi, M. (2011), Transferring human-human interaction studies to hri scenarios in public space, *in* 'IFIP Conference on Human-Computer Interaction', Springer, pp. 230–247.

Weiss, A., Mirnig, N. & Förster, F. (2011), What users expect of a proactive navigation robot, *in* 'Proceedings of the workshop Expectations in intuitive interaction on the 6th HRI International conference on Human-Robot Interaction'.

Wojek, C., Nickel, K. & Stiefelhagen, R. (2006), Activity recognition and room-level tracking in an office environment, *in* 'IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, 2006', IEEE, pp. 25–30.

Wojek, C., Walk, S., Roth, S. & Schiele, B. (2011), Monocular 3d scene understanding with explicit occlusion reasoning, *in* 'IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011', IEEE, pp. 1993–2000.

Yuan, F., Twardon, L. & Hanheide, M. (2010), Dynamic path planning adopting human navigation strategies for a domestic mobile robot, *in* 'IROS, 2010.', IEEE, pp. 3275–3281.

Ziebart, B. D., Ratliff, N., Gallagher, G., Mertz, C., Peterson, K., Bagnell, J. A., Hebert, M., Dey, A. K. & Srinivasa, S. (2009), Planning-based prediction for pedestrians, *in* 'IROS, 2009.', IEEE, pp. 3931–3936.

A Publication List

The following is a comprehensive list of scientific publications that this thesis is based on or are cited throughout the paper. The main contribution of each paper is briefly described and the contribution of the author of this thesis is detailed to highlight the original work undertake by *Christian Dondrup*. Only publications relevant to the presented work are listed.

A.1 Probabilistic Qualitative Models for HRSI

Dondrup, Lichtenthäler & Hanheide (2014)

> **Dondrup, C., Lichtenthäler, C. & Hanheide, M., Hesitation signals in human-robot head-on encounters: a pilot study, *in* 'Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction', (2014), ACM, pp. 154–155.**

Dondrup, Lichtenthäler & Hanheide's (2014) describes an expirment to find so-called hesitation signals in HRSI. The hesitation signals are a sudden decrease in velocity of the human participant when confronted with inconsistent robot behaviour. This feedback could then be used as negative reward for possible reinforcement learning approaches. The experiment conducted was a hypothetical restaurant scenario where human and robot played th role of waiters in a confined shared space with the robot showing two different behaviours, i.e. stop-and-wait to let the human pass and ignoring the human entirely. Using this set-up hesitation signals could be found but there was no significant difference between the two conditions.

Author contributions: *Christian Dondrup* contributed the implementation of the two different behaviours, conducted the user study, and wrote the paper. *Christina Lichtenthäler* evaluated the data of the user study and crated the graphics. *Marc Hanheide* helped conceiving the idea and the experiment design, and proofread the paper.

Dondrup, Bellotto & Hanheide (2014*a*)

Dondrup, C., Bellotto, N. & Hanheide, M., A probabilistic model of human-robot spatial interaction using a qualitative trajectory calculus, *in* '2014 AAAI Spring Symposium Series', 2014.

Dondrup, Bellotto & Hanheide's (2014*a*) work describes the first approach of using Qualitative Trajectory Calculus – Double-Cross (QTC_C) in combination with a Hidden Markov Model (HMM) to represent HRSI. The HMM is used to create valid state chains even though the discrete nature of the input data produced by the robot's sensors will result in illegal QTC state chains because it was designed for continuous space.

Author contributions: *Christian Dondrup* contributed the implementation of the Qualitative Trajectory Calculus – Double-Cross (QTC_C) in matlab and as a python library, the implementation of the Hidden Markov Model (HMM) in python, the data collection, the evaluation, and conducted the experiment and wrote the paper. *Nicola Bellotto* helped with the definition of QTC_C and proofread the paper. *Marc Hanheide* helped conceiving the idea and implementing QTC_C in matlab, helped creating the images, and proof read the paper.

