

**Personenwahrnehmung für eine
sozialverträgliche und nutzerzentrierte Roboternavigation
in öffentlichen Einsatzumgebungen**

Dissertation
zur Erlangung des akademischen Grades
Doktor-Ingenieur (Dr.-Ing.)

vorgelegt der
Fakultät für Informatik und Automatisierung
der Technischen Universität Ilmenau

von

Dipl.-Inf. Christoph Weinrich
geboren am 08.01.1984 in Heilbad Heiligenstadt

Tag der Einreichung: 24.06.2015
Tag der Verteidigung: 28.01.2016

Gutachter

- 1.) Prof. Dr. Horst-Michael Groß
- 2.) Prof. Dr. Maren Bennewitz
- 3.) Prof. Dr. Andreas Zell

Kurzfassung

Insbesondere bei mobilen Servicerobotern, welche in öffentlichen Einrichtungen mit nicht eingewiesenen Personen interagieren, hat die Navigation auch eine sozio-emotionale Bedeutung und stellt eine Form der nonverbalen Kommunikation mit den anwesenden Menschen dar. Zur Optimierung der Mensch-Roboter-Interaktion ist es deshalb erforderlich, dass diese Roboter ein sozialverträgliches (d. h. verständliches, angenehmes und nicht störendes) Navigationsverhalten gegenüber ihren Nutzern und unbeteiligten Personen aufweisen.

Zur Steigerung der sozialen Verträglichkeit mobiler Serviceroboter werden in dieser Arbeit vier Navigationsverhalten entwickelt, welche im Kontext der Anwendungsszenarien Gebäudelotse und Rehabilitationsroboter besonders relevant sind: 1. die *gezielte Kontaktaufnahme* des Roboters mit potenziellen Nutzern, 2. das *personenfokussierte Lotsen* von Nutzern unter Berücksichtigung ihrer Bewegungsgeschwindigkeit, 3. das nutzerzentrierte *Begleiten von Personen* und 4. die *respektvolle Navigation* unter Berücksichtigung der Proxemik, insbesondere von unbeteiligten Personen.

Die Umsetzung dieser Verhaltensweisen erfordert neben der Implementierung der eigentlichen Navigationsalgorithmen auch die Anpassung und Weiterentwicklung verschiedener technischer Funktionalitäten zur Informationsgewinnung über die anwesenden Personen. So werden im Kontext der laserbasierten Personendetektion generische Merkmale vorgestellt, mittels welcher Personen auch dann detektiert werden können, wenn sie Gehhilfen wie Rollstühle oder Rollatoren verwenden. Zur kamerabasierten Schätzung der Oberkörperorientierung wird ein Personendetektor präsentiert, welcher mittels eines Entscheidungsbaumes mit Support Vector Machines als binäre Entscheider acht Orientierungsklassen effizient unterscheidet. Außerdem wird ein erscheinungsbasiertes Verfahren zur Schätzung der Gelenkstellungen des menschlichen Oberkörpers und zur Wiedererkennung von Personen beschrieben. Darüber hinaus wird gezeigt, wie mittels eines on-line lernenden Verfahrens zur Prädiktion der Personenbewegung eine vorausschauende Umsetzung der bearbeiteten Verhaltensweisen realisiert wird.

Neben der separaten Evaluation der technischen Funktionalitäten zur Informationsgewinnung werden die Ergebnisse einer experimentellen Untersuchung der Gesamtlösung bzgl. der Verhaltensweisen präsentiert. Unter Nutzung eines externen Trackingsystems wird gezeigt, dass der Roboter tatsächlich das gewünschte Navigationsverhalten aufweist und dass sich dieses positiv auf die Bewegung der menschlichen Interaktionspartner auswirkt.

Abstract

Especially for mobile service robots, which operate in public environments with uninstructed people, navigation has a socio-emotional importance. Thereby, the robot's motion constitutes an interaction with the present people, that resembles a non-verbal communication. To optimize the human-robot interaction, these robots should show socially compliant navigation behavior towards its users and bystanders.

To improve the social acceptance of mobile service robots this work focuses on the development of four navigation behaviors. These behaviors are particularly relevant for the application scenarios of this work, which are a guidance robot and a rehabilitation robot: 1. selective contact initiation with (potential) users, 2. person-focused guiding considering the users speed of movement, 3. the user-centered accompanying of people, and 4. the respectful navigation considering the proxemics of users and bystanders.

The realization of these navigation behaviors requires the implementation of the actual navigation algorithms. Furthermore, the adaptation and improvement of features for information retrieval about the present people is of fundamental importance. Accordingly, this work presents features for laser-based person detection, which enable to detect people regardless of their use of walking aids like walkers or wheelchairs. For estimation of the people's upper-body orientation a camera-based detector is introduced. It utilizes a binary decision tree with Support Vector Machines for binary decision making, to differentiate eight upper-body classes. Moreover, an appearance-based approach for estimation of the joint angles of the human upper-body and for people re-identification is investigated. Additionally, an on-line learning approach is developed to predict the peoples motion in proximity to robots. This allows a proactive realization of the behaviors.

Besides the separate evaluation of the aforementioned features for information retrieval, the complete robot system is evaluated with respect to the previously mentioned behaviors. By the use of an external tracking system it is demonstrated, that the robot actually implements the desired behavior. The analysis of the human trajectories shows, that the realized behaviors have a measurable, positive impact on the motion of human interaction partners.

Danksagung

Mein herzlicher Dank gilt all denen, die mich auf unterschiedliche Weise während der Erstellung dieser Dissertation unterstützt haben.

Zuerst möchte ich Prof. Dr. Horst-Michael Groß danken, für die Möglichkeit an einem interessanten Thema zu arbeiten. Sie haben mir einerseits viele Freiräume eingeräumt und waren andererseits zur Stelle um die Richtung zu weisen.

Besonders schätze ich auch die tolle Atmosphäre, welche ich während meiner Zeit am Fachgebiet Neuroinformatik und Kognitive Robotik erleben durfte. Ich bedanke mich bei meinen Kollegen für die anregenden Diskussionen und die gegenseitige Unterstützung, aufgrund welcher mir die tägliche Arbeit sehr viel Freude gemacht hat: Klaus Debes, Konrad Schenk, Sandra Helsper, Cornelia Dittmar, Thomas Schmiedel, Jens Keßler, Markus Eisenbach, Erik Schaffernicht, Alexander König, Hans-Joachim Böhme, Sven Hellbach, Andrea Scheidig, Robert Kaltenhäuser, Heike Groß, Sabine Schulz, Anja Zwetkow-Schilling, Katja Hamatschek und Ute Schütz.

Hervorheben möchte ich die Kollegen, mit welchen ich besonders eng zusammen gearbeitet habe und mit denen sich auch über die Arbeit hinaus Freundschaften entwickelt haben: Erik Einhorn, Christof Schröter, Michael Volkhardt, Steffen Müller, Ronny Stricker und Christian Vollmer.

Weiterhin möchte ich auch denen danken, die bereits während ihres Studiums durch Abschlussarbeiten zu dieser Arbeit beigetragen haben: Benedikt Seidl, Jonathan Liebner, Thanh Quang Trinh, Christian Henke, Thomas Golda, Tim Wengefeld und Alexander Kolarow.

Zuletzt gilt mein Dank denen, die jederzeit für mich da sind und mir dadurch die Kraft für diese Arbeit gegeben haben: meine Familie, meine Freunde und ganz besonders Annika.

Anmerkungen

Diese Dissertation ist in deutscher Sprache verfasst. Da aus der Fachliteratur jedoch meist englische Fachbegriffe bekannt sind, werden zur besseren Verständlichkeit häufig Lehnwörter aus dem Englischen verwendet. Sowohl Großschreibung als auch Deklination dieser Lehnwörter erfolgt entsprechend der deutschen Orthografie und Grammatik.

Zur Erleichterung des Leseflusses wird in dieser Arbeit häufig der männliche Singular verwendet. So beispielsweise bei dem Wort „Nutzer“. Sofern keine explizite Unterscheidung gemacht wird, beziehen sich die Aussagen sowohl auf männliche als auch auf weibliche Personen.

Inhaltsverzeichnis

0.1. Verwendete Notation	1
1. Einleitung	3
1.1. Interaktion mit sozialen Assistenzrobotern	3
1.2. Einordnung der Arbeit	4
1.2.1. Sozialverträgliche Navigation	4
1.2.2. Vorschau auf die Anwendungsszenarien	5
1.2.3. Fokus und Anspruch der Arbeit	6
1.2.4. Einordnung bezüglich anderer Arbeiten am Fachgebiet	6
1.3. Aufbau dieser Arbeit und zu erwartende Beiträge	8
1.4. Publikationen	10
2. Anforderungsanalyse und Systemüberblick	13
2.1. Anwendungsszenarien	13
2.2. Benötigte sozialverträgliche Verhaltensweisen	15
2.3. Technische Funktionalitäten	18
2.4. Rahmenbedingungen und Experimentierplattform	19
2.5. Soziale Assistenzroboter in öffentlichen Einsatzumgebungen	21
2.6. Gesamtsystem und Steuerungsarchitektur	23
2.7. Fazit	26
3. Sozialverträgliche Verhaltensweisen	27
3.1. Die drei Ebenen der Bewegungsplanung	27
3.1.1. Reaktiv: Dynamic Window Approach (DWA)	29
3.1.2. Taktisch: Pfadplanung mit dem E*-Algorithmus	33
3.1.3. Strategisch: Festlegung von Teilaufgaben und Zwischenzielen	34
3.2. Gezielte Kontaktaufnahme	34
3.3. Respektvolle Navigation	36
3.3.1. Reaktives Ausweichen	37
3.3.2. Taktisches Ausweichen stehender Personen	37

3.3.3. Reaktives Einhalten des Rechtsverkehrs	39
3.4. Personenfokussiertes Lotsen	41
3.5. Begleiten einer Person	43
3.5.1. Gleichmäßige Fahrweise durch Double Buffering	44
3.5.2. Bewegungsprädiktion zum vorausschauenden Begleiten	44
3.6. Zusammenfassung	46
4. Personentracking mit Zusatzinformationen	47
4.1. Personendetektion und Unterscheidung von Gehhilfen	47
4.1.1. Verfahren zur Personendetektion in Laserdaten	48
4.1.2. Struktur des laserbasierten Personendetektors	51
4.1.3. Segmentierung	52
4.1.4. Merkmalsextraktion	52
4.1.5. Klassifikation	56
4.1.6. Hypothesengenerierung	58
4.1.7. Experimente	58
4.1.8. Zusammenfassung und Ausblick	65
4.2. Personendetektion mit Orientierungsschätzung	65
4.2.1. Forschungsstand	67
4.2.2. Struktur des kamerabasierten Personendetektors	72
4.2.3. Generierung von Trainingsdaten	72
4.2.4. Mehrklassenunterscheidung mittels SVM Tree	74
4.2.5. Experimente	77
4.2.6. Zusammenfassung und Ausblick	81
4.3. Tracking der laser- und bildbasierten Hypothesen	82
4.4. Zusammenfassung	85
5. Erscheinungsbasierte Oberkörpermodellierung	87
5.1. Bildbasierte Schätzung von Gelenkstellungen	88
5.2. Appearance Model zur Hypothesenoptimierung	91
5.3. Experimente	98
5.4. Kritische Bewertung der Ergebnisse	101
6. Prädiktion menschlichen Ausweichverhaltens	103
6.1. Prädiktion von Personenbewegungen	104
6.2. On-line Lernen menschlicher Bewegungsmuster	106
6.3. Experimente	108
6.4. Zusammenfassung und Ausblick	116

7. Evaluation der Navigationsverhalten	117
7.1. Evaluationskriterien	118
7.1.1. Bewertungsmetriken für die Mensch-Roboter-Interaktion	118
7.1.2. Verwendete Metriken	120
7.2. Respektvolle Navigation	122
7.2.1. Ausweichen vor Personen	122
7.2.2. Abstand zu Personen in der Rehaklinik	127
7.2.3. Rechtsfahren	128
7.2.4. Fazit zur respektvollen Navigation	132
7.3. Lotsen	132
7.4. Begleiten	136
7.5. Zusammenfassung und Ausblick	140
8. Zusammenfassung und Ausblick	143
8.1. Zusammenfassung	143
8.2. Ausblick	145
A. Soziale Assistenzroboter in öffentlichen Einsatzumgebungen	149
B. Kontrollparadigmen für Roboter	155
B.1. Populäre Kontrollparadigmen für Roboter	155
B.2. Einordnung des Paradigmas dieser Arbeit	158
C. Umsetzung der Verhaltensweisen	161
C.1. Gezielte Kontaktaufnahme	161
C.2. Reaktives Ausweichen im Kontext der respektvollen Navigation	163
C.3. Taktisches Ausweichen im Kontext der respektvollen Navigation	164
C.4. Reaktives Einhalten des Rechtsverkehrs	165
D. Laser- und kamerabasierte Personendetektion	169
D.1. Parametrierung der laserbasierten Personendetektion	169
D.2. Mehrklassenunterscheidung mittels SVMs	169
D.3. Optimierung der SVM eines Knotens des Entscheidungsbaumes	172
D.4. Evaluation des SVM Entscheidungsbaumes	175
D.5. Transformation von 3D-Hypothesen mit Unsicherheitsfortpflanzung	178
D.5.1. Euklidische Transformation im 3D	178
D.5.2. Transformationen zwischen Koordinatensystemen	179
E. Erscheinungsbasierte Oberkörpermodellierung	183

E.1. Shader des Kantenmodells	183
E.2. Shader des Farbmodells	185
E.3. Eigenschaften der Matching-Funktion	186
F. On-line Lernen menschlicher Bewegungsmuster	189
F.1. Gelernte Prädiktion zum reaktiven Ausweichen	189
F.2. Korrelation zwischen Prädiktion und tatsächlicher Bewegung	191
F.3. Momentumform und kanonische Form einer Normalverteilung	192
F.4. Produkt zweier Normalverteilungen	193
Literaturverzeichnis	199

0.1. Verwendete Notation

Allgemeine Notation

\mathbb{R}	Körper der reellen Zahlen
\mathbb{R}^n	n -dimensionaler euklidischer Raum
$\mathbf{v} \in \mathbb{R}^n$	Spaltenvektor
$\mathbf{M} \in \mathbb{R}^{n \times m}$	Matrix

Auswahl konkreter Bezeichner

$\mathbf{H} = \{\mathbf{h}_0, \mathbf{h}_1, \dots, \mathbf{h}_i, \dots\}$	Menge der getrackten Personenhypothesen
$\mathbf{h}_i = \{\mathit{pose}, \mathit{cov}, \mathit{conf}\}$	einzelne Personenhypothese als Aufenthaltswahrscheinlichkeit im \mathbb{R}^7
$\mathit{pose}(\mathbf{h}_i) = (\mathit{pos}, \mathit{ori})^T \in \mathbb{R}^7$	Mittlere Pose der Wahrscheinlichkeitsverteilung der Hypothese
$\mathit{pos}(\mathbf{h}_i) = (x, y, z)^T \in \mathbb{R}^3$	3D-Position
$\mathit{ori}(\mathbf{h}_i) = (q_r, q_x, q_y, q_z)^T \in \mathbb{R}^4$	Orientierung im 3D als Quaternion
$\mathit{cov}(\mathbf{h}_i) \in \mathbb{R}^{7 \times 7}$	Kovarianzmatrix der Wahrscheinlichkeitsverteilung über die Pose der Person
$\mathit{conf}(\mathbf{h}_i) \in \mathbb{R}$	Sicherheit der Hypothese (Amplitude der Gaußverteilung)
$\mathit{vel}(\mathbf{h}_i) \in \mathbb{R}^3$	translatorische Geschwindigkeit der Hypothese
\mathbf{h}_G	Hypothese von gelotster Person
\mathbf{h}_F	Hypothese von begleiteter Person
$\alpha_{Dis}, \alpha_{Dir}, \alpha_{Hea}, \dots$	Wichtungsfaktoren der Objectives des DWA
$\mathbf{M}_{\text{travel costs}} \in \mathbb{R}^{n \times m}$	Karte der Fahrtkosten
$\mathbf{I} \in \mathbb{R}^{n \times m \times 3}$	Kamerabild
$\boldsymbol{\theta} \in \mathbb{R}^{14}$	Parameter des ercheinungsbasierten Modells
$\mathbf{O}^{obs} \in \mathbb{R}^{n \times m}$	Gradientenorientierungen des aufgenommenen Bildes \mathbf{I}
$\mathbf{O}^{mod} \in \mathbb{R}^{n \times m}$	Modellierte Gradientenorientierungen
$\mathbf{M}^{obs} \in \mathbb{R}^{n \times m}$	Gradientenmagnituden des aufgenommenen Bildes \mathbf{I}
$\mathbf{M}^{mod} \in \mathbb{R}^{n \times m}$	Modellierte Gradientenmagnituden

1

Einleitung

1.1. Interaktion mit sozialen Assistenzrobotern

Im Zuge des technischen Fortschritts werden neue Anwendungsgebiete für Roboter erschlossen, welche immer enger mit dem alltäglichen Leben der Menschen verbunden sind. Dies zeigt sich auch an der Einführung neuer Begriffe zur Unterscheidung von Roboterarten. So werden laut [ISO 8373, 2012] Serviceroboter explizit von Industrierobotern abgegrenzt. Serviceroboter werden sowohl privat als auch professionell in der Umgebung von Menschen, z. B. zur Schwimmbadreinigung, zum Rasen mähen oder zum Staub saugen, eingesetzt. Persönliche Serviceroboter erbringen teil- oder vollautonom Dienstleistungen für Menschen, welche nicht zur Bedienung des Roboters geschult werden brauchen.

Ein Teilgebiet der Servicerobotik ist die soziale Assistenzrobotik, in welches sich diese Arbeit einordnet. Assistenzroboter liefern physikalische, mentale oder soziale Unterstützung für hilfebedürftige Personen, wie Ältere oder Behinderte [Goodrich, 2007, S. 214]. Eine besondere Herausforderungen dieser Art der Mensch-Roboter-Interaktion (Human-Robot Interaction, HRI) ist der sichere physische Kontakt mit Menschen, oder zumindest die Bewegung in geringer Entfernung zu diesen [Goodrich, 2007, S. 214]. Effektive soziale Interaktion erfordert kognitive und emotionale Informationsverarbeitung sowie natürliche Interaktion durch Gesten und Sprache. Auch wenn der Roboter in seiner Position als Helfender dem Menschen unterstellt ist, nimmt er in der praktischen Anwendung, z. B. als Begleiter, häufig eine gleichgestellte Rolle ein. Damit sich der Nutzer auf den Roboter und die angebotenen Dienste einlässt, muss zuvor eine Vertrauensbasis geschaffen werden [Goodrich, 2007, S. 235 f.].