Dondrup, Bellotto & Hanheide (2014*b*)

Dondrup, C., Bellotto, N. & Hanheide, M., Social distance augmented qualitative trajectory calculus for human-robot spatial interaction, *in* 'Robot and Human Interactive Communication, 2014 RO-MAN: The 23rd IEEE International Symposium on Robot and Human Interactive Communication', pp. 519–524, 2014.

Dondrup, Bellotto & Hanheide's (2014*b*) work describes the first attempt of modelling a distance threshold in QTC by combing the Qualitative Trajectory Calculus – Basic (QTC_B) and the Qualitative Trajectory Calculus – Double-Cross (QTC_C) into the Qualitative Trajectory Calculus – Basic/Double-Cross (QTC_{BC}) using the HMM based approach described by Dondrup, Bellotto & Hanheide (2014*a*).

Author contributions: *Christian Dondrup* contributed the implementation of all the QTC variants used in matlab and as a python library, the implementation of the Hidden Markov Model (HMM) in python, the data collection, the evaluation, and conducted the experiment and wrote the paper. *Nicola Bellotto* helped with the definition of QTC and proofread the paper. *Marc Hanheide* helped conceiving the idea, implementing QTC_{BC} in matlab, and proofread the paper.

Dondrup, Bellotto, Hanheide, Eder & Leonards (2015)

Dondrup, C., Bellotto, N., Hanheide, M., Eder, K. & Leonards, U., 'A computational model of human-robot spatial interactions based on a qualitative trajectory calculus', *Robotics* 4(1), 63–102, 2015.

Dondrup, Bellotto, Hanheide, Eder & Leonards's (2015) work describes the entire effort on using QTC for HRSI in one concise paper and evaluates its appropriateness for the use

in human-aware navigation by showing that it can be used to reliably classify different HRSI encounter from two separate and distinct experiments.

Author contributions: *Christian Dondrup* contributed the implementation of all the QTC variants used in matlab and as a python library, the implementation of the Hidden Markov Model (HMM) in python, the evaluation of the experiment data, conducted the restaurant experiment, and wrote the paper. *Nicola Bellotto* helped with the definition of QTC and proofread the paper. *Kersting Eder* and *Ute Leonards* conducted the "Bristol experiment", prepared the raw data, and proofread the paper. *Marc Hanheide* helped conceiving the idea, implementing QTC_{BC} in matlab, and proofread the paper.

A.2 People Perception and QTC State Generation

Dondrup, Bellotto, Jovan & Hanheide (2015)

> **Dondrup, C., Bellotto, N., Jovan, F. & Hanheide, M., Real-time multisensor people tracking for human-robot spatial interaction, *in* 'Workshop on Machine Learning for Social Robotics', ICRA/IEEE, (2015)**

Dondrup, Bellotto, Jovan & Hanheide (2015) describe A modular and freely available implementation for a multi-sensor Bayesian tracking framework in the Robot Operating System (ROS). This frame work is able to produce people tracks in real-time at a frequency of $30Hz$ and is able to produce incrementally updated QTC state chains for every of the tracked persons.

Author contributions: *Christian Dondrup* contributed the implementation of the ROS wrapper around the Bayesian tracking library, ported the upper body detector into ROS, implemented the online QTC generation in the so-called *QSR Lib*, conducted the experiment and evaluated the results, and wrote the paper. *Nicola Bellotto* contributed the original implementation of the tracker and helped with its description. *Ferdian Jovan* contributed the trajectory stitching component. *Marc Hanheide* helped conceiving the idea and proofread the paper.

A.3 Constraint based HRSI Behaviour Generation

Dondrup & Hanheide (2016)

This paper has been published at the IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN 2016).

> **Dondrup, C. & Hanheide, M., Qualitative constraints for human-aware robot navigation using velocity costmaps, *in* 'IEEE International Symposium on Robot and Human Interactive Communication', RO-MAN, (2016).**

Dondrup & Hanheide (2016) describe a novel approach to generate movement commands for a mobile robot from Qualitative Spatial Relations (QSR). As a specific application, the paper introduces Velocity Costmaps based on the Qualitative Trajectory Calculus (QTC) for human-aware navigation. These Velocity Costmaps are used as a so-called critique function in the widely used and popular ROS implementation of the Dynamic Window Approach (DWA) local planner for dynamic obstacle avoidance. The approach is evaluated in simulation and as a proof of concept experiment on a non-holonomic mobile robot.

Author contributions: *Christian Dondrup* contributed the implementation of the people perception pipeline, the online QTC generation, Velocity Costmap generation, the Velocity Costmap critique function in the DWA planner, conducted the experiment and evaluated the results, and wrote the paper. *Marc Hanheide* helped conceiving the idea and proofread the paper.

A.4 Other Publications

The papers in this section did not directly contribute to the thesis but had significant involvement of the author and are, therefore, listed for the sake of completeness.

May, Dondrup & Hanheide (2015)

> **May, A. D., Dondrup, C. & Hanheide, M., Show me your moves! Conveying navigation intention of a mobile robot to humans, *in* '2015 European Conference on Mobile Robots (ECMR)', IEEE, pp. 1–6, (2015).**

May, Dondrup & Hanheide's (2015) work evaluates the use of head movement, i.e. joint attention, versus visual light indicators, adopting known principle of the automotive industry, to convey navigational intent. Ina comprehensive study, they found that participants preferred the visual light indicators over head movement and felt more comfortable in this condition.

Author contributions: *Alyxander May* conducted the experiment, implemented the different behaviours, evaluated the data, and wrote parts of the paper. *Christian Dondrup* helped with the implementation of the visual light indicators and the head movement, helped with the evaluation of the data, and wrote parts of the paper. *Marc Hanheide* helped conceiving the idea, and wrote parts of the paper.

Lightbody, Dondrup & Hanheide (2015)

> **Lightbody, P., Dondrup, C. & Hanheide, M., 'Make me a Sandwich! Intrinsic Human Identification from their Course of Action', *in* ICSR, pp. 1–4, 2015.**

Lightbody, Dondrup & Hanheide's (2015) work uses Qualitative Spatial Relations (QSR) to classify interactants in a table top assembly task based on their course of actions. This work does not make use of QTC but uses the developed HMM framework from

Chapter 3 which shows that it generalises to any kind of of QSR and not only QTC and therefore proves its general applicability.

Author contributions: *Peter Lightbody* Implemented the detection, extended the tracking framework of Dondrup, Bellotto, Jovan & Hanheide (2015) to use 3D data, conducted the study and evaluated the data, and wrote parts of the paper. *Christian Dondrup* contributed the implementation of the Hidden Markov Model (HMM) in python, and the 2D tracking framework, helped with the evaluation, and wrote parts of the paper. *Marc Hanheide* help conceiving the idea and implementing the software components, and wrote parts of the paper.

Hebesberger, Dondrup, Körtner, Gisinger & Pripfl (2016)

Hebesberger, D., Dondrup, C., Körtner, T., Gisinger, C. & Pripfl, J., Lessons learned from the deployment of a long-term autonomous robot as companion in physical therapy for older adults with dementia - A Mixed Methods Study, *in* '11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)"', (2016).

Hebesberger, Dondrup, Körtner, Gisinger & Pripfl's (2016) work investigates the use of o mobile robot as therapeutic aid for physical exercise of Dementia patients. In these so-called walking groups, the robot provides entertainment during phases where the participants have to rest to distract them and help the more restless patients to focus. During the walking phases the robot accompanies the group, playing old German hiking songs for them to sing along and dance or sway. Evaluation shows that if the robot works reliably, it is a great aid to therapists and improves the mood of the group significantly.

Author contributions: *Denise Hebesberger* annotated the walking groups, evaluated the data, and wrote the paper. *Christian Dondrup* provided the implementation of the walking group and remotely supervised each group to take care of possible errors that might occur during, and wrote parts of the paper. *Tobias Körtner* helped with the data evaluation. *Christoph Gisinger* and *Jürgen Pripfl* are the heads of the research group.