Die Besonderheit der sozialen Assistenzrobotik gegenüber der Assistenzrobotik im Allgemeinen ist, dass der Nutzer vorrangig mittels sozialer Interaktion und ohne direkten physischen Kontakt unterstützt wird [Feil-Seifer, 2005]. Der eigentliche Zweck des Roboters ist jedoch weiterhin die Unterstützung des Nutzers. Damit unterscheidet sich die soziale Assistenzrobotik von der sozialen interaktiven Robotik, einem weiteren Teilgebiet der Servicerobotik (socially interactive robotics), bei welchem die Interaktion im Vordergrund steht [Feil-Seifer, 2005]. Ein sozialer Assistenzroboter zur Schlaganfallnachsorge [Gross, 2014] würde beispielsweise den Nutzer ohne physischen Kontakt und nur durch soziale Interaktion bei der physischen und mentalen Genesung unterstützen.

1.2. Einordnung der Arbeit

Nachdem die besondere Relevanz der Mensch-Roboter-Interaktion im Kontext der sozialen Assistenzrobotik angesprochen wurde, soll in diesem Abschnitt die Fokussierung dieser Arbeit auf die sozialverträgliche und nutzerzentrierte Navigation motiviert werden. Des Weiteren werden zwei konkrete Anwendungsszenarien dieser Arbeit kurz angerissen. Dabei handelt es sich um einen Lotsenroboter in einem Bürogebäude und um einen robotischen Laufcoach zur Rehabilitation von Schlaganfallpatienten. Danach wird beschrieben, worauf der Fokus dieser Arbeit liegt und welche Arbeiten am Fachgebiet Neuroinformatik und Kognitive Robotik (NIKR) im Zusammenhang mit dieser Arbeit stehen. Zum Abschluss der Einleitung wird die Gliederung der weiteren Arbeit vorgestellt.

1.2.1. Sozialverträgliche Navigation

Bei der Mensch-Mensch-Interaktion wird zwischen einer instrumentalen (aufgabenorientierten) und einer sozio-emotionalen (beziehungsorientierten) Interaktionsebene unterschieden [Bales, 1951]. Mit der technischen Weiterentwicklung und der Steigerung der Alltagstauglichkeit von Service- und Assistenzrobotern wächst auch das Interesse an der sozio-emotionalen Ebene der Mensch-Roboter-Interaktion (Human-Robot Interaction, HRI). Der sozio-emotionale Aspekt der HRI ist ein wesentlicher Faktor für die Akzeptanz von Servicerobotern durch die Nutzer [Sardar, 2012, Dautenhahn, 2006].

Bei mobilen Assistenzrobotern stellt auch die Navigation des Roboters eine Form der nonverbalen Interaktion mit den Menschen dar. Damit hat die Navigation nicht nur eine instrumentelle Funktion zur Erfüllung der Aufgaben des Roboters, sondern auch sozio-emotionale Bedeutung. Zur Optimierung der Mensch-Roboter-Interaktion sollten Assistenzroboter deshalb ein Navigationsverhalten aufweisen, welches sowohl für die Nutzer,

als auch für unbeteiligte anwesende Personen sozialverträglich ist. Der Roboter sollte darum so navigieren, dass es für die Menschen intuitiv verständlich ist und möglichst keine negativen Effekte wie Verunsicherung, Erschrecken, Unbehagen usw. auslöst [Satake, 2009, Hüttenrauch, 2006, Fong, 2003].

Dabei ist das Einhalten und Respektieren des persönlichen Raumes (proxemics) ein zentraler Bestandteil des sozialen Interaktionsverhaltens [Hall, 1963]. Eine Orientierung der HRI an solchen sozialen Verhaltensweisen ist wünschenswert, da so keine ungewohnten Interaktionsformen erlernt werden müssen und dies den Zugang zu dem jeweiligen Robotersystem erleichtert [Bente, 2001, Cassell, 1999]. Eine solche HRI wird durch den Menschen als vertraut und intuitiv verständlich wahrgenommen [Dautenhahn, 2004, Syrdal, 2006].

1.2.2. Vorschau auf die Anwendungsszenarien

Das Ziel dieser Arbeit ist die Entwicklung technischer Funktionalitäten zur Umsetzung sozialer Verhaltensweisen, welche in verschiedensten Szenarien angewendet werden können. Zur Untersuchung der Verhaltensweisen im realen Einsatz werden jedoch zwei konkrete Anwendungsszenarien genutzt.

Das Anwendungsszenario Gebäudelotse besteht aus zwei Lotsenrobotern, welche im Fakultätsgebäude „Zusebau“ der TU Ilmenau als Besucherinformationssystem eingesetzt werden und die Nutzer beispielsweise zu den Räumen führen, welche diese gerade suchen.

Das zweite Anwendungsszenario besteht in dem Forschungsprojekt ROREAS, in dessen Rahmen ein interaktiver robotischer Reha-Assistent für das Lauf- und Orientierungstraining von Patienten nach Schlaganfällen entwickelt wird. Auch in diesem Szenario navigiert ein mobiler Roboter in einer öffentlichen Einsatzumgebung gemeinsam mit seinem Nutzer und unbeteiligten Personen. So sollen die Patienten z. B. zu Therapieräumen geleitet werden, oder es wird aufgepasst, dass sie selbst im Sinne eines kognitiven Orientierungstrainings den richtigen Weg zu den entsprechenden Räumen wählen. Darüber hinaus werden die Patienten auch bei ihren Trainingsläufen begleitet, um ihr Laufverhalten zu protokollieren. Die Notwendigkeit des sozialverträglichen Verhaltens des Roboters ist hier besonders hoch, weil mit Gehbehinderten und kognitiv beeinträchtigten Personen interagiert wird. Beide Szenarien werden in Abschnitt 2.1 ausführlicher beschrieben.

1.2.3. Fokus und Anspruch der Arbeit

Der Fokus dieser Arbeit besteht in der Implementierung der notwendigen technischen Funktionalitäten, um die folgenden vier Verhaltensweisen umzusetzen: 1. die *gezielte Kontaktaufnahme* des Roboters mit potenziellen Nutzern, 2. das *personenfokussierte Lotsen* von Nutzern unter Berücksichtigung ihrer Bewegungsgeschwindigkeit, 3. das nutzerzentrierte *Begleiten von Personen* und 4. die *respektvolle Navigation* unter Berücksichtigung der Proxemics [Hall, 1963] insbesondere von unbeteiligten Personen. Zur Umsetzung dieser Verhaltensweisen sollen keine Veränderungen an der Einsatzumgebung vorgenommen werden, und die Umsetzung soll basierend auf der vorhandenen Roboterplattform mit deren technischen Rahmenbedingungen erfolgen. Die aufgeführten Verhaltensweisen, die dafür notwendigen technischen Funktionalitäten und die Roboterplattform sind in Kapitel 2 ausführlicher beschrieben.

1.2.4. Einordnung bezüglich anderer Arbeiten am Fachgebiet

Vorangegangene Arbeiten

Am Fachgebiet Neuroinformatik und Kognitive Robotik (NIKR) der TU Ilmenau wird seit Ende der 1990er Jahre im Rahmen verschiedener Projekte an mobilen Servicerobotern mit Lotsenfunktion in öffentlichen Einsatzumgebungen gearbeitet. Der Fokus dieser Projekte lag vor allem auf der robusten Lokalisierung, Pfadplanung und der sicheren Navigation, ohne Personen und Einsatzumgebung zu gefährden. Während der Langzeittests im Baumarkt [Gross, 2009] wurde festgestellt, dass zur weiteren Steigerung der Akzeptanz bei den Baumarktkunden verschiedene sozialverträgliche Verhaltensweisen [Pöschl, 2009, S. 28] notwendig sind. Wie oben beschrieben, muss der Roboter ohne langwierige Instruktionen intuitiv verständlich sein und sollte deshalb ein Verhalten aufweisen, welche die Nutzer von der Interaktion mit Menschen gewohnt sind. [Pöschl, 2009, S. 28] Aus dieser Erkenntnis leitete sich die Zielstellung dieser Arbeit ab, mit deren Bearbeitung im November 2009 begonnen wurde.

Parallele Arbeiten

Am Fachgebiet NIKR wurden insbesondere im Rahmen der Projekte ALIAS, CompanionAble und SERROGA weitere Forschungsarbeiten im Bereich der Personenwahrnehmung und der Navigation in öffentlichen Einsatzumgebungen bearbeitet. Diese Arbeiten werden im Folgenden kurz vorgestellt.

Um das Interaktionsinteresse der Nutzer eines Roboters zu schätzen, wird in [Martin, 2012] die Oberkörperpose bzw., bei geringem Abstand zwischen Roboter und Nutzer, die Blickrichtung geschätzt. Zusätzlich findet eine Schätzung der Emotionen des Nutzers auf Basis der Mimik statt. Diese Informationen werden dann mittels Bayes-Filter [Thrun, 2005] fusioniert und die Aufmerksamkeit bzw. das Interesse des Nutzers geschätzt. Zur Informationsgewinnung werden bildbasierte Analyse-durch-Synthese Verfahren eingesetzt. Zur Schätzung der Oberkörperpose sind dies „Active Shape Models“ (ASMs) [Cootes, 1995], zur Blickrichtungs- und Emotionsschätzung „Active Appearance Models“ (AAMs) [Cootes, 1998]. Um aus den Modellparametern der ASMs bzw. der AAMs die Oberkörperorientierung, Blickrichtung oder Emotion abzuschätzen, wurden verschiedene maschinelle Lernverfahren untersucht. Insbesondere die Schätzung der Oberkörperorientierung ist auch Gegenstand dieser Arbeit. Eine Abgrenzung zu dieser Arbeit findet in Abschnitt 4.2.1 statt.

Im Europäischen Verbundprojekt „**A**daptable Ambient **L**iving **A**ssistent“ (ALIAS) wurde ein mobiles interaktives Robotersystem für ältere, alleinlebende Nutzer entwickelt. Ziel war dabei die Unterstützung des alltäglichen Lebens sowie die Förderung der Kommunikation und der sozialen Integration, um einer drohenden Vereinsamung vorzubeugen. Außerdem sollten die Nutzer durch das System regelmäßig zu kognitiven Aktivitäten stimuliert werden, um ihre Lebensqualität sowie ihre Gedächtnis- und Erinnerungsfähigkeit aufrecht zu erhalten. Der Beitrag des Fachgebietes NIKR erfolgte im Bereich der sozialverträglichen Navigation im häuslichen Umfeld. So wurden beispielsweise geeignete Wartepositionen für den Roboter bestimmt, an welchen der Nutzer beobachtet werden kann ohne ihn zu stören [Kessler, 2012].

Ziel des EU-Projektes „Integrated Cognitive Assistive & Domestic Companion Robotic Systems for Ability & Security“ (CompanionAble) war die Entwicklung eines sozialen Assistenzroboters für ältere Menschen im häuslichen Umfeld. Dieses Ziel wird durch die Forschergruppe **S**ervice-**R**obotik für die **G**esundheits**a**ssistenz (SERROGA) [Scheidig, 2015a, Scheidig, 2014] weiter verfolgt. Dabei steht vor allem die Akzeptanz der Nutzer (Senioren, Angehörige, Pflegedienste) im Vordergrund. Die Funktionalitäten als Gesundheitsassistent sind das Gesundheitsmonitoring und Dienste eines Sekretärs bzw. Unterhalters. Die robotischen Kernfunktionalitäten sind die intuitive Interaktion mit den Nutzern und die Möglichkeit der Fernsteuerung (Telepräsenz). Mit der Entwicklung sozialverträglicher Navigationsverhalten wurde im Rahmen von SERROGA zwar begonnen, diese sind jedoch im häuslichen Umfeld aufgrund der ohnehin sehr beengten Verhältnisse von eher untergeordneter Bedeutung.

1.3. Aufbau dieser Arbeit und zu erwartende Beiträge

Nach der Einführung des Themas dieser Arbeit werden im 2. Kapitel die Zielstellungen ausführlicher erläutert. Ausgehend von den Anwendungsszenarien werden vier ausgewählte soziale Verhaltensweisen genauer beschrieben und die dafür notwendigen technischen Funktionalitäten aufgeführt. Außerdem werden die Anforderungen an die gewünschte Lösung konkretisiert. Dies beinhaltet auch die Spezifikation der Roboterplattform. Des Weiteren wird eine Übersicht über bereits bestehende robotische Lösungen gegeben, welche im Hinblick auf die vier Verhaltensweisen eingeordnet werden. Dadurch sollen die erforderlichen Forschungsaufgaben heraus gestellt werden, um diese Arbeit bzgl. des Forschungsstandes einordnen zu können. Um einen Überblick über das Gesamtsystem zu geben, erfolgt zum Ende des 2. Kapitels die Vorstellung der Systemarchitektur dieser Arbeit mit den selbst bearbeiteten Teilkomponenten und den verwendeten Fremdentwicklungen. Darüber hinaus wird die Architektur bzgl. der populären Steuerungsparadigmen eingeordnet.

In Kapitel 3 wird die konkrete Realisierung der vier Verhaltensweisen des Roboters, welche die Zielstellung dieser Arbeit darstellen, beschrieben. Zur Realisierung dieser Verhaltensweisen ist umfangreiches Modellwissen über die Umwelt des Roboters, insbesondere über die anwesenden Personen, notwendig. Da dieses Wissen häufig mit Unsicherheiten behaftet ist, wird bei der Umsetzung der Verhaltensweisen besonderer Wert auf die Toleranz dieser Unsicherheiten gelegt. Außerdem wird dem reibungslosen Einsatz der Verhaltensweisen in einer Gesamtanwendung in realistischer Einsatzumgebung besondere Relevanz beigemessen.

In den Kapiteln 4 bis 6 werden die in dieser Arbeit entwickelten Verfahren zur Informationsgewinnung über die menschlichen Interaktionspartner des Roboters beschrieben. So wird in Kapitel 4 ein System zum Tracking der menschlichen Oberkörperposition und -drehung beschrieben. Das entwickelte System mit diesen Erkennungsleistungen besteht aus einem kamera- und einem laserbasierten Detektor, deren Hypothesen durch einen Kalman-Filter-basierten Tracker verfolgt werden. Der Neuheitswert im Bereich der laserbasierten Personendetektion besteht in der Steigerung der Robustheit und Effizienz, sowie der Unterscheidung von Gehhilfen. Im Bereich der kamerabasierten Detektion ergibt sich dieser aus der effizienten Klassifikation der Oberkörperorientierung.

In Kapitel 5 erfolgt die Beschreibung eines erscheinungsbasierten Ansatzes zur Verbesserung der Posenschätzung ausgehend von diesen getrackten Hypothesen. Außerdem unterstützt dieser Ansatz die Wiedererkennung von Personen.

Neben dem Wissen über die aktuellen Posen der anwesenden Personen, ist es für eine vorausschauende Umsetzung des sozialen Navigationsverhaltens erforderlich, dass deren Bewegung über mehrere Sekunden prädiziert werden kann. Ein zu diesem Zweck entwickeltes, on-line lernendes Verfahren wird im 6. Kapitel beschrieben. Mit diesem Verfahren ist es möglich, die Bewegung von Personen gegenüber dem jeweiligen Roboter, einschließlich seiner speziellen Erscheinung und seines Verhaltens, zu lernen. Außerdem passt sich die Prädiktion an, wenn sich das Verhalten der Personen ändert, weil sich diese beispielsweise an den Roboter gewöhnen, oder auf eine Verhaltensänderung des Roboters reagieren.

Die technischen Funktionalitäten werden einzeln in dem jeweiligen Kapitel evaluiert. Die Evaluation des realisierten Roboterverhaltens findet abschließend in Kapitel 7 statt. Wie oben beschrieben, basieren die in Kapitel 3 beschriebenen Verhaltensweisen auf den Erkennungsleistungen der Funktionalitäten und sollen deren Fehler und Unsicherheiten möglichst gut tolerieren. Deshalb findet die experimentelle Untersuchung der Verhaltensweisen unter dem Einfluss der technischen Funktionalitäten statt. Im 7. Kapitel werden somit das Gesamtsystem dieser Arbeit bewertet und die Auswirkungen der realisierten Verhaltensweisen auf das beobachtbare Verhalten des Roboters gezeigt.

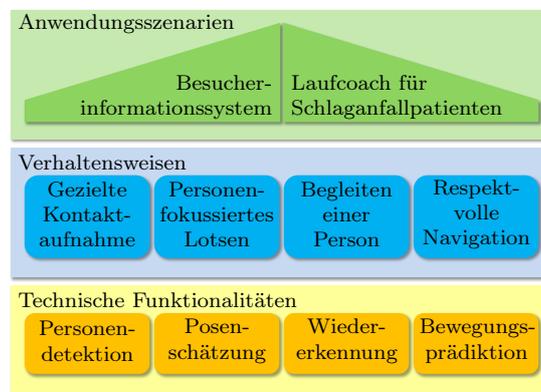


Abbildung 1.1. Aufbau dieser Arbeit: Im Hinblick auf die Anwendungsszenarien wurden vier Verhaltensweisen zur Verbesserung der sozialen Verträglichkeit des Roboters gewählt. Zu deren Umsetzung wurden die aufgeführten technischen Funktionalitäten bearbeitet.

Abbildung 1.1 zeigt eine Übersicht über die in dieser Arbeit bearbeiteten technischen Funktionalitäten und die sozialen Verhaltensweisen, zu deren Umsetzung die Funktionalitäten benötigt werden. Darüber hinaus sind Anwendungsszenarien dargestellt, in deren Kontext die Verhaltensweisen besonders relevant sind, um die soziale Akzeptanz des Roboters zu steigern. Diese Übersicht wird auch zu Beginn jedes Kapitels genutzt um zu zeigen, was in dem jeweiligen Kapitel behandelt wird.

1.4. Publikationen

Teile dieser Dissertation wurden bereits auf internationalen referierten Konferenzen publiziert:

[Weinrich, 2014b] C. Weinrich, T. Wengefeld, M. Volkhardt, A. Scheidig, H.-M. Gross, *Generic Distance-Invariant Features for Detecting People with Walking Aid in 2D Laser Range Data*. In Proc. of Int. Conf. on Intelligent Autonomous Systems (IAS), Padova, Italy, 2014

Diese Veröffentlichung beschreibt generische Merkmale zur Detektion von Objekten in Laserdaten. Mittels dieser Merkmale ist ein Personendetektor in der Lage auch Personen zu detektieren, wenn diese eine Gehhilfe benutzen, so dass deren Beine verdeckt werden.

[Weinrich, 2014a] C. Weinrich, T. Wengefeld, C. Schröter, H.-M. Gross, *People Detection and Distinction of their Walking Aids in 2D Laser Range Data based on Generic Distance-Invariant Features*. In Proc. of Int. Symposium on Robot and Human Interactive Communication (RO-MAN), pp. 767–773, Edinburgh (UK), 2014

Hier werden die zuvor erwähnten Merkmale nicht nur zur Detektion von Personen mit Gehhilfen verwendet, sondern zusätzlich wird eine Klassifikation der Gehhilfen durchgeführt. Dies wird in dieser Arbeit in Abschnitt 4.1 beschrieben.

[Weinrich, 2012] C. Weinrich, C. Vollmer, H.-M. Gross, *Estimation of Human Upper Body Orientation for Mobile Robotics using an SVM Decision Tree on Monocular Images*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 2147–2152, Vilamoura (Portugal), 2012

Der hier beschriebene kamerabasierte Detektor verwendet zur Klassifikation von Histogramms of Oriented Gradients (HOG)-Merkmalen einen Entscheidungsbaum mit Support Vector Machines (SVMs) als binäre Entscheider. Dadurch ist eine effiziente Unterscheidung von acht Oberkörperorientierungen möglich. Dieses Verfahren wird in Abschnitt 4.2 beschrieben.

[Weinrich, 2010] C. Weinrich, S. Müller, H.-M. Gross, *Appearance-based Person Tracking and 3D Pose Estimation of Upper-Body and Head*. In Proc. of Int. Scientific Colloquium (IWK), pp. 606 – 611, Ilmenau, Germany, 2010

In dieser Veröffentlichung wird ein erscheinungsbasiertes Verfahren zur Schätzung der 3D-Pose des Oberkörpers mit Gelenkstellung beschrieben.

[Weinrich, 2013b] C. Weinrich, M. Volkhardt, H.-M. Gross, *Appearance-Based 3D Upper-Body Pose Estimation and Person Recognition on Mobile Robots*. In Proc. of IEEE Int. Conf. on Systems, Man, and Cybernetics (SMC), pp. 4384–4390, Manchester (UK), 2013

Aufbauend auf [Weinrich, 2010] wird hier statt einzelnen Punkten eines Formmodells die geschlossene Oberfläche und Kontur eines 3D-Modells durch spezielle Shader auf der Grafikkarte bewertet. Dieses Verfahren, welches auch zur Wiedererkennung von Personen eingesetzt werden kann, ist in Kapitel 5 beschrieben.

[Weinrich, 2013a] C. Weinrich, M. Volkhardt, E. Einhorn, H.-M. Gross, *Prediction of Human Avoidance Behavior by Lifelong Learning for Socially Compliant Robot Navigation*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 376–381, Karlsruhe (Germany), 2013

Hier wird ein Verfahren zur Prädiktion der Bewegung von Personen in der Nähe des Roboters präsentiert. Um die tatsächlichen Trajektorien der Personen gegenüber dem konkreten Roboter präzisieren zu können, werden die Aufenthaltswahrscheinlichkeiten der Personen on-line gelernt. Dieses Prädiktionsverfahren ist in Kapitel 6 beschrieben.

An den nachfolgenden Publikationen wurde im Rahmen dieser Arbeit als Co-Autor mitgewirkt:

[Volkhardt, 2009] M. Volkhardt, C. Weinrich, C. Schröter, H.-M. Gross, *A Concept for Detection and Tracking of People in Smart Home Environments with a Mobile Robot*. In 2nd CompanionAble Workshop co-located with the 3rd European Conf. on Ambient Intelligence (AmI), AmI '09, Springer-Verlag, Berlin, Heidelberg, 2009

In dieser Veröffentlichung wird das Konzept für den später in dieser Arbeit verwendeten Tracker vorgestellt.

[Volkhardt, 2013a] M. Volkhardt, C. Weinrich, H.-M. Gross, *Multi-modal people tracking on a mobile companion robot*. In Proc. of European Conference on Mobile Robots (ECMR), pp. 288–293, 2013

[Volkhardt, 2013b] M. Volkhardt, C. Weinrich, H.-M. Gross, *People Tracking on a Mobile Companion Robot*. In Proc. of IEEE Int. Conf. on Systems, Man, and Cybernetics (SMC), pp. 4354–4359, 2013

Hier und in [Volkhardt, 2013a] wird der Personentracker zur Filterung der Hypothesen der Detektoren beschrieben. Die Schnittstelle und die Haupteigenschaften des Trackingsystems sind in Abschnitt 4.3 zu finden. Der Beitrag zu diesen Veröffentlichungen besteht hauptsächlich in der Verrechnung der Unsicherheiten bei der

Transformation der Hypothesen der unterschiedlichen Detektoren in ein gemeinsames statisches Koordinatensystem. Darüber hinaus ist die Covariance Intersection zur Fusion der Hypothesen eines Detektors, welche durch dasselbe Objekt hervorgerufen wurden, im Rahmen dieser Arbeit in den Tracker integriert wurden.

[Gross, 2014] H.-M. Gross, K. Debes, E. Einhorn, S. Mueller, A. Scheidig, C. Weinrich, A. Bley, C. Martin, *Mobile Robotic Rehabilitation Assistant for Walking and Orientation Training of Stroke Patients: A Report on Work in Progress*. In Proc. of IEEE Int. Conf. on Systems, Man, and Cybernetics (SMC), pp. 1880–1887, 2014
Der wesentliche Beitrag zu dieser Publikation besteht in den sozialverträglichen und personenfokussierten Navigationsverhalten zum Ausweichen, Führen und Folgen von Personen, welche in Kapitel 3 dieser Arbeit beschrieben sind.

[Scheidig, 2015b] A. Scheidig, E. Einhorn, C. Weinrich, M. Eisenbach, S. Müller, T. Schmiedel, T. Wengefeld, T. Trinh, H.-M. Gross, A. Bley, R. Scheidig, G. Pfeiffer, S. Meyer, S. Oelkers, *Robotischer Reha-Assistent zum Lauftraining von Patienten nach Schlaganfall: Erste Ergebnisse zum Laufcoach*. In Proc. of German Conf. on Ambient Assisted Living (AAL), 2015

Hier werden erste Ergebnisse zu den zuvor beschriebenen Navigationsverhalten und zum Rechtsfahrverhalten aufgeführt. Diese Ergebnisse sind auch in Kapitel 7 zu finden.

Darüber hinaus wurde auch an den nachfolgenden Publikationen, welche nicht in direktem Bezug zu dieser Arbeit stehen, als Co-Autor mit gewirkt:

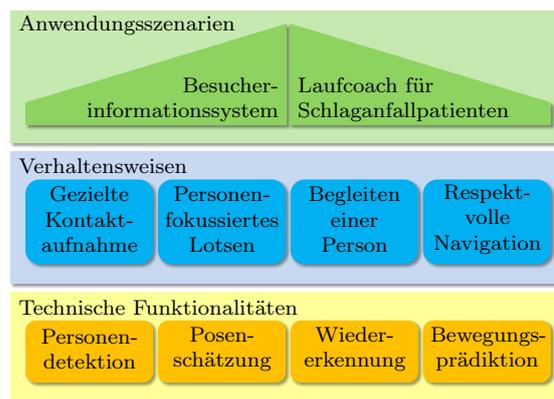
[Akanyeti, 2007] O. Akanyeti, U. Nehmzow, C. Weinrich, T. Kyriacou, S. Billings, *Programming Mobile Robots by Demonstration through System Identification*. In Proc. of European Conference on Mobile Robots (ECMR), pp. 192–197, Freiburg, Germany, 2007

[Nehmzow, 2007b] U. Nehmzow, O. Akanyeti, C. Weinrich, T. Kyriacou, S. Billings, *Robot programming by demonstration through system identification*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 801–806, 2007

[Nehmzow, 2007a] U. Nehmzow, O. Akanyeti, C. Weinrich, T. Kyriacou, S. Billings, *Learning by observation through system identification*. In Proc. Towards Autonomous Robotic Systems (TAROS), pp. 17–24, 2007

2

Anforderungsanalyse und Überblick auf das Gesamtsystem



Nach der Einleitung und der Motivation dieser Arbeit werden in diesem Kapitel die Anwendungsszenarien genauer beschrieben. Ausgehend von den Szenarien werden die für diese Arbeit relevanten Verhaltensweisen erläutert und die dafür notwendigen technischen Funktionalitäten abgeleitet. Anschließend werden die Rahmenbedingungen für die Umsetzung der technischen Lösung festgelegt. Das betrifft insbesondere die vorhandene Experimentierplattform. Um den Forschungsbedarf heraus zu stellen, werden danach einige vorhandene Robotersysteme für öffentliche Einsatzumgebungen bzgl. dieser Anforderungen eingeordnet. Den Abschluss des Kapitels bildet eine Übersicht auf das Gesamtsystem dieser Arbeit, bevor in den nachfolgenden Kapiteln auf die einzelnen Teilmodule eingegangen wird.

2.1. Anwendungsszenarien

Besucherinformationssystem: Lotsenroboter im Fakultätsgebäude

Aufbauend auf den in Abschnitt 1.2.4 beschriebenen Projekten zum Lotsenroboter im Baumarkt, wurde 2012 ein roboterbasiertes Besucherinformationssystem im Gebäude der Fakultät Informatik und Automatisierung der TU Ilmenau installiert [Stricker, 2012]. Die beiden Roboter „Konrad“ und „Suse“ ermöglichen die experimentelle Untersuchung

neu entwickelter Algorithmen im realen Einsatz in unmittelbarer Nähe zum Fachgebiet NIKR. Im Vordergrund steht dabei vor allem die Lokalisierung und Navigation über mehrere Ebenen des Fakultätsgebäudes unter Nutzung der Aufzüge. Eine gewisse Herausforderung war es, zu verhindern, dass die Roboter herabführende Treppen herunterstürzen bzw. gegen hoch führende Treppen fahren. Die Roboter fahren eigenständig auf ihre Ladestation und können komfortabel überwacht bzw. ferngesteuert werden. Bei den Lotsenrobotern wird besonderer Wert auf den Dialog mit den Nutzern gelegt. Mittels natürlicher Sprachausgaben sowie grafischer Ausgaben auf dem berührungsempfindlichen Bildschirm werden beispielsweise die Pfade zu gesuchten Zielen beschrieben.

In Bezug auf diese Arbeit soll erwähnt werden, dass die Lotsenroboter insbesondere für Personen relevant sind, welche nicht mit dem Gebäude und den Robotern vertraut sind. Das bedeutet, der Roboter soll auf intuitiv verständliche Weise versuchen, einen Kontakt zu den Besuchern des Fakultätsgebäudes aufzubauen, welche möglicherweise an einer Interaktion mit dem Roboter interessiert sind. Die Roboter sollen jedoch nicht von uninteressierten Personen als störend empfunden werden. Nachdem ein Roboter mit einem Besucher in Kontakt getreten ist und der Nutzer ein Ziel gewählt hat, soll der Roboter die Person so durch das Gebäude lotsen, dass diese ein ihr angenehmes Lauftempo wählen und unterwegs stehen bleiben kann, ohne dass der Kontakt zum Roboter abreißt. Das Fakultätsgebäude besteht aus sehr weitläufigen Hauptgängen und schmaleren Seitengängen. In den breiteren Gängen sollten die Roboter ausreichend Abstand zu den anwesenden Personen halten, damit sich diese nicht in ihrer Intimsphäre verletzt fühlen. In den schmalen Gängen ist es teilweise sogar erforderlich, dass der Roboter vor den Personen zurück weicht, um deren Weg nicht zu versperren.

ROREAS: Interaktiver robotischer Reha-Assistent

Schlaganfälle sind bei Erwachsenen der Industrienationen die häufigste Ursache für ernsthafte dauerhafte Behinderungen [Kolominsky-Rabas, 2006]. Weil die Schlaganfallwahrscheinlichkeit mit dem Alter steigt und sich die Überlebensrate von Schlaganfallpatienten verbessert hat [Go, 2013, S. e15 f.], wird sich dieser Anteil, auch wegen des demografischen Wandels, weiter erhöhen. Erfreulicherweise können jedoch viele Fähigkeiten, welche aufgrund eines Schlaganfalls verloren gegangen sind, wieder erlernt werden [Schallert, 2003]. Der Erfolg bzgl. des funktionellen Ergebnisses ist jedoch unter anderem abhängig von der Intensität und der Häufigkeit fokussierter Therapie [Teasell, 2005].

Dieser Herausforderung nimmt sich das Forschungsprojekt „Interaktiver **robotischer Reha-Assistent** für das Lauf- und Orientierungstraining von Patienten nach Schlaganfällen“

(ROREAS) [Gross, 2014] an, zu welchem diese Arbeit einen Beitrag leisten soll. Um die Selbstständigkeit der Patienten zu fördern, soll schon während der Rehabilitationsphase das Eigentraining als Ergänzung zu der konventionellen Physiotherapie unterstützt werden. Zu diesem Zweck wird ein Roboter entwickelt, welcher die Schlaganfallpatienten als Reha-Assistent während des Lauf- und Orientierungstrainings begleitet. Die Nutzer können sich durch den Roboter zu einem Zielpunkt im Krankenhaus (Cafeteria, Therapieräume, etc.) lotsen lassen oder im Rahmen des Orientierungstrainings selbst versuchen, den Weg zu einem Ziel zu finden. Dabei begleitet sie der Roboter und leistet bei eventuellem Abweichen vom Zielpfad Hilfestellung. Darüber hinaus kann der Roboter jederzeit Hilfe rufen und die Trainingsleistung der Patienten protokollieren. Es wird erwartet, dass die Patienten durch die Protokollierung zusätzlich zum Training motiviert werden und die Begleitung durch den Roboter zu einer Steigerung ihres Sicherheitsgefühls führt. Letztlich ist beabsichtigt, dass die Patienten ein intensiveres, protokolliertes Eigentraining durchführen, welches eine wesentliche Ergänzung zu den Trainingseinheiten mit einem professionellen Therapeuten darstellt.

Die motorischen und kognitiven Einschränkungen der Patienten stellen eine besondere Anforderung an das Navigationsverhalten des Roboters dar. So haben die Patienten ein gesteigertes Sicherheitsbedürfnis, da sie weniger gut in der Lage sind, eine Kollision mit dem Roboter aktiv zu verhindern bzw. auf eine Kollision angemessen zu reagieren. Dabei reicht es nicht aus sicherzustellen, dass der Roboter nicht mit einem Patienten kollidiert. Vielmehr ist es notwendig, dass die Patienten durch das Navigationsverhalten nicht beunruhigt oder gar erschreckt werden und sich in der Nähe des Roboters sicher fühlen. In den Gängen des Krankenhausgebäudes verkehren zeitweise sehr viele Personen. Zusätzlich wird die Durchfahrtsbreite der Gänge häufig durch Stationswagen oder wartende Patienten etc. eingeschränkt. Dadurch ist es erforderlich, dass die Roboter so navigieren, dass die Abläufe von Patienten und Personal nicht behindert werden. Während ein Patient von dem Roboter gelotst wird oder der Roboter einen Patienten begleitet, ist die Anpassung der Fahrgeschwindigkeit an den Nutzer nicht nur in Bezug auf dessen Wohlbefinden relevant. Da der Patient gemeinsam mit dem Roboter Laufübungen im Kontext seines Trainingsplanes durchführt, hätten Einschränkungen der Laufübungen auch einen negativen Einfluss auf den Genesungsprozess.

2.2. Benötigte sozialverträgliche Verhaltensweisen

Das Ziel dieser Arbeit ist die Verbesserung der sozialen Verträglichkeit mobiler Serviceroboter in öffentlichen Einsatzumgebungen. Im Wesentlichen sollen vier Navigationsverhal-

ten entwickelt werden, welche direkt auf die in der Umgebung des Roboters anwesenden Personen abgestimmt sind. Dazu ist es erforderlich, dass verschiedene Informationen über die Personen gewonnen werden. Bevor in Abschnitt 2.3 die bearbeiteten technischen Funktionalitäten zur Nutzereinschätzung erläutert werden, folgt eine genauere Beschreibung der bereits in Absatz 1.2.3 erwähnten Verhaltensweisen.

Gezielte Kontaktaufnahme Gerade im öffentlichen Einsatz wird die Robotertechnik teilweise noch mit Skepsis betrachtet. Viele Menschen würden nicht von sich aus mit einem Roboter in Kontakt treten, da beispielsweise nicht immer erkennbar ist, welche Dienste der Roboter zur Verfügung stellt. Deshalb ist es erforderlich, dass die Interaktion vom Roboter aktiv gefördert wird. Gleichfalls darf das Verhalten zur Kontakthanbahnung von den anwesenden Menschen auch nicht als störend empfunden werden.

„Bevor eine Interaktion beginnen kann, müssen die Teilnehmer einander Interaktionsbereitschaft signalisieren. Dies geschieht in der Mensch-Mensch-Interaktion typischerweise über Blickkontakt [Goffman, 1963, Goffman, 1971, Argyle, 1976, Goodwin, 1981] sowie über die Kopf- und Körperausrichtung [Yamazaki, 2007]. Dies gilt auch für die Mensch-Roboter-Interaktion (HRI): So ermutigt der Blickkontakt des Roboters die Nutzer zur Aufnahme und Aufrechterhaltung einer Interaktion [Mutlu, 2009, Holthaus, 2010]. Umgekehrt sind der Blickkontakt einer Person und vor allem die Ausrichtung ihres Kopfes und Oberkörpers Indikatoren, anhand derer ein sozialverträglicher mobiler Assistenzroboter frühzeitig das Interaktionsinteresse eines menschlichen Gegenübers einschätzen sollte. Sofern der Roboter einen potenziell interessierten Kunden in seinem Umfeld detektiert, sollte er darauf mit Signalen der Interaktionsbereitschaft (Blickkontakt und Begrüßung) reagieren.“ [Weinrich, 2011, S. 4]

Nachdem die Interaktionsbereitschaft aus der Distanz signalisiert wurde, folgt in der Mensch-Mensch-Interaktion typischerweise eine räumliche Annäherung, während welcher der Blickkontakt gehalten wird. In verschiedenen Arbeiten wurde in diesem Zusammenhang untersucht wie sich die Interaktionsdistanz [Hall, 1963], die Richtung [Dautenhahn, 2006, Hüttenrauch, 2006], die Geschwindigkeit [Sardar, 2012, Shi, 2008] und die Erscheinung des Roboters [Brandl, 2013] auf das Wohlbefinden der Person auswirken.

Personenfokussiertes Lotsen Im Wesentlichen besteht die Lotsenfunktion eines Roboters darin, zu einem gewünschten Ziel zu fahren und dadurch auch dem Nutzer den Weg zu diesem Ziel zu weisen [Buhmann, 1995]. Allerdings steigert es das Wohlbefinden der geloteten Person, wenn der Roboter seine Geschwindigkeit an die bevorzugte Geschwindigkeit der Person anpasst und auch registriert, ob die Person noch folgt. In

Bezug auf das Anwendungsszenario Gebäudelotse wird dadurch erreicht, dass sich der Nutzer weder unter Druck gesetzt fühlt, noch das Gefühl hat aufgehalten zu werden. Im Kontext von ROREAS steht neben dem reinen Erreichen des Ziels vor allem das Lauftraining im Vordergrund. Die Fahrgeschwindigkeit des Roboters hat somit nicht nur Einfluss auf das emotionale Wohlbefinden des Patienten, sondern auch auf den physischen Trainingseffekt. Fährt der Roboter zu schnell, so würde die Sturzgefahr mancher Patienten mit Gehbehinderung steigen oder sie könnten gar nicht mehr folgen. Bei zu langsamer Fahrt würden andere Patienten nicht ausreichend trainiert werden. Deshalb ist es erforderlich, dass für den Patienten erkennbar ist, wie der Roboter seine Geschwindigkeit an den Nutzer anpasst.

Begleiten einer Person Eine direkte Abstimmung des Navigationsverhaltens ist auch dann erforderlich, wenn ein Serviceroboter eine Person auf ihrem Weg begleitet und seine Dienste mobil zur Verfügung stellt. Im Rahmen des ROREAS-Projektes wird der Nutzer beispielsweise begleitet, um sicher zu stellen, dass er sich auf dem richtigen Weg zu einem vorher definierten Ziel befindet. Wie bei dem *personenfokussierten Lotsen* ist es auch hier notwendig, dass sich der Nutzer nicht durch den Roboter bedrängt fühlt. Andererseits darf sich der Roboter auch nicht so weit von der Person entfernen, dass er diese verliert oder die Person nicht mehr auf dessen Dienste zugreifen kann.

Respektvolle Navigation Beim Ausweichen und Passieren von Personen sowie dem Zurückweichen vor Personen hat die Proxemik [Hall, 1963] in der Mensch-Mensch-Interaktion eine wesentliche Bedeutung. Dementsprechend sollten auch Roboter die räumliche Privatsphäre (Personal Space) der Menschen respektieren [Sardar, 2012, Walters, 2011, Takayama, 2009, Hüttenrauch, 2006, Pacchierotti, 2006]. Dafür dürfen Menschen im Gegensatz zu anderen Objekten nicht ausschließlich als physische Hindernisse betrachtet werden, sondern es müssen bestimmte Abstände gewahrt werden. Der durch die Menschen beanspruchte Raum hängt auch von der Orientierung des Menschen und der Wahrnehmung des Roboters durch den Menschen ab. Sollte eine Unterschreitung der entsprechenden Abstände unausweichlich sein, so müsste zumindest die Geschwindigkeit des Roboters reduziert werden.

Die zuvor beschriebenen Verhaltensweisen zur sozialverträglichen Navigation werden durch die Bewegung der anwesenden Personen beeinflusst. Würden die Verhaltensweisen immer nur die aktuellen Posen der Personen berücksichtigen, würde der Roboter häufig zu spät reagieren oder gar Bewegungen ausführen, welche sich später als ungünstig erweisen würden. Der Roboter würde seine Bewegungsrichtung häufig ändern, und sein Verhalten wäre

für die Anwesenden schwer vorhersehbar. Damit der Roboter gleichmäßig navigiert muss er die Bewegung der anwesenden Personen präzisieren. Auf diese Weise wird beispielsweise bei der *respektvollen Navigation* mit dem Ausweichen einer herannahenden Person begonnen, bevor der Personal Space (persönlich beanspruchter Freiraum) verletzt würde.

2.3. Technische Funktionalitäten

Damit die beschriebenen Verhaltensweisen realisiert werden können, sind verschiedene technische Funktionalitäten zur Personenwahrnehmung erforderlich, welche im Rahmen dieser Arbeit entwickelt oder angepasst werden.

Ziel der *Personendetektion* ist es zu erkennen, wo sich in der Umgebung des Roboters Personen befinden. Häufig ist es jedoch hilfreich, wenn neben der detektierten Position von Personen auch deren *Orientierung geschätzt* wird. Mittels eines Trackers werden die einzelnen geschätzten Posen in einen zeitlichen Kontext gesetzt, um die Bewegung der Personen zu verfolgen. Falls eine Person nicht durchgängig getrackt werden kann, ist es häufig erforderlich, *Personen wiederzuerkennen*, damit eine neue getrackte Bewegungsspur der selben Person zugeordnet werden kann. Auf Basis der Bewegungsspuren kann die zukünftige *Bewegung der Personen* *prädictiert* werden.

Je nach Verhaltensweise sind bestimmte Funktionalitäten von mehr oder weniger großer Bedeutung. Der Zusammenhang zwischen den Verhaltensweisen und den dafür notwendigen technischen Funktionalitäten ist in Tabelle 2.1 dargestellt.

Tabelle 2.1. Verhaltensweisen und Funktionalitäten: Zusammenhang zwischen Verhaltensweisen und den dafür notwendigen (■) bzw. hilfreichen (◻) technischen Funktionalitäten

Verhaltensweise Funktionalität	Gezielte Kontakt- aufnahme	Personen- fokussiertes Lotsen	Begleiten einer Person	Respektvolle Navigation
Personen- detektion	■	■	■	■
Orientierungs- schätzung	■	◻	◻	◻
Wieder- erkennung	◻	■	■	◻
Bewegungs- prädiction	◻	■	■	■

2.4. Rahmenbedingungen und Experimentierplattform

Für die Umsetzung der gewünschten Verhaltensweisen und der dafür notwendigen technischen Funktionalitäten wurden im Rahmen dieser Arbeit verschiedene Rahmenbedingungen festgelegt. Diese sollen sicherstellen, dass die Gesamtlösung praktisch auf mobilen Robotern einsetzbar ist und auch nach Abschluss der Arbeit in andere Arbeiten integriert und dort verändert werden kann. Im Folgenden werden die Rahmenbedingungen kurz vorgestellt:

Natürliche Einsatzumgebung Der Einsatz des Roboters darf für den regulären Betrieb keine Veränderungen an der Einsatzumgebung erfordern. Davon ausgenommen ist die Ausstattung der Einsatzumgebung zu Sicherheitszwecken, um beispielsweise sicher zu stellen, dass der Roboter auch bei fehlerhafter Lokalisierung keine Treppen hinunter fällt. Außerdem benötigt der Roboter eine Ladestation. Eine Kommunikationsschnittstelle mittels drahtlosem Funknetzwerk ist je nach Anwendung hilfreich.

Verwendung vorhandener Sensorik Die in dieser Arbeit eingesetzten Verfahren sollen mit der Nutzung der auf der Roboterplattform Cora vorhandenen Sensorik auskommen. Die Experimentierplattform Cora mit der nachfolgend aufgeführten Sensorik ist in Abbildung 2.1 dargestellt.

Omnidirektionale Farbkamera: Die omnidirektionale Farbkamera der Firma Metra-Labs, bestehend aus vier einzelnen uEye UI-122xLE-C Kameras, ist in ca. 1,5 m Höhe auf dem Kopf des Roboters angebracht. Der vertikale Öffnungswinkel beträgt nach Rektifizierung ca. 65° . Jede einzelne Kamera hat eine Auflösung von 752×480 Pixeln und nach Rektifizierung einen horizontalen Öffnungswinkel von ca. 90° .

Berührungsempfindliches Display: Zur Wahrnehmung von Benutzereingaben verfügt der Roboter über ein berührungsempfindliches Display.

Tiefenkamera: Zur 3D-Hindernisvermeidung ist die Roboterplattform mit einer Asus Xtion Pro Live Tiefenkamera auf einer Schwenk-Neige-Einheit ausgestattet.

Laserscanner: Zur omnidirektionalen Tiefenwahrnehmung der Umgebung des Roboters in einer horizontalen Ebene 0,4 m über dem Boden, ist dieser mit zwei Laserscannern (Laser Range Scanner, LRS) ausgestattet. Ein SICK S300 mit 270° Öffnungswinkel scannt den Bereich vor dem Roboter. Ein weiterer Laserscanner LZR-U901 der Firma BEA mit ca. 96° Öffnungswinkel dient zur Wahrnehmung des verbleibenden Bereichs hinter dem Roboter.

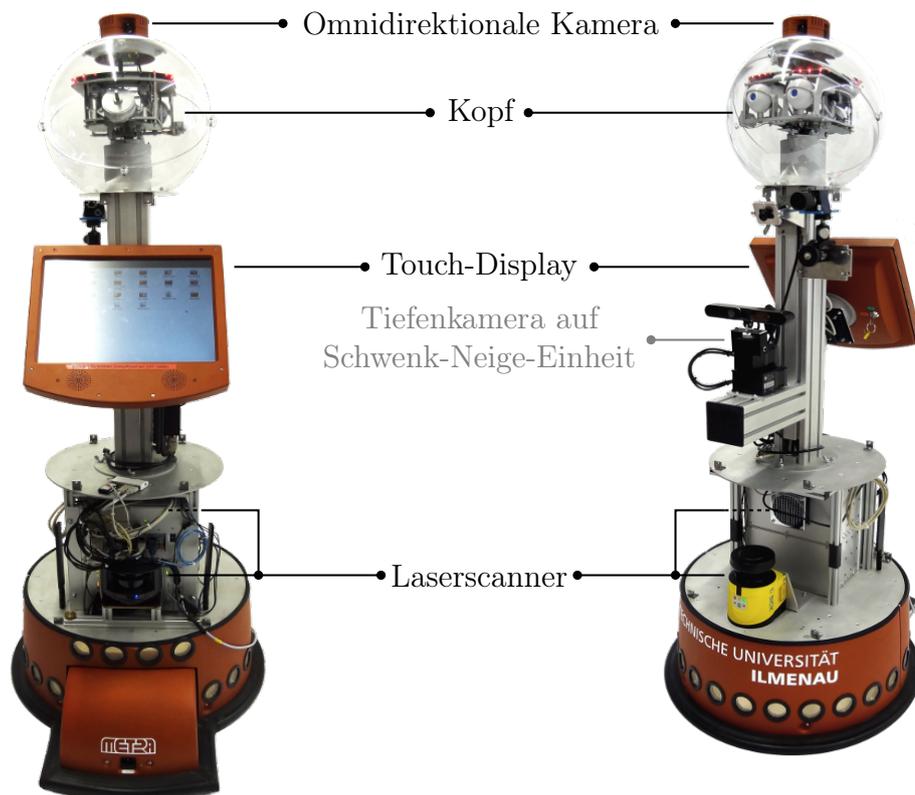


Abbildung 2.1. Experimentierplattform Cora: Zur Durchführung der Experimente wurde im Rahmen dieser Arbeit vorrangig dieser Roboter, welcher auf der Roboterplattform SCITOS A5 der Firma MetraLabs basiert, eingesetzt.

Stoßleistsensor (Bumper): Um in Notsituationen die Motoren des Roboters zum Stillstand zu bringen und Kollisionen mit der Umgebung wahrzunehmen, ist der Roboter rundherum knapp über dem Boden mit einem Bumper ausgestattet.

Odometer: Zur Wahrnehmung der zurückgelegten Strecke des Roboters sind die Motoren des Differentialantriebes mit Odometern versehen.

Verwendung vorhandener Aktorik Die Basisplattform Coras ist ein SCITOS A5 der Firma MetraLabs. Sie ist mit einem Differentialantrieb mit zwei angetriebenen Rädern und einem Castorrad ausgestattet. Darüber hinaus besitzt sie zur Interaktion ein 15-Zoll Display und einen Kopf mit fünf Freiheitsgraden.

Echtzeitfähigkeit bei beschränkter Rechenkapazität Die mobile Roboterplattform Cora ist mit Bleigelakkus mit einer Gesamtenergie von 1008 Wh ausgerüstet. Damit der Roboter damit ca. 8h ohne Unterbrechung operieren kann, ist der Onboard-PC mit einem 2.7GHz Intel® Core™ i7-2620M Prozessor und 8 GB Arbeitsspeicher ausgestattet. Zusätzlich befindet sich auf dem Roboter noch ein kompakter, lüfterloser PC, welcher mit einem 1.7GHz Intel® Core™ i7-3517UE Prozessor und 8 GB Arbeitsspeicher bestückt ist. Die beiden PCs sind per LAN verbunden. Mehr als diese Rechenkapazität darf zur Ausführung der technischen Gesamtlösung aus integrierten und selbst entwickelten Algorithmen nicht benötigt werden.

Austauschbarkeit, Erweiterbarkeit, Wartung Die zu entwickelnde Gesamtlösung wird auf dem Robotik-Framework „Middleware for Robotic Applications“ (MIRA) [Einhorn, 2012] basieren. Durch eine einheitliche Datenrepräsentation zur Kommunikation zwischen einzelnen Modulen soll der problemlose Austausch einzelner Module bzw. die unkomplizierte Erweiterung des Gesamtsystems unterstützt werden.

Technische Evaluation nach sozialwissenschaftlichen Kriterien Da durch diese Arbeit die Mensch-Roboter-Interaktion verbessert werden soll, wäre es optimal, wenn auch eine sozialwissenschaftliche Evaluation der sozialen Verträglichkeit der entwickelten Verfahren mittels Nutzerstudien durchgeführt würde. Aufgrund fehlender eigener Expertise im Bereich der sozialwissenschaftlichen Evaluation, die zudem den Rahmen dieser Arbeit sprengen würde, findet jedoch nur eine technische Evaluation statt. Allerdings werden dabei sozialwissenschaftliche Kriterien aus der sozialwissenschaftlichen Literatur angewendet.

2.5. Zum Forschungsstand sozialer Assistenzroboter in öffentlichen Einsatzumgebungen

Seit den 1990er Jahren wurde begonnen, mobile Serviceroboter für öffentliche Einsatzumgebungen zu entwickeln [Buhmann, 1995]. Anfangs stand dabei vor allem die instrumentale Interaktionsebene der Mensch-Roboter-Interaktion im Vordergrund. Im Laufe der Zeit wurde jedoch der sozio-emotionale Aspekt zur Steigerung der Akzeptanz der Roboter verstärkt Forschungsgegenstand. Tabelle 2.2 gibt einen Überblick über Robotersysteme mit Bezug zu den in dieser Arbeit betrachteten Anwendungsszenarien. Ausführlichere Informationen zu dieser Tabelle sind im Anhang A zu finden. Aufgeführt sind vorrangig

Lotsenroboter, welche für öffentliche Einsatzumgebungen und reale Anwendung entwickelt wurden. Da für diese Arbeit insbesondere die implementierten technischen Funktionalitäten für nutzerzentriertes Navigationsverhalten von Interesse sind, wurden die publizierten Arbeiten gezielt dahingehend aufgearbeitet, um zu verdeutlichen, wo derzeit noch Forschungsbedarf für eine sozialverträgliche Navigation besteht.

Tabelle 2.2. Prominente Robotersysteme mit Bezug zu dieser Arbeit: Für jedes System wird dargestellt, welche sozialen Verhaltensweisen implementiert (■) sind. Ausführlichere Informationen zu dieser Tabelle sind im Anhang A zu finden.

Verhaltensweise System	Gezielte Kontakt- aufnahme	Personen- fokussiertes Lotsen	Begleiten einer Person	Respektvolle Navigation
Robotersysteme in natürlichen Einsatzumgebungen				
RHINO & MINER-VA	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Nursebot Pearl	<input type="checkbox"/>	■	<input type="checkbox"/>	<input type="checkbox"/>
TOOMAS	■	■	<input type="checkbox"/>	<input type="checkbox"/>
RoBoX	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Robovie	■	<input type="checkbox"/>	■	<input type="checkbox"/>
RoboCart	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
InBOT(2)	<input type="checkbox"/>	■	<input type="checkbox"/>	■
Laborstudien				
Nomadic Scout II	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BIRON	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
PeopleBot	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Grace	<input type="checkbox"/>	<input type="checkbox"/>	■	<input type="checkbox"/>

Wie Tabelle 2.2 zeigt, gibt es bislang kein System, in dem alle notwendigen technisch-algorithmischen Voraussetzungen für eine wirklich autonome, nutzerzentrierte und damit auch sozialverträgliche Roboternavigation geschaffen wurden.

Die aufgeführten Robotersysteme haben ihren Schwerpunkt entweder in einer praktischen Anwendung in einer natürlichen Einsatzumgebung oder in der Umsetzung einzelner sozialverträglicher Verhaltensweisen. Die Kombination aus beidem war jedoch in der recherchierten Literatur bisher nicht zu finden. Eine besondere Herausforderung ist die Informationsgewinnung über die menschlichen Interaktionspartner mit onboard-Sensorik ohne die Nutzer instruieren oder besonders ausstatten zu müssen. Bei verschiedenen Systemen, wie Robovie [Kanda, 2009], werden jedoch externe Laserscanner zum Personen-tracking eingesetzt. Bei InBOT [Germa, 2009] tragen die Nutzer einen RFID-Chip, und der Roboter ist mit RFID-Antennen ausgestattet und kann so auch Personen unterscheiden.

Hier soll diese Arbeit ansetzen um die Informationsgewinnung über die menschlichen Interaktionspartner mit onboard-Sensorik und -Rechentechnik für die sozialverträgliche Navigation in natürlichen, öffentlichen Einsatzumgebungen zu ermöglichen. Eine funktionierende Gesamtlösung erfordert, dass bei der Umsetzung der Verhaltensweisen die Unsicherheiten der Informationen über die anwesenden Personen toleriert werden.

2.6. Gesamtsystem und Steuerungsarchitektur

In diesem Abschnitt wird das Gesamtsystem, welches zur Umsetzung der Zielstellungen dieser Arbeit zum Einsatz kommt, vorgestellt. Die Architektur des Systems ist in Abbildung 2.2 dargestellt. Das verwendete Kontrollparadigma wird dem Paradigma Behavior-Based Control zugeordnet. Eine Erläuterung der gebräuchlichen Steuerparadigmen für Roboter und eine Begründung dieser Zuordnung ist im Anhang in Kapitel B zu finden.

Die einzelnen Module der Architektur sind in verschiedenen Schichten strukturiert, welche nachfolgend erklärt werden. Im Kontext des Paradigmas Behavior-Based Control werden diese Module als Behaviors bezeichnet. In dieser Arbeit wird der Begriff Behavior jedoch zur Bezeichnung der für einen Beobachter sichtbaren Aktivitätsmuster des Roboters genutzt.

Application Layer Die wesentliche Komponente der Application Layer ist eine Zustandsmaschine, welche im Kontext des *Behavior-Based Control* Paradigmas einen zentralen Arbiträr darstellt. Die Zustandsübergänge werden durch die grafische Nutzerschnittstelle (GUI) und Informationen von den Behaviors, der untergeordneten Behavior Layer, beeinflusst. Für jeden Zustand wird ein entsprechendes Behavior aktiviert und die GUI angepasst.

Behavior Layer Die Behaviors, wie z. B. das *personenfokussierte Lotsen* oder *Begleiten einer Person*, stellen die Schnittstelle zwischen der Application Layer und den untergeordneten Skills dar. So werden durch die Behaviors einzelne Skills gestartet bzw. beendet und Informationen zwischen der Skill Layer und der Application Layer ausgetauscht. Viele der Behaviors beinhalten selbst auch eine Zustandsmaschine um gewisse Abläufe zu realisieren oder auf Informationen der Skills zu reagieren. Ein wesentliches Instrument der Behaviors zur Aktivierung und Deaktivierung einzelner Skills ist die Generierung von Tasks, den aktuellen Navigationsaufgaben des Roboters. Ein Task besteht aus mehreren Subtasks. Bestimmte Skills sind nur aktiv, wenn der aktuelle Task auch einen Subtask enthält, für welchen der Skill zuständig ist.

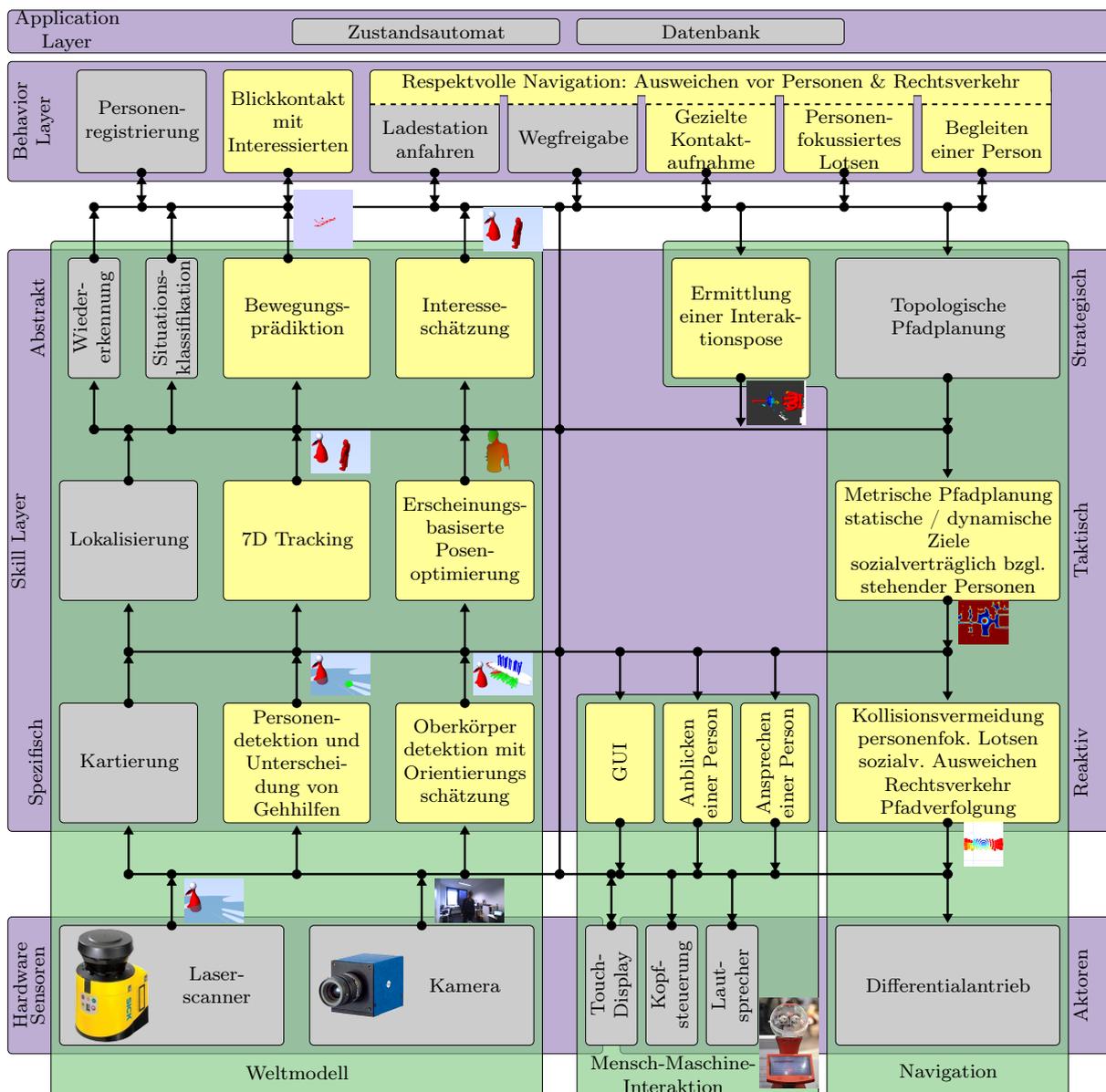


Abbildung 2.2. Systemarchitektur: Architektur des Gesamtsystems zur Umsetzung der Zielstellungen dieser Arbeit. Die gelb hervorgehobenen Module sind Gegenstand dieser Arbeit und werden in den entsprechenden Kapiteln beschrieben. Bildquellen: www.sick.de (Laserscanner), www.svs-vistek.com (Kamera)

Skill Layer In der Skill Layer sind die übrigen Module eingeordnet, welche in verschiedenen Abstraktionsstufen ein verteiltes Modell der Umwelt bilden und den Roboter steuern. Die Skills zur Modellbildung sind in Abbildung 2.2 entsprechend des Abstraktionslevels ihrer Informationen aufgeteilt. Ebenso sind die Skills zur Navigation in einer strategischen, einer taktischen und einer reaktiven Schicht angeordnet.

Die verteilte Repräsentation der Umwelt durch die einzelnen Skills ist eine wesentliche Eigenschaft dieser Architektur und der wesentliche Grund für die Zuordnung zum Para-

digma Behavior-Based Control (Kapitel B). Alle Skills beinhalten eine mehr oder weniger stark ausgeprägte Zustands- und Planungskomponente. Dank der verteilten Repräsentation der Umwelt können die einzelnen Skills auf verschiedenen PCs berechnet werden. Dafür brauchen nur die Daten zwischen den PCs übertragen zu werden, welche durch ein Modul auf einem PC erzeugt und durch ein Modul auf einem anderen PC verwendet werden.

Nachdem die einzelnen Ebenen der Architektur vorgestellt wurden, werden nun weitere Eigenschaften des Gesamtsystems beschrieben, welche im Kontext der Kontrollparadigmen relevant sind.

Fusion der Aktionsvorschläge Im Kontext der Navigation wird zur reaktiven Bewegungsplanung der „Dynamic Window Approach“ (DWA) [Fox, 1997] verwendet. Um nicht zu viel aus Kapitel 3 vorweg zu nehmen, sei hier nur erwähnt, dass sich dieses Verfahren sehr gut zur Fusion der Aktionsvorschläge der einzelnen reaktiven Skills, welche beim DWA als Objectives bezeichnet werden, eignet. Bei der Potentialfeldmethode werden die Potentialfelder der einzelnen Zielstellungen gewichtet überlagert. Im Gegensatz dazu werden beim DWA durch jedes Objective die zukünftigen, physikalisch durch den Roboter erreichbaren, Geschwindigkeiten bewertet, und diese Kostenfunktionen über dem zweidimensionalen Raum möglicher Geschwindigkeiten werden gewichtet überlagert. Die Menge an aktiven Objectives wird, wie weiter oben beschrieben, durch den aktuellen Task festgelegt.

Prädiktion der Dynamik des Weltmodells Eine wesentliche Anforderung an die reaktiven Objectives des DWA ist, innerhalb weniger Millisekunden auf Veränderungen der Umwelt zu reagieren. Zur Berechnung des Umweltmodells benötigen einige Skills der abstrakteren Schichten jedoch weitaus mehr Berechnungszeit. Modellieren diese Skills dynamische Bestandteile des Umweltmodells, wären die entsprechenden Informationen bereits veraltet, sobald sie den Objectives zur Verfügung stehen. Um die Berechnungszeit aufwändiger Detektions- und Planungsalgorithmen auszugleichen, werden die entsprechenden dynamischen Bestandteile des Umweltmodells prädiziert. Die prädizierten Informationen weisen entsprechend zusätzliche Unsicherheiten auf. Das Umweltmodell soll eine Abschätzung dieser Unsicherheiten beinhalten, damit diese bei der Weiterverarbeitung berücksichtigt werden können.

Details zur verwendeten Middleware MIRA In diesem Abschnitt werden ausgewählte Details der verwendeten Middleware aufgeführt, welche in Bezug auf die zuvor beschriebene Architektur relevant sind. Grundsätzlich könnten für die Entwicklung der Roboterapplikation dieser Arbeit verschiedene Middlewares verwendet werden. Besonders

weit verbreitet ist beispielsweise das Robot Operating System¹ (ROS). Da ROS jedoch verschiedene Anforderungen des Fachgebietes NIKR und der MetraLabs GmbH² nicht erfüllt [Einhorn, 2012], wurde, im Rahmen der bearbeiteten Förderprojekte CompanionAble, ALIAS, SERROGA und ROREAS, von diesen beiden Partnern die Middleware for Robotic Applications MIRA³ entwickelt. MIRA stellt somit auch das Framework dieser Arbeit dar. Die in Abbildung 2.2 gezeigten Module aller Schichten sind als MIRA-Units in den Programmiersprachen C++ und Python implementiert. Zum Datenaustausch zwischen den Units werden sogenannte Channels genutzt. Diese erlauben die nicht-blockierende Nutzung der Daten durch verschiedene Units. Zusätzlich zu den eigentlichen Daten speichert jedes Datenelement (Slot) eines Channels einen Zeitstempel und einen Frame. Der Frame bezeichnet das Koordinatensystem, auf welches sich die Daten beziehen. Die relative Lage der Frames zueinander wird ebenfalls auf Channels gespeichert, wobei auch Unsicherheiten bzgl. der relativen Lage durch Kovarianzmatritzen repräsentiert werden. Ein Transformation-Framework erlaubt es, die relative Lage zwischen zwei Frames auch über mehrere Frames hinweg zu berechnen. Somit ist das Transformation-Framework gut geeignet, um das unsichere Modellwissen beispielsweise über Personenhypothesen und die Roboterposition zu repräsentieren.

So, wie der Datenaustausch mittels der Channels transparent über Prozess- und Rechengrenzen hinweg möglich ist, können mittels RPCs (Remote Procedure Calls) auch Prozeduren transparent und uneingeschränkt über Prozess- und Rechengrenzen hinweg aufgerufen werden. Auf diese Weise wird durch MIRA die Nutzung des verteilten Weltmodells auf effiziente Weise unterstützt. Sensoren oder Prozessoren verschiedener PCs können mit geringem Aufwand genutzt werden.

2.7. Fazit

In diesem Kapitel wurden die sozialverträglichen und personenfokussierten Verhaltensweisen anhand der Anwendungsszenarien motiviert und die dafür notwendigen technischen Funktionalitäten spezifiziert. Darüber hinaus wurden der Forschungsbedarf bzgl. der Funktionalitäten heraus gestellt und die Rahmenbedingungen im Kontext der Anwendungen erläutert. Abschließend wurde gezeigt, wie sich die einzelnen Softwaremodule in das Gesamtsystem dieser Arbeit einordnen. Damit sollte ein ausreichender Überblick gegeben sein, um im nächsten Kapitel mit einer detaillierteren Beschreibung der Umsetzung der Navigationsverhalten fortzusetzen.

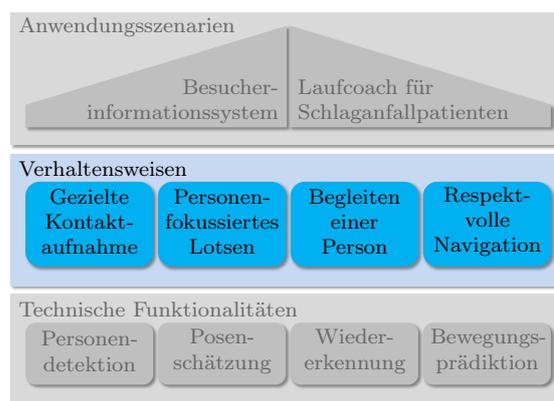
¹www.ros.org

²www.metralabs.com

³www.mira-project.org

3

Umsetzung der Verhaltensweisen zur Steigerung der sozialen Verträglichkeit



Nachdem in den vorangegangenen Kapiteln die in dieser Arbeit bearbeiteten Verhaltensweisen vorgestellt und motiviert wurden, wird in diesem Kapitel beschrieben, wie diese Verhaltensweisen tatsächlich umgesetzt werden. Dazu wird im nächsten Abschnitt die grundsätzliche Teilarchitektur zur Bewegungssteuerung des Roboters vorgestellt. Danach wird beschrieben, wie die in Kapitel 2.2 aufgeführten sozialen Verhaltensweisen realisiert werden.

Zur Umsetzung der Verhaltensweisen werden verschiedene Informationen über die anwesenden Personen in der Nähe des Roboters benötigt. Damit motiviert dieses Kapitel auch die technischen Funktionalitäten zur Informationsgewinnung, auf welche in den nachfolgenden Kapiteln eingegangen wird.

3.1. Die drei Ebenen der Bewegungsplanung

Das Navigationsverhalten des Roboters wird auf reaktiver, taktischer und strategischer Ebene entwickelt (Abb. 3.1). Auf *reaktiver Ebene* findet die Auswahl der unmittelbaren Bewegungskommandos unter Berücksichtigung verschiedener Zielstellungen, den sogenannten Objectives, statt. Dies erfolgt mehrfach pro Sekunde, wobei innerhalb weniger

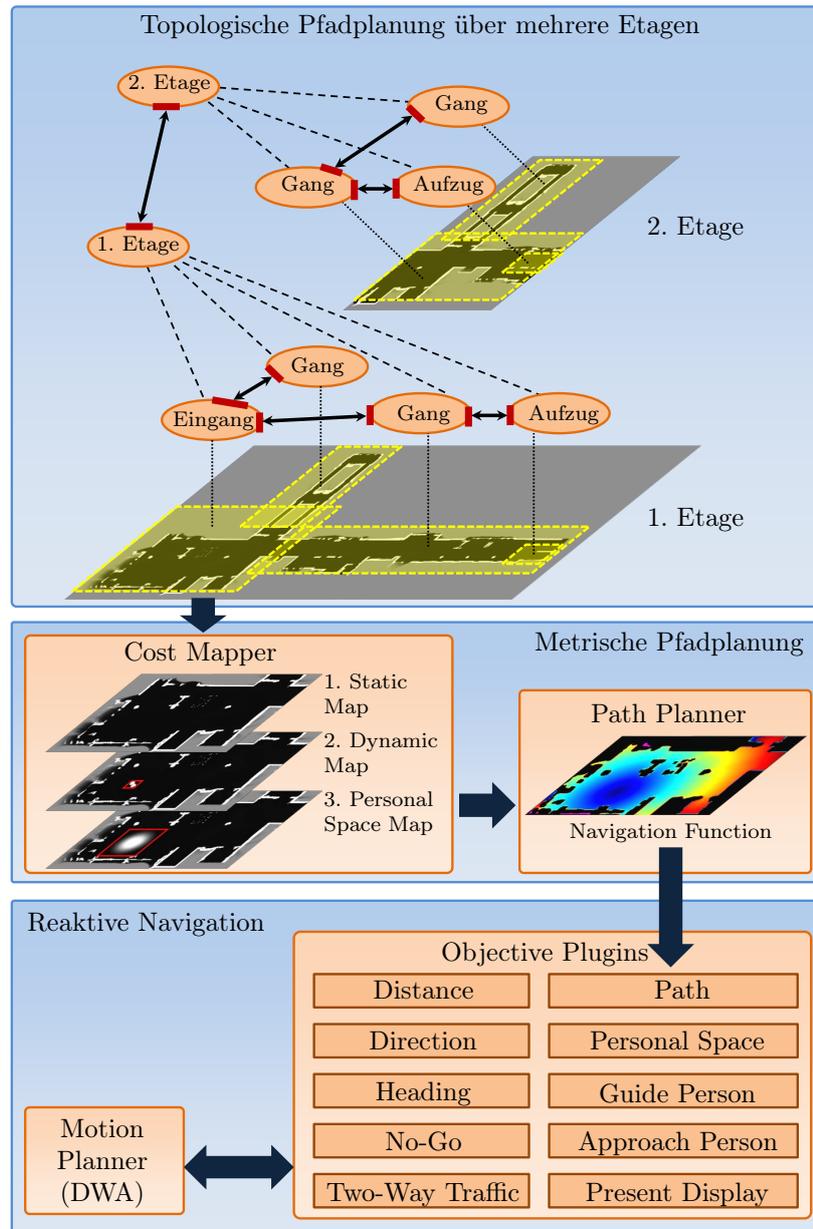


Abbildung 3.1. Navigationskomponenten der drei Planungsebenen: Strategische (topologische Planung), taktische (metrische Planung) und reaktive Navigationsebene zur Umsetzung der Navigationsverhalten. Diese Abbildung stellt eine detailliertere Sicht der Navigationsmodule aus Abb. 2.2 dar. (Abbildung im Rahmen dieser Arbeit auch für [Gross, 2014] angefertigt)

Millisekunden auf die aktuellen Sensordaten und das Umweltmodell reagiert wird. Da somit nur sehr begrenzte Rechenzeit zur Verfügung steht, kann bei der reaktiven Bewegungsplanung nur ein Planungshorizont von wenigen Sekunden berücksichtigt werden.

Um zu weiter entfernten Zielen zu navigieren, erfolgt eine langfristige Bewegungsplanung auf *taktischer Ebene*. Das Ergebnis dieser Bewegungsplanung ist eine Navigationsfunktio-

on, welche von einem *Path Objective* verwendet wird und so in die reaktive Bewegungsplanung einfließt. Da die Beobachtung der Umwelt unvollständig ist und sich deren Dynamik nicht sicher vorhersagen lässt, kann kein konkreter Pfad bis zum Ziel geplant werden. Deshalb liefert die Navigationsfunktion für die Gridzellen der Karte die Länge des kürzesten Weges zum Ziel. Damit verliert die Navigationsfunktion nur ihre Gültigkeit, wenn neue Beobachtungen dynamischer Hindernisse vorliegen. In diesen Situationen müssen die entsprechenden Teilbereiche der Funktion aktualisiert werden. Die tatsächlich gefahrene Trajektorie des Roboters, welcher von den verschiedenen Objectives beeinflusst wird, hat keinen Einfluss auf die Gültigkeit der Navigationsfunktion. Aufgrund der längeren Geltungsdauer der Navigationsfunktion steht für die taktische Bewegungsplanung auch mehr Rechenzeit zur Verfügung.

Zusammengefasst wird auf taktischer Ebene die metrische Pfadplanung zu bestimmten Zielpositionen durchgeführt, und auf reaktiver Ebene findet die tatsächliche Festlegung der unmittelbaren Bewegungskommandos unter Berücksichtigung verschiedener Zielstellungen statt. Es fehlt folglich noch eine Instanz, welche zu jedem Zeitpunkt die relevanten Sub-Ziele (Zielstellungen und Zielpositionen) des Roboters im Kontext der Gesamtapplikation bestimmt. Auf *strategischer Ebene* wird im Kontext der globalen Navigation mittels topologischer Pfadplanung das nächste Zwischenziel des Roboters ermittelt und der metrischen Pfadplanung übergeben. Außerdem wird durch einen Zustandsautomat das aktuell aktive Behavior (Abb. 2.2) gewählt. Dieses wiederum legt den aktuellen Task bzgl. der Roboternavigation fest. Ein Task besteht aus verschiedenen Sub-Tasks, welche indirekt bestimmen, welche Objectives auf reaktiver Ebene aktiv sind.

3.1.1. Reaktiv: Dynamic Window Approach (DWA)

Die reaktive Steuerung der Roboterbewegung erfolgt mittels „Dynamic Window Approach“ (DWA) [Fox, 1997]. Das wesentliche Merkmal dieses Verfahrens ist die Berücksichtigung der Roboterdynamik. Ausgehend von der aktuellen Geschwindigkeit des Roboters werden alle erreichbaren Geschwindigkeitskommandos des Roboters ermittelt. Die durchführbaren Steuerungen sind limitiert durch die physikalisch erreichbaren Geschwindigkeiten und Beschleunigungen des Roboters. Innerhalb des verbleibenden Raumes an Geschwindigkeitskommandos, welche das dynamic Window bilden, wird die optimale Geschwindigkeit gesucht. Dazu wird das Fenster diskretisiert, und die diskreten Geschwindigkeiten werden bewertet. Im Gegensatz zu [Fox, 1997] wird in dieser Arbeit der Raum der möglichen Geschwindigkeiten nicht äquidistant abgetastet. Stattdessen ist die Abtastdichte des Fensters in der Nähe der aktuellen Geschwindigkeit des Roboters höher und nimmt mit wachsender Geschwindigkeitsänderung ab. Damit spiegelt die Abtastdichte

die Wahrscheinlichkeit wider, dass sich das optimale Bewegungskommando in der Nähe der aktuellen Geschwindigkeit des Roboters befindet, und es werden entsprechend feine Anpassungen der Geschwindigkeit ermöglicht. Diese Abtastung des dynamic Window und die entsprechenden Trajektorien, welche der Roboter bei den jeweiligen Kommandos fahren würde, sind in Abb. 3.2 dargestellt.

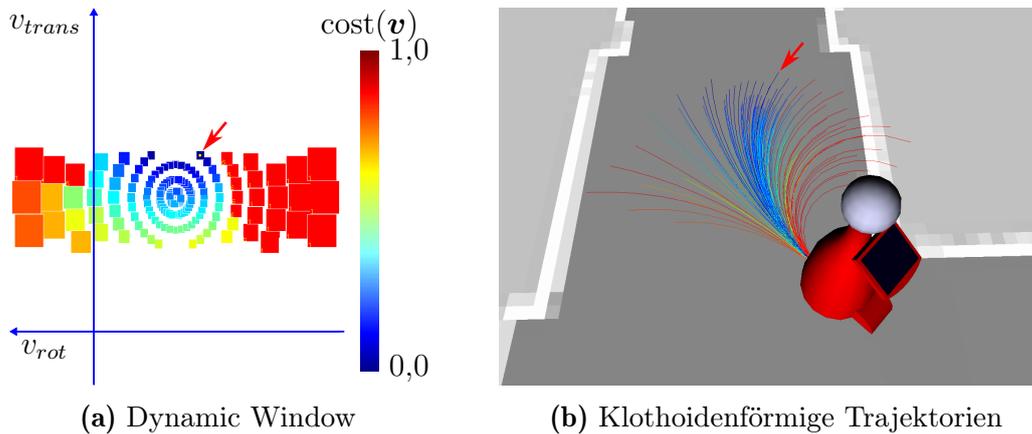


Abbildung 3.2. Adaptive dynamic Window: Das dynamische Fenster der durch den Roboter zukünftig erreichbaren Geschwindigkeiten (a) wird in der Nähe zur aktuellen Geschwindigkeit sehr dicht abgetastet und mit wachsender Geschwindigkeitsänderung reduziert sich die Abtastdichte. Das Geschwindigkeitskommando mit den niedrigsten Kosten ist durch den roten Pfeil gekennzeichnet. In (b) sind die abgetasteten, klothoidenförmigen Trajektorien des dynamic Window dargestellt. Die Geschwindigkeitskommandos, welche den Roboter mit der Wand kollidieren lassen würden, haben die Kosten 1,0 (rot). Für geeignete Geschwindigkeitskommandos werden niedrige Kosten (blau) berechnet.

Zur Bewertung eines potenziellen Geschwindigkeitskommandos werden durch die verschiedenen Objectives Kosten berechnet, welche der Verletzung der jeweiligen Zielstellung entsprechen. Die Geschwindigkeit, deren gewichtete Überlagerung dieser Kosten am niedrigsten ist, wird tatsächlich angewendet. In der ursprünglichen Veröffentlichung [Fox, 1997] wurden die Drehung des Roboters relativ zu einer Zielposition, der Abstand zu Hindernissen und die translatorische Geschwindigkeit des Roboters bewertet. Dadurch wird erreicht, dass sich der Roboter in Umgebungen ohne „Sackgassen“ möglichst schnell in Richtung Ziel bewegt, ohne mit Hindernissen zu kollidieren. Die in dieser Arbeit verwendeten Zielstellungen unterscheiden sich von denen aus [Fox, 1997] und sind in Tabelle 3.1 aufgeführt.

Um ein Geschwindigkeitskommando zu bewerten, wird die klothoidenförmige Bewegung des Roboters entsprechend des Kommandos über einen Zeitraum von wenigen Sekunden prädiert. Für diese Trajektorie werden die Kostenfunktionen der aktiven Objectives berechnet. Der Prädiktionshorizont Δt_{pred} hängt von dem jeweiligen Objective und der Robotergeschwindigkeit ab. So sollte er beispielsweise für das *Distance Objective*, welches

Tabelle 3.1. Objectives und Verhaltensweisen, welche diese aktivieren: Übersicht über die in dieser Arbeit verwendeten und **bearbeiteten** (durch Fettschrift hervorgehoben) Objectives. Zu jedem Objective sind dessen Gewicht während der Experimente (Kapitel 7) und die Verhaltensweisen angegeben, welche das Objective aktivieren. Zu den bearbeiteten Objectives ist jeweils die Verhaltensweise (mit Kapitelangabe) hervorgehoben, in dessen Kontext das Objective beschrieben wird.

	Objective, Gewicht	Aktivierende Verhaltensweisen	Beschreibung
Personenunabhängig	Distance $\alpha_{Dis} = 1$	alle	Verhinderung von Kollisionen, durch Bewertung der Distanz zwischen Hindernissen und zukünftigen Roboterpositionen.
	Direction $\alpha_{Dir} = 50$	alle	Verhindert, dass der Roboter längere Strecken rückwärts zurück legt. Weiteres Rückwärtsfahren wird in Abhängigkeit von der bis dahin rückwärts zurückgelegten Strecke immer schlechter bewertet.
	Heading $\alpha_{Hea} = 1$	Lotsen, Kontaktaufnahme, Zielanfahrt, Wegfreigabe	Einnehmen einer gewünschten Orientierung durch den Roboter, sobald dieser den gewünschten Zielbereich erreicht hat.
	No Go $\alpha_{NG} = 1$	alle	Es wird verhindert, dass der Roboter in bestimmte, in einer No-Go Map vordefinierte, Bereiche fährt.
	Two-Way Traffic $\alpha_{TWT} = 1$	Respektvolle Navigation (3.3) beim Lotsen und Zielanfahrt	Nach Möglichkeit wird auf der in Fahrtrichtung linken Seite des Roboters Freiraum für entgegenkommende Personen gelassen.
Personenfokussiert	Path $\alpha_{Pat} = 1$	Begleiten (3.5), Kontaktaufnahme (3.2) und alle Anderen	Annäherung an eine (möglicherweise dynamische) Zielposition.
	Personal Space $\alpha_{PS} =$ 1.000	Respektvolle Navigation (3.3) bei allen Verhaltensweisen	Wahrung des Personal Space von sich bewegenden und stehenden Personen.
	Guide $\alpha_{Gui} =$ 10.000	Lotsen (3.4)	Verhindert, dass der Roboter einer geführten Person zu weit vorausfährt.
	Present Display $\alpha_{PD} = 10$	Kontaktaufnahme (3.2)	Drehung des Displays (Rumpf) und Kopfs des Roboters zu interessierten Personen.

die Distanz zu Hindernissen bewertet, um Kollisionen zu verhindern, so groß sein, dass der Roboter innerhalb des Prädiktionshorizontes zum Stillstand kommen kann. Die zukünftige Roboterbewegung wird für den Prädiktionshorizont durch eine Klothoide approximiert. Gewöhnlich tasten die Objectives diskrete Punkte der Klothoide ab und berechnen auf Basis dieser Roboterpositionen die Höhe der Kosten, welche der Verletzung der Zielstellung entsprechen. Im Gegensatz dazu bewerten andere Objectives direkt die Geschwindigkeitsregelungen, da so bzgl. mancher Zielstellungen eine effizientere Berechnung möglich ist. Neben der Bewertung von Geschwindigkeitskommandos mit kontinuierlichen Werten können sicherheitskritische Objectives Geschwindigkeiten, welche beispielsweise den Roboter mit einem Hindernis kollidieren lassen, direkt ausschließen. Das entspricht in etwa einer Bewertung mit unendlichen Kosten. So bewertet das *Distance Objective* Kommandos, welche den Roboter in die Nähe von Hindernissen fahren lassen, mit kontinuierlich wachsenden Kosten umso geringer der Abstand zum Hindernis ist. Die Geschwindigkeitskommandos, welche den Roboter mit einem Hindernis kollidieren lassen würden, werden gänzlich ausgeschlossen. Entsprechend ergibt sich für die Objectives, welche danach die Geschwindigkeitskommandos bewerten, ein reduziertes dynamic Window, und der Berechnungsaufwand sinkt.

Anforderungen an die Objectives

Wie zuvor beschrieben, verwendet der DWA zur Verrechnung der einzelnen Objectives zu einem Gesamtverhalten jeweils ein Gewicht für jedes Objective. Die Gewichte werden auch in dieser Arbeit manuell gewählt, und deshalb sollte das Gesamtverhalten des Roboters nicht zu sensitiv bzgl. der Parameterwahl sein, weil sonst eine aufwändigere Optimierung der Parameter erforderlich wäre. Vielmehr dienen die Gewichte dazu, eine Hierarchie der Objectives festzulegen.

Um dies zu erreichen, sollten die Objectives die Geschwindigkeitskommandos entweder mit Kosten innerhalb eines festen Wertebereichs $[0.0, 1.0]$ bewerten oder die Kommandos gänzlich verbieten. Damit die weniger stark gewichteten Objectives auch Einfluss auf das Gesamtverhalten nehmen können, sollten die Kostenfunktionen aller Objectives diesen Wertebereich vollständig ausnutzen. Die Kostenfunktion jeder einzelnen Zielstellung sollte im tolerierbaren Bereich nur gering ansteigen, beim Übergang zu kritischen Werten dann jedoch stark ansteigen. Auf diese Weise haben die weniger relevanten Objectives bei tolerierbaren Geschwindigkeiten der relevanteren Objectives einen Einfluss auf das Roboterverhalten, ohne dass geringe Änderungen an den Gewichten das Gesamtverhalten wesentlich verändern.

Damit der Roboter ein gleichmäßiges Fahrverhalten aufweist, sollten die Kostenfunktionen sowohl über die Zeit als auch über die potenziellen Geschwindigkeiten stetig sein.

3.1.2. Taktisch: Pfadplanung mit dem E*-Algorithmus

Da die Bewegungsplanung auf reaktiver Ebene nur einen Prädiktionshorizont von wenigen Sekunden betrachtet, ist zum Erreichen weiter entfernter Ziele eine zusätzliche Planungsinstanz notwendig. Auf taktischer Ebene wird eine Navigationsfunktion über die metrische Karte der Einsatzumgebung des Roboters berechnet. Diese Navigationsfunktion steigt ausgehend von der gewählten Zielposition kontinuierlich an und kann somit auf reaktiver Ebene durch das *Path Objective* zum Erreichen des Zieles genutzt werden. Dazu bewertet das *Path Objective* die Geschwindigkeitskommandos bzgl. deren Gradientenabstieg auf der Navigationsfunktion, was gleichbedeutend mit einer Annäherung an das Ziel ist. Würde sich der Roboter mit maximaler Geschwindigkeit in Richtung Ziel bewegen, so würde das *Path Objective* das entsprechende Bewegungskommando mit den Kosten 0,0 bewerten. Führt ein Bewegungskommando dazu, dass sich der Roboter mit Maximalgeschwindigkeit vom Ziel entfernt, so werden diesem Kommando die Kosten 1,0 zugeordnet. Die Navigationsfunktion wird ausgehend von einer metrischen Karte $M_{\text{travel_costs}}$ mittels E*-Algorithmus [Philippsen, 2005] berechnet. Die Werte in den Gridzellen von $M_{\text{travel_costs}}$ repräsentieren die Kosten bzw. das Risiko beim Durchfahren der jeweiligen Zelle mit dem Roboter. Diese Kosten hängen z. B. von Hindernissen bzw. der Nähe zu diesen ab. Die Navigationsfunktionen, welche mittels E*-Algorithmus berechnet werden, zeichnen sich (im Gegensatz zum A* und D*-Algorithmus [Hart, 1968, Stentz, 1994]) dadurch aus, dass der Gradientenabstieg auf diesen Funktionen zu besonders glatten Trajektorien führt. Darüber hinaus ist die dynamische Neuberechnung der Navigationsfunktion bei Änderung der zugrunde liegenden Kostenfunktion $M_{\text{travel_costs}}$ effizient möglich. Denn, wie auch beim D*-Algorithmus, werden nur die von der Änderung betroffenen Teilbereiche der Navigationsfunktion aktualisiert.

Modulare Generierung der Kostenkarte Damit die Kostenkarte $M_{\text{travel_costs}}$ neben der Distanz zu Hindernissen auch soziale und andere Kriterien in die Pfadplanung einbringen kann, wurde ein modulares mehrstufiges System zur Manipulation der Kostenkarte entwickelt (siehe Cost Mapper in Abb. 3.1). In jeder Stufe manipuliert ein entsprechendes Modul die relevanten Bereiche der Kostenkarte. Ab der zweiten Stufe wird die Kostenkarte der entsprechenden Stufe durch lokale Veränderung der Kostenkarte der untergeordneten Stufe erzeugt. Damit die Stärke des E*-Algorithmus' im Umgang mit dynamischen Veränderungen der Kostenkarte genutzt werden kann, ist es erforderlich, dass bekannt ist, welche Bereiche der endgültigen Kostenkarte über die Zeit verändert wurden. Zu diesem Zweck speichert das Framework für jede Stufe die aktuelle Kostenkarte. Dadurch brauchen in den einzelnen Stufen zu jedem Zeitschritt anstatt der gesamten Kostenkarte

nur die Bereiche neu berechnet zu werden, welche sich gegenüber dem letzten Zeitschritt verändert haben. Dies betrifft neben den Bereichen, in welchen sich bzgl. des Kriteriums der Stufe etwas verändert hat, auch die Bereiche, welche durch die niedrigeren Stufen verändert wurden. Deshalb wird von jeder Stufe zur übergeordneten Stufe eine Liste von Dirty Regions übergeben. Die endgültige Liste der Dirty Regions wird gemeinsam mit der aktuellen Kostenkarte durch den E*-Algorithmus verwendet, um nur die betreffenden Bereiche der Navigationsfunktion zu aktualisieren.

3.1.3. Strategisch: Festlegung von Teilaufgaben und Zwischenzielen

Wie in den letzten beiden Abschnitten beschrieben, können auf reaktiver und taktischer Ebene der Bewegungsplanung verschiedene Zielstellungen verfolgt werden. Manche dieser Zielstellungen können in Kombination erreicht werden, andere stehen im Konflikt zueinander. Im Gesamtkontext der Applikation ist es deshalb notwendig festzulegen, welche Subziele zu einem bestimmten Zeitpunkt verfolgt werden sollen. Diese Festlegung findet auf strategischer Ebene in Form von Tasks statt. Ein Task besteht aus einer Menge von Subtasks, und es sind immer nur die Objectives aktiv, welche zum Erreichen eines aktuellen Subtasks dienen. Wenn ein Subtask darin besteht, dass der Roboter zu einer bestimmten Position fährt, werden außerdem auf strategischer Ebene mittels topologischer Pfadplanung die Zwischenziele festgelegt, welche auf taktischer Ebene durch die metrische Pfadplanung verfolgt werden. Wie in Abschnitt 2.6 beschrieben, erfolgt die Generierung der Tasks durch die Behaviors.

In den nächsten Abschnitten wird die Umsetzung der im Rahmen dieser Arbeit bearbeiteten Verhaltensweisen beschrieben. Zu jeder Verhaltensweise werden die Objectives erläutert, welche im Rahmen dieser Arbeit entwickelt wurden und durch das entsprechende Behavior aktiviert werden. Darüber hinaus werden im Kontext der Verhaltensweise zum *Begleiten von Personen* die notwendigen Veränderungen bzgl. der taktischen Ebene des Navigationsstacks erläutert. Der Zusammenhang zwischen den Objectives und den Behaviors, welche diese aktivieren, ist in Tabelle 3.1 gezeigt.

3.2. Gezielte Kontaktaufnahme

Bevor der Roboter Kontakt mit einer Person aufnimmt, wird basierend auf der Oberkörperorientierung (Abschnitt 4.2) das Interesse der anwesenden Personen an dem Roboter

geschätzt. Ist eine Person längere Zeit dem Roboter zugewandt, signalisiert auch dieser seine Interaktionsbereitschaft (Abschnitt 2.2), indem er seinen Kopf in Richtung Person dreht, dieser zuzwinkert und sie anspricht. Danach wird das *Present Display Objective* aktiviert, und der Roboter dreht sich so, dass der Person das Display zugewandt ist. Das *Present Display Objective* ermittelt die Roboterpose zum Ende des Prädiktionshorizonts entsprechend des jeweiligen Geschwindigkeitskommandos. Dann wird die Winkelabweichung zwischen Displayorientierung und der Richtung, in welcher sich die Person befindet, zur Bewertung der Winkelgeschwindigkeit verwendet. Je nach Situation innerhalb der Gesamtanwendung kann ab einem bestimmten Interaktionsinteresse an die Person herangefahren werden.

In dieser Arbeit wird das Heranfahen an Personen vor allem im Kontext des Anwendungsszenarios Rehabilitationsassistent bearbeitet. Im Gegensatz zum Szenario Besucherinformationssystem ist hier der Roboter für längere Zeiträume einem bestimmten Nutzer zugeteilt. Aus diesem Grund wird interessierten Zuschauern nur der Kopf zugewandt und zugezwinkert, herangefahren wird ausschließlich an den aktuellen Nutzer. Wie in Abschnitt 2.2 aufgeführt, hängt das Wohlbefinden einer Person, an welche ein Roboter heranfährt, von sehr vielen Faktoren ab. Gegenüber dem Heranfahen an eine frei stehende Person gibt es im ROREAS Szenario jedoch verschiedene Einschränkungen. Wenn der Patient während des Lauftrainings ermüdet und sich auf einen Stuhl setzt, ist es die Aufgabe des Roboters sich so der Person zu nähern, dass diese den Touchscreen erreichen kann, um den weiteren Trainingsablauf zu bestimmen. Da die Stühle in den Ruhebereichen an der Wand aufgereiht sind und die Patienten während des Lauftrainings häufig einen Rollator mit sich führen, sind die möglichen Interaktionspositionen und das Fahrverhalten, um diese zu erreichen, meist stark eingeschränkt. Deshalb liegt beim Heranfahen an eine Person der Fokus dieser Arbeit auf dem Auffinden einer Zielpose, welche dem Patienten das Erreichen des Displays erlaubt und durch den Roboter erreichbar ist. Somit bestehen die Optimierungskriterien bei der Suche einer Interaktionspose in (a) der Erreichbarkeit durch den Roboter, (b) im Abstand und (c) der Orientierung des Roboters relativ zur Person und (d) der Distanz zu Hindernissen. Darüber hinaus werden (e) Zielpositionen bevorzugt deren Abstand zur aktuellen Roboterposition möglichst gering ist.

Da zum Erreichen einer geeigneten Interaktionsposition häufig Hindernisse, wie der bereits erwähnte Rollator, umfahren werden müssen, ist eine reaktive Optimierung auf Basis der klothoidenförmigen Trajektorien nicht ausreichend. Deshalb wird die Interaktionspose durch ein separates Modul ermittelt, und die eigentliche Bewegungsplanung zu diesem Ziel findet durch metrische Pfadplanung (Abschnitt 3.1.2) und das *Path Objective* statt. Die Ermittlung der Zielpose erfolgt mittels Particle Swarm Optimization [Kennedy, 1995]. Die

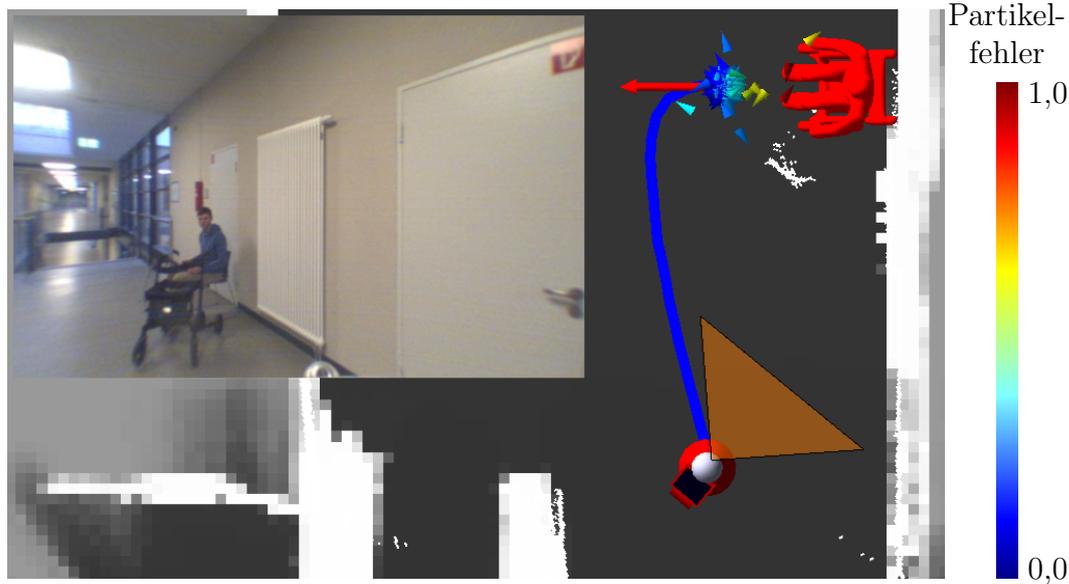


Abbildung 3.3. Ermittlung der Interaktionspose: Dargestellt ist die Vogelperspektive der Karte, in welcher für den Roboter eine geeignete Position zur Interaktion mit dem Nutzer (rot, oben rechts) gesucht wird. Links oben ist das Bild der Frontkamera des Roboters gezeigt. Die Optimierung der Interaktionspose erfolgt mittels Particle Swarm Optimierung. Die Farbe der Dreiecke, welche den bereits konvergierten Partikelschwarm darstellen, stellt den Fehler der Partikel dar. Der rote Pfeil kennzeichnet die optimierte Interaktionspose. Wie der rote Pfeil zeigt, ist der Rumpf des Roboters während der Interaktion von der Person abgewandt, da sich das Display auf der Rückseite des Roboters befindet. Die blaue Linie stellt den geplanten Pfad von der aktuellen Roboterposition zu der Interaktionspose dar.

Kostenfunktion zur Bewertung der Partikel bzgl. der oben genannten Optimierungskriterien ist im Anhang C.1 beschrieben. Abbildung 3.3 zeigt für eine beispielhafte Situation, wie die Partikel um die Interaktionsposition (roter Pfeil) konvergiert sind und welcher Pfad um den Rollator herum zu dieser Zielposition geplant wird.

3.3. Respektvolle Navigation

Im Gegensatz zu leblosen Objekten, welche ausschließlich einen physischen Raum einnehmen, stellen Menschen zusätzlich psychische Ansprüche an ihre Umgebung. Dringt ein anderer Mensch oder ein Roboter in deren beanspruchten Raum ein, so beeinflusst dies das Wohlbefinden der Person [Hall, 1963]. Der beanspruchte Raum dient als eine Art Sicherheitspuffer, dessen Ausdehnung von verschiedenen Faktoren abhängt. So hat die Vertrautheit, das Aussehen und die Geschwindigkeit des Interaktionspartners wesentlichen Einfluss auf die gewünschte soziale Distanz [Sardar, 2012]. Bei der Interaktion mit

verletzlichen Personen, wie beim Anwendungsszenario ROREAS, hat die soziale Distanz besonders starken Einfluss auf deren Wohlbefinden.

Um das Wohlbefinden der Personen in der Umgebung des Roboters nicht zu beeinträchtigen, wird deren soziale Distanz sowohl auf reaktiver als auch auf taktischer Ebene der Bewegungsplanung berücksichtigt.

3.3.1. Reaktives Ausweichen

Im Kontext einer sozialverträglichen Navigation kann das Wohlbefinden einer Person wesentlich durch die Geschwindigkeit und den Abstand des Roboters relativ zur Person beeinflusst werden [Shi, 2008]. Um diesen Zusammenhang auf reaktiver Ebene in das Fahrverhalten des Roboters zu integrieren, wurde im Rahmen dieser Arbeit ein *Personal Space Objective* entwickelt. Entsprechend der nachfolgenden Formel bestimmt dieses Objective für jedes zum aktuellen Zeitpunkt t mögliche Bewegungskommando \mathbf{v} die Kosten $\text{cost}_{PS}(\mathbf{v}, t)$, welche die Verletzung des Personal Space bewerten:

$$\text{cost}_{PS}(\mathbf{v}, t) = \max_{\Delta t_1, \dots, \Delta t_T} \max_{\mathbf{h}_i \in \mathbf{H}} (\text{conf}(\mathbf{h}_i, t) \cdot \text{cost}_{PS}^*(\mathbf{h}_i, t, \Delta t, \mathbf{v})) \quad (3.1)$$

Dazu werden die Zeitpunkte Δt_i des Prädiktionshorizonts Δt_T gesampelt und für jeden Zeitpunkt Δt_i wird für alle Personenhypothesen \mathbf{h}_i die Verletzung deren Personal Space cost_{PS}^* berechnet. Diese Verletzung wird zusätzlich mit der Sicherheit der Hypothese $\text{conf}(\mathbf{h}_i, t)$ gewichtet. In die Berechnung der Kosten fließt der Abstand zwischen Roboter und Person, sowie die Geschwindigkeit des Roboters in Richtung Person ein. Weitere Details zur Berechnung der Personal Space Kosten sind im Anhang in Abschnitt C.2 zu finden. Voraussetzungen für die Ermittlung dieser Kosten ist das Tracking der Personen in der Nähe des Roboters (Kapitel 4) und deren Bewegungsprädiktion (Kapitel 6).

3.3.2. Taktisches Ausweichen stehender Personen

Wie Abb. 3.4c veranschaulicht, würde die Wahrung des Personal Space auf rein reaktiver Ebene unter Umständen dazu führen, dass der Roboter eine Zielposition nicht mehr erreichen kann, obwohl genug Freiraum zur Umfahrung der Person zur Verfügung stünde. Der Gradientenabstieg auf der geplanten Navigationsfunktion würde direkt zwischen Roboter und Wand zum Ziel führen, und entsprechend würde das *Path Objective* diesen Weg positiv bewerten. Dieser Weg würde jedoch aufgrund des *Personal Space Objective* nicht

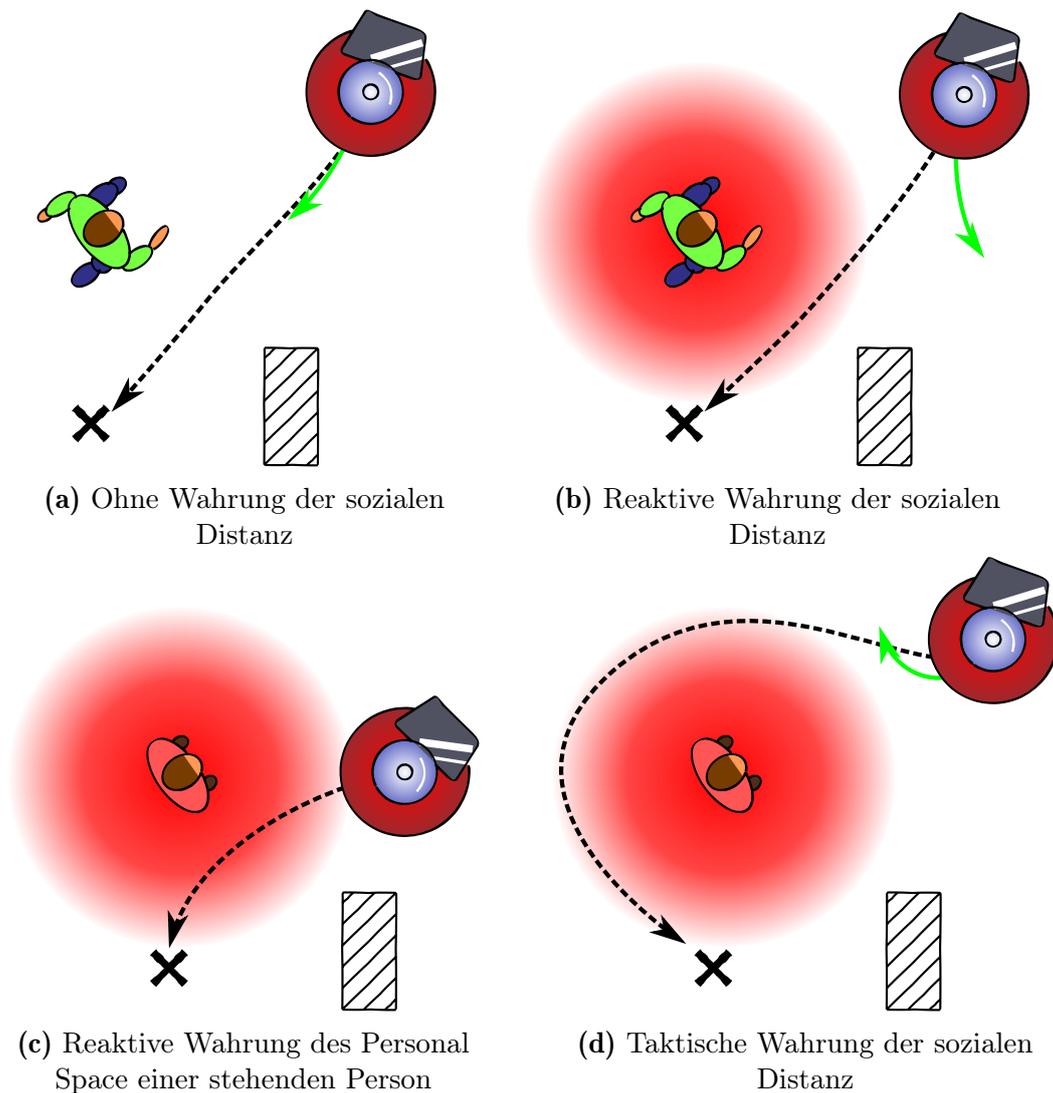


Abbildung 3.4. Reaktives und taktisches Ausweichen: Skizzierung der Auswirkungen des reaktiven und des taktischen Ausweichens. (a) Ohne Wahrung der sozialen Distanz würde der Roboter an Personen genauso nah vorbeifahren, wie an leblosen Hindernissen. (b) Mittels *Personal Space Objective* wird verhindert, dass der Roboter in den Personal Space der Personen eindringt. (c) Die rein reaktive Wahrung des Personal Space steht möglicherweise im Konflikt mit dem Erreichen einer Zielposition. (d) Wird der Personal Space stehender Personen auch bei der langfristigen Pfadplanung berücksichtigt, so wird dieser Konflikt aufgelöst und um die Person herum gefahren.

gefahren werden. Eine Umfahrung der Person würde wiederum durch das *Path Objective* negativ bewertet. Dies würde dann dazu führen, dass der Roboter letztlich stehen bleiben würde. Bei sich bewegenden Personen ist dies unproblematisch, da sich diese Situation mit der Bewegung der Person von selbst auflösen würde. Es wäre jedoch weniger respektvoll, wenn auch von stehenden Personen erwartet würde, dass diese dem Roboter den Weg freigeben. Deshalb ist es zumindest bei stehenden Personen notwendig, dass auch

deren Personal Space während der metrischen Pfadplanung berücksichtigt wird.

Zu diesem Zweck ergänzt die 3. Stufe des Cost Mappers (Abbildung 3.1) den Personal Space stehender Personen in der Kostenkarte M_{travel_costs} . Bei allen getrackten Personen H , deren absolute Geschwindigkeit unter einem bestimmten Grenzwert v_{still} liegt, wird eine gaußförmige Kostenfunktion¹ zur Repräsentation ihres Personal Space in der Kostenkarte ergänzt.

Die prädizierten Trajektorien sich bewogender Personen werden hingegen bei der globalen Pfadplanung in dieser Arbeit nicht berücksichtigt. Verfahren wie [Bennewitz, 2003, Bennewitz, 2005, Göller, 2010], welche die zeitliche Veränderung einer dynamischen Umgebung während der globalen Pfadplanung berücksichtigen, benötigen die Prädiktion der Personenbewegung über den Berechnungszeitraum der Pfadplanung plus den Zeitraum, welcher für das eigentliche Ausweichen der Person notwendig ist. Die Unsicherheit dieser langfristigen Prädiktion der Personenbewegung würde bei den Anwendungsszenarien dieser Arbeit häufig so groß werden, dass für den Roboter nicht mehr ausreichend Freiraum zur Planung kollisionsfreier Pfade zur Verfügung stünde. Dieses Problem ist in der Literatur als Freezing Robot Problem [Trautman, 2010] beschrieben. Günstigerweise besteht bei sich bewogenden Personen weniger das Problem, dass diese aufgrund des physisch beanspruchten Raumes längere Zeit den kürzesten Pfad zum Ziel des Roboters blockieren. Denn nachdem der Roboter aufgrund des reaktiven *Personal Space Objectives* den Weg der sich bewogenden Person frei gegeben hat, wird sich die Person erwartungsgemäß weiter bewegen und so könnte danach auch der Roboter weiter fahren. Zwar ist es an Engstellen möglich, dass der Roboter nicht mehr durch ein rein reaktives Verhalten den Weg freigeben kann, jedoch sollen solche Situationen ohnehin frühzeitig durch eine explizite Situationserkennung detektiert werden. Die Situationserkennung ist jedoch nicht Bestandteil dieser Arbeit.

3.3.3. Reaktives Einhalten des Rechtsverkehrs

Ein weiterer Aspekt der respektvollen Navigation ist die Befolgung des Rechtsverkehrs. Untersuchungen in amerikanischen Shopping-Malls haben gezeigt, dass Personen in solchen Einrichtungen auf der rechten Seite der Gänge laufen [Bitgood, 2006]. Wenn auch der Roboter diesem Bewegungsmuster folgt, können somit Konfliktsituationen präventiv reduziert werden, und das Verhalten des Roboters ist einfacher für die Personen vorherzusagen.

¹In Abschnitt C.3 werden die Auswirkung der Parametrierung der Kostenfunktion behandelt

Zu diesem Zweck wurde das *Two-Way-Traffic Objective* entwickelt, mit dem Ziel wenn möglich einen linksseitigen Mindestabstand zwischen Roboter und Hindernissen einzuhalten. Das Objective orientiert sich dabei an dem direkten Pfad zum Ziel (Abb. 3.5) und ermittelt davon ausgehend einen angepassten Pfad, dessen Abweichung vom ursprünglichen Pfad entlang des ursprünglichen Pfades linear steigt. Die Parametrierung des korrigierten Pfades erfolgt entsprechend durch einen Korrekturfaktor θ_{TWT} .

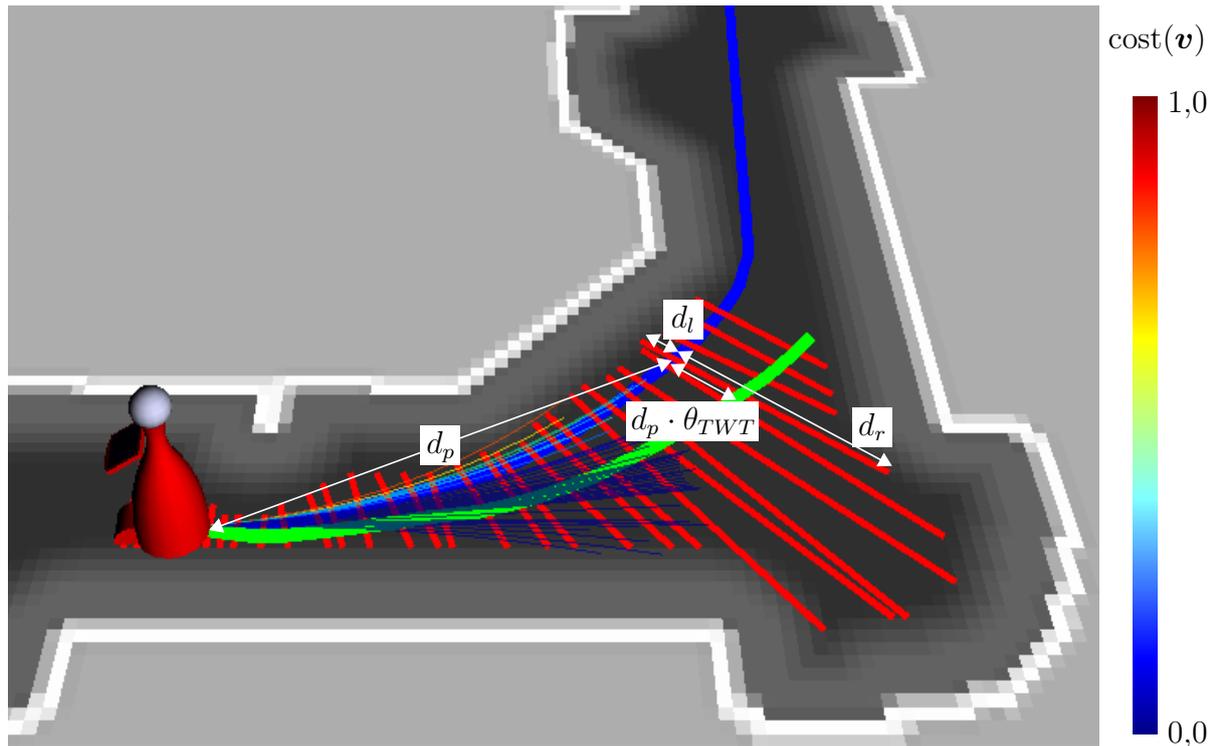


Abbildung 3.5. Anpassung der Roboterbewegung entsprechend des Rechtsverkehrs: Ohne *Two-Way-Traffic (TWT) Objective* würde der Roboter dem direkten Pfad zum Ziel (blau) folgen. Das *TWT Objective* ermittelt den linksseitigen d_l und rechtsseitigen d_r Abstand zwischen dem Pfad und den Hindernissen entlang der Orthogonalen des Pfades (rot). Basierend auf dem zur Verfügung stehenden Freiraum wird ein angepasster Pfad (grün) bestimmt. Dieser Pfad ist definiert durch einen Korrekturfaktor θ_{TWT} , welcher angibt wie sehr der angepasste Pfad (grün) mit wachsender Distanz zum Roboter d_p von dem ursprünglichen Pfad (blau) abweicht. Die Trajektorien des DWA, welche auf die linke Seite des angepassten Pfades führen, werden durch das *TWT Objective* mit höheren Kosten bewertet.

Wie in Abbildung 3.5 dargestellt, werden zur Ermittlung des Korrekturfaktors θ_{TWT} die nächsten Meter des geplanten Pfades (blau) abgetastet, und es wird jeweils der links- d_l und rechtsseitige d_r Abstand zum nächsten Hindernis orthogonal zum Pfad ermittelt. Auf Basis dieser Werte wird für jede Orthogonale bestimmt, wie weit der Roboter maximal

nach rechts vom Pfad abweichen kann, ohne mit einem Hindernis zu kollidieren. Darüber hinaus wird ermittelt, wie weit er nach rechts von dem Pfad abweichen müsste, um den linksseitigen Mindestabstand einzuhalten. Basierend auf diesen Werten wird über alle betrachteten Orthogonalen der Korrekturfaktor θ_{TWT} berechnet. Genauere Angaben dazu sind in Abschnitt C.4 zu finden. Ein wesentlicher Zielkonflikt dieser Verhaltensweise besteht in der Ausnutzung von rechtsseitigen Freiräumen und einer geradlinigen Fahrweise. So soll der Roboter beispielsweise nicht in kurze Ausbuchtungen oder Seitengänge hinein fahren, und Kurven wie in Abb. 3.5 sollen als gleichmäßiger Bogen gefahren werden. Andererseits sollen längere rechtsseitige Freiräume zur Wahrung des Rechtsverkehrs genutzt werden. Dieser Zielkonflikt wird durch die Länge des angepassten Pfades bestimmt. Je länger er ist, desto weniger werden kurzfristige Freiräume ausgenutzt und umso ähnlicher ist der Pfad dem effizienten Ursprungspfad.

Nachdem der Korrekturfaktor bekannt ist, wird dieser zur Bewertung der Bewegungskommandos \mathbf{v} genutzt, um letztlich die gewünschte Anpassung gegenüber dem ursprünglichen Pfad zu erreichen. Auch dies ist genauer im Anhang C.4 beschrieben.

3.4. Personenfokussiertes Lotsen

Das Lotsenverhalten eines mobilen Roboters besteht im Wesentlichen darin, eine Zielposition anzufahren und dadurch einem menschlichen Interaktionspartner den Weg zu diesem Ziel zu weisen. Allerdings ist es für das Wohlbefinden der geführten Person entscheidend, dass der Roboter während der Lotsenfahrt sein Fahrverhalten an die gelotste Person anpasst. Die Person soll sich weder durch einen zu schnell fahrenden Roboter unter Druck gesetzt fühlen, noch durch einen zu langsamen Roboter aufgehalten werden. Das Ziel des entwickelten *Guide Objective* ist es deshalb, die Geschwindigkeit des Roboters so anzupassen, dass dieser möglichst gleichmäßig mit konstantem Abstand vor der Person fährt.

Das *Guide Objective* schätzt für die möglichen Bewegungskommandos \mathbf{v} den zukünftigen Abstand d_{pred} zwischen Person und Roboter zum Ende des Prädiktionshorizonts Δt_T . Dabei wird die zukünftige Position der Person linear prädiziert, und die zukünftige Position des Roboters ergibt sich aus dem jeweiligen Bewegungskommando. Die Bewertung der Bewegungskommandos erfolgt basierend auf dem Fehler zwischen prädiziertem Abstand d_{pred} und gewünschtem Abstand d_G . Dabei werden jedoch nur Bewegungskommandos bestraft, welche dazu führen, dass der Roboter der Person zu weit vorausfährt:

3.5. Begleiten einer Person

Damit ein Roboter eine Person begleiten kann, ist eine Verhaltensweise notwendig, welche die in Bezug auf die Wahrnehmung getrackte Person (Kapitel 4) auch physisch verfolgt. Aus der Bewegung der Person ergibt sich für den Roboter die Aufgabe zu einer dynamisch veränderlichen Zielposition zu fahren. Im Gegensatz zur Navigation zu einer statischen Zielposition erfordert dies, dass die Bewegungsplanung des Roboters aktualisiert wird, während sich dieser bewegt. Die Herausforderungen dabei sind die echtzeitfähige Bewegungsplanung und eine gleichmäßige Fahrweise trotz Änderung des Navigationszieles. Beide Anforderungen ließen sich gut durch ein reaktives Objective erfüllen. Versuche mit einem solchen Objective ergaben jedoch, dass bei der Verfolgung einer Person häufig auch Hindernisse umfahren werden müssen. Wie Abb. 3.7a zeigt, steht in diesen Situationen die reaktive Verfolgung der Person im Konflikt mit der Kollisionsvermeidung des Hindernisses. Das Umfahren der Hindernisse erfordert eine taktische Planung von Pfaden, deren Komplexität über die der reaktiv geplanten Klothoide hinausgeht.

Um dynamische Ziele auf taktischer Ebene zu verfolgen, wurden im Rahmen dieser Arbeit wesentliche Erweiterungen am *Path Objective* und an der metrischen Pfadplanung vorgenommen. In Abschnitt 3.5.1 wird beschrieben, wie trotz dynamischer Ziele eine gleichmäßige Fahrweise erreicht wird. Um ein vorausschauendes Begleitverhalten zu realisieren, wird während der Pfadplanung eine Prädiktion der Personenbewegung berücksichtigt (Abschnitt 3.5.2).

In verschiedenen Arbeiten [Morales, 2014] wird Wert darauf gelegt, dass der Roboter beim Begleiten einer Person eine bestimmte Pose relativ zu der Person einnimmt. Die Verwendung einer solchen Relativpose zur prädizierten Personenposition wird auch durch das in dieser Arbeit entwickelte Verfahren unterstützt. Bei der praktischen Anwendung wurde die Relativpose jedoch so parametrisiert, dass die Person selbst das Ziel der Pfadplanung darstellt. Das *Personal Space Objective* verhindert, dass der Roboter zu nah an die Person heranzfährt, und so nimmt der Roboter eine beliebige Position am Rande des Personal Space der Person ein. Die Festlegung einer bestimmten Relativpose würde, insbesondere bei den belebten Gängen im Anwendungsszenario ROREAS, die ohnehin eingeschränkten Bewegungsmöglichkeiten des Roboters weiter reduzieren.

In [Gockley, 2007] wurde eine Pfadverfolgung, bei welcher der Roboter dem Pfad der begleiteten Person folgt, und eine Richtungsverfolgung, bei welcher der Roboter möglichst direkt zur Person fährt, untersucht. Die Richtungsverfolgung wurde dabei von den Probanden als natürlicher empfunden. In [Gockley, 2007] wird jedoch auch angemerkt, dass in Einsatzumgebungen, in welchen der Roboter nicht alle Hindernisse selbst wahrnehmen

kann, ein hybrides Verfahren empfehlenswert sei. Denn dadurch wird verhindert, dass der Roboter einen gänzlich anderen Weg als die Person zurück legt. Ein solches Verfahren wurde in [Bohlmann, 2012] für einen Outdoor-Roboter implementiert. Dies ist jedoch in den Einsatzumgebungen dieser Arbeit nicht notwendig, und so wird auch aus Effizienzgründen eine Richtungsverfolgung zur prädierten Position der begleiteten Person umgesetzt.

3.5.1. Gleichmäßige Fahrweise durch Double Buffering

Ein wesentlicher Unterschied zur herkömmlichen Pfadplanung ist, dass sich aufgrund der Bewegung der Person die Zielposition immer wieder ändert, während der Roboter bereits die Navigationsfunktion zur Navigation verwendet. Um zu gewährleisten, dass die verwendete Navigationsfunktion konsistent bleibt, wird, wie auch in der Computergrafik, ein Double Buffer verwendet. Es gibt dementsprechend zwei Speicherbereiche für Navigationsfunktionen. Sobald die Navigationsfunktion zwischen Zielposition und Roboter in dem einen Speicherbereich fertig berechnet wurde, wird dieser Speicherbereich zur Navigation genutzt. In dem anderen Speicherbereich wird mit der Berechnung der Navigationsfunktion für das aktualisierte Ziel begonnen. Ist die Berechnung dieser Navigationsfunktion abgeschlossen, wechselt die Nutzung der Speicherbereiche. Werden während der Berechnung der Navigationsfunktion dynamische Hindernisse wahrgenommen, werden die entsprechenden Bereiche der Navigationsfunktion aktualisiert. Entscheidend ist, dass die zur Navigation verwendete Navigationsfunktion in dem anderen Speicherbereich durchgehend stetig ist. Dies wäre nicht gegeben, wenn während der Verwendung durch das *Path Objective* eine Aktualisierung von Teilbereichen durchgeführt würde. Der Wechsel des durch das *Path Objective* verwendeten Buffers ist hingegen unkritisch, da dieser so synchronisiert ist, dass er nicht während der Bewertung der Bewegungskommandos statt findet.

3.5.2. Bewegungsprädiktion zum vorausschauenden Begleiten

Würde die Navigationsfunktion immer zu der Zielposition geplant, an welcher sich die Person aktuell befindet, würde der Roboter gegenüber der Person im Rückstand bleiben. Während der Pfad zu dieser Zielposition geplant wird und der Roboter zu dieser Position fährt, hätte sich die Person bereits weiterbewegt. Da sich die begleitete Person fortwährend in der Nähe des Roboters befindet und andernfalls gar nicht mehr getrackt würde, beträgt die Berechnungszeit der Navigationsfunktion nur Bruchteile einer Sekunde. Entscheidender ist, dass sich die Person in der Zeit, welche der Roboter zum Erreichen

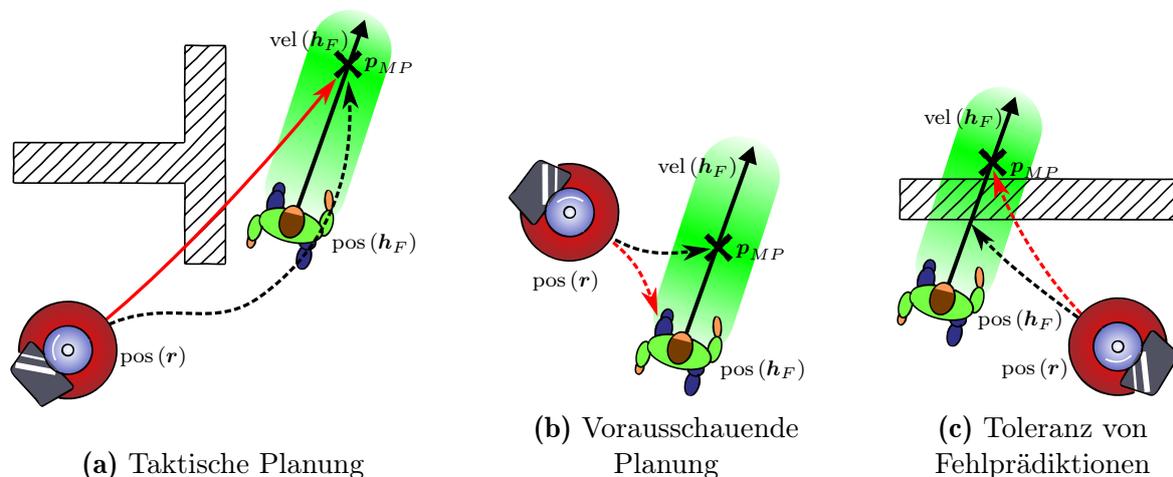


Abbildung 3.7. Aspekte des Begleitens von Personen auf taktischer Ebene: Ausgehend von der aktuellen Personenposition $\text{pos}(\mathbf{h}_F, t)$, deren Geschwindigkeit $\text{vel}(\mathbf{h}_F, t)$, der Roboterposition $\text{pos}(\mathbf{r}, t)$ und der durchschnittlichen Robotergeschwindigkeit wird ein möglicher Treffpunkt \mathbf{p}_{MP} geschätzt. Die Navigationsfunktion des Planungsalgorithmus' wird innerhalb der Zielregion (grün) ausgehend von dem Treffpunkt mit wachsenden Werte initialisiert. (a) Mittels taktischer Pfadplanung zu der Zielregion kann, wie bei statischen Zielen, auch Hindernissen ausgewichen werden. Würden die klothoidenförmigen Trajektorien nur auf reaktiver Ebene bzgl. deren Annäherung an den Treffpunkt bewertet, würde es zu Deadlock Situationen an Hindernissen kommen (\rightarrow). (b) Durch die Vorhersage des Treffpunkts kann der Roboter der Person in geringem Abstand folgen ($--\rightarrow$). Würde bei der Pfadplanung ausschließlich von der aktuellen Position der Person ausgegangen ($---\rightarrow$), hätte sich die Person beim Erreichen der Zielposition bereits weiterbewegt. (c) Die Zielregion des Planungsalgorithmus' (grün) erstreckt sich ausgehend von der aktuellen Position der Person entlang deren prädiizierten Trajektorie. Ist der Treffpunkt, z. B. wegen fehlerhafter Prädiktion, nicht erreichbar ($---\rightarrow$), wird der Roboter in Richtung der erreichbaren Position innerhalb der Zielregion fahren, welche dem Treffpunkt am nächsten ist ($--\rightarrow$).

deren aktuellen Position benötigt, weiterbewegt. Um diese Verzögerungen auszugleichen und unnötige Umwege zu vermeiden, wird die Personenbewegung prädiiziert und eine Art Abfangverhalten (Abb. 3.7b) realisiert. Basierend auf der prädiizierten Trajektorie der Person, der aktuellen Roboterposition und der durchschnittlichen Geschwindigkeit des Roboters wird ein möglicher Treffpunkt \mathbf{p}_{MP} ermittelt. Der Planungsalgorithmus wird nicht mit der aktuellen Personenposition, sondern mit einer speziellen Zielregion initialisiert, von welcher ausgehend die Navigationsfunktion berechnet wird. Diese Zielregion erstreckt sich entlang der prädiizierten Trajektorie der Person, wobei die Werte der Navigationsfunktion innerhalb der Zielregion mit dem euklidischen Abstand zu \mathbf{p}_{MP} steigen. Dadurch wird erreicht, dass der Roboter auf dem schnellsten Weg zum Treffpunkt \mathbf{p}_{MP} fährt, falls dieser erreichbar ist. Bei einer fehlerhaften Prädiktion der Personenbewegung ist es jedoch möglich, dass der prädiizierte Treffpunkt aufgrund von Hindernissen nicht durch den Roboter erreichbar ist. In solchen Situationen soll verhindert werden, dass der

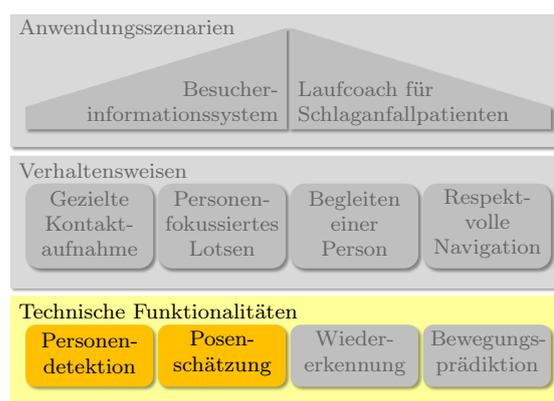
Planungsalgorithmus sehr lange plant, ohne dass die Navigationsfunktion den Roboter erreicht. Aus diesem Grund befindet sich die gesamte Strecke zwischen aktueller Personenposition und prädictiertem Treffpunkt in der Zielregion. So würde zu der erreichbaren Position geplant, welche dem prädictierten Treffpunkt am nächsten ist. Wenn der prädictierte Treffpunkt nicht erreichbar ist, sondern nur die aktuelle Personenposition, so würde der Roboter wenigstens zu dieser Position fahren.

3.6. Zusammenfassung

In diesem Kapitel wurde die Umsetzung von vier Verhaltensweisen zur sozialverträglichen Navigation beschrieben. Für drei Verhaltensweisen (*gezieltes Heranfahren an Personen*, *personenfokussiertes Lotsen* und *Begleiten von Personen*) gibt es ein entsprechendes Behavior, welches die notwendigen Objectives der Bewegungsplanung aktiviert. Darüber hinaus aktiviert jedes dieser drei Behaviors zusätzlich die Objectives zur Realisierung des *respektvollen Navigationsverhaltens*. Bei den sozialen Verhaltensweisen stehen offenbar die menschlichen Interaktionspartner im Fokus. Deren Posen, zukünftigen Bewegungen und die Unterscheidung von Personen haben wesentlichen Einfluss auf das Roboterverhalten. Folglich hängt die Qualität des letztlich realisierten Roboterhaltens sehr stark von den Erkennungsleistungen ab, welche in den nachfolgenden Kapiteln der Arbeit behandelt werden. Bei der Realisierung der Verhaltensweisen wurde, im Hinblick auf ein funktionierendes Gesamtsystem, besonders Wert auf die Toleranz von Fehlern und Unsicherheiten bei der Informationsgewinnung gelegt. Deshalb erfolgt auch die Evaluation der Verhaltensweisen als Gesamtsystem mit den technischen Funktionalitäten zur Modellbildung in Kapitel 7. In den Kapiteln 4 bis 6 werden vorher die einzelnen Module zur Informationsgewinnung über die anwesenden Personen beschrieben und einzeln evaluiert.

4

Personentracking mit Schätzung von Orientierung und Gehhilfe



Um Personen in der Umgebung des Roboters tracken zu können, erfolgt die Detektion von Personen basierend auf den Tiefendaten der beiden Laserscanner und den Bilddaten der omnidirektionalen Kamera. Die Detektionsverfahren für beide Sensorarten werden in den Abschnitten 4.1 und 4.2 beschrieben. Neben der reinen Detektion schätzt das kamerabasierte Verfahren außerdem die Oberkörperorientierung, und der laserbasierte Detektor unterscheidet zusätzlich, ob und welche Gehhilfe eine Person verwendet. In Abschnitt 4.3 wird gezeigt, wie sich die Hypothesen der beiden Detektoren sehr gut ergänzen und wie diese durch einen Tracker fusioniert werden.

4.1. Laserbasierte Personendetektion und Unterscheidung von Gehhilfen

Auf mobilen Robotern werden häufig laserbasierte Personendetektoren verwendet, weil Laserscanner ohnehin auf vielen Robotern verfügbar sind, um zur Lokalisation und Navigation eingesetzt zu werden. Außerdem haben sie eine hohe Aktualisierungsrate und ein großes Sichtfeld. Die Laserdaten sind robust gegenüber Umwelteinflüssen wie z. B. Beleuchtung. Darüber hinaus erlauben die expliziten Tiefendaten mit geringem Messfehler

eine sehr genaue Positionsbestimmung der detektierten Personen, und die Datenmenge der 2D-Laserdaten ist relativ gering.

Andererseits werden von den zu detektierenden Objekten nur Messpunkte innerhalb einer Scanebene wahrgenommen. Somit ist der räumliche Bereich der zu detektierenden Objekte, welcher durch die 2D-Tiefeninformationen beschrieben wird, relativ gering. Außerdem variiert die Erscheinung von Beinpaaren in Tiefendaten stark. Im Anwendungsszenario