Gerling, Hebesberger, Dondrup, Körtner & Hanheide (2016)

Gerling, K., Hebesberger, D., Dondrup, C., Körtner, T. & Hanheide, M., 'Robotereinsatz in der Langzeitpflege – Fallstudie zum Einsatz eines mobilen Roboters zur Unterstützung von Physiotherapie', *Zeitschrift fr Gerontologie und Geriatrie*, (2016). To appear.

Gerling, Hebesberger, Dondrup, Körtner & Hanheide's (2016) describe the deployment of a mobile robot, i.e. *Henry*, in an elder care home in Austria and describes the applications that the STRANDS project has implemented for this scenario and evaluates their possible impact on an ageing society.

Author contributions: *Katrin Gerling* and *Denise Hebesberger* jointly wrote the paper. *Christian Dondrup* was to varying degrees involved in the technical realisation of all

of the tasks executed at the care home and took care of the robot remotely during the entirety of the deployments, and provided some of the images. *Tobias Körtner* was involved in all the deployments of the robot. *Marc Hanheide* was responsible for the organisation of the deployments.

B
Web Resources

All the mentioned ROS implementations are freely available and open source. Please find a list of source code repositories and other useful links below. Additionally, videos and a concise overview of the published systems can be found at `http://www.dondrup.net` and instructions on how to configure your system to use the STRANDS repositories and package servers for automated install of the system can be found at `http://lncn.eu/strands`.

QSR Lib this library contains a collection of many different QSRs and also includes the implementation of all the mentioned QTC variants provided by the author:
`http://github.com/strands-project/strands_qsr_lib/tree/master/qsr_lib`

QSR Learning All the learning approaches, i.e. HMMs and the Particle Filter, which can be used with any kind of QSR not only QTC have been implemented by the author and can be found at:
`http://github.com/strands-project/strands_qsr_lib/tree/master/qsr_prob_rep`

People Perception The perception pipeline described in this work, has been implemented into ROS by the author and can be found at:
`http://github.com/strands-project/strands_perception_people`

Automatic QTC state generation This module generates QTC states using the input of above pipeline and validates them automatically. It has been implemented by the author and can be found at:
`http://github.com/strands-project/strands_hri/tree/hydro-devel/hrsi_representation`

State Prediction Using the state chains generated by above module and the particle filter from the QSR learning library, are used in this module to generate the current belief of the world. It also uses the learned conditional probability table to find the next best action for the robot. This has been implemented by the author and can

be found at:
http://github.com/strands-project/strands_hri/tree/hydro-devel/hrsi_state_prediction

Velocity Costmap Server Using the next best robot action, this module generates the Velocity Costmap which is sent to the DWA local planner. This has been implemented by the author and can be found at:
http://github.com/strands-project/strands_hri/tree/hydro-devel/hrsi_velocity_costmaps

DWA Local Planner The updated version of the DWA planner using Velocity Costmaps has been modified by the author and can be found at:
http://github.com/strands-project/navigation/tree/indigo-devel/dwa_local_planner

C
Questionnaires

C.1 Behaviour Generation Questionnaire

Edit this form

Between-run QA

State your agreement with the following statements in the situation you just encountered.

*Required

Subject ID *

Situation *

- Pass-by
- Pass-crossing

I felt safe when I encountered the robot in this situation. *

	1	2	3	4	5	
fully DISagree						fully agree

I felt comfortable when I encountered the robot in this situation. *

	1	2	3	4	5	
fully DISagree						fully agree

I was able to follow my intended path with no disruption by the robot in this situation. *

	1	2	3	4	5	
fully DISagree						fully agree

I had to slow down to let the robot pass *

	1	2	3	4	5	
Slowed down a lot and waited						Had not to slow down significantly

The robot behaved appropriately in this situation. *

	1	2	3	4	5	
fully DISagree						fully agree

Please identify any problem you saw in this situation that made you feel less comfortable

and safe

- [ ] Robot made physical contact with me
- [ ] Robot was too fast
- [ ] Robot was too slow
- [ ] Robot came to close too me
- [ ] I was not sure where the robot was going
- [ ] The robot did not see me or ignored me
- [ ] Other:

Any comment (by experimenter)

Submit

Never submit passwords through Google Forms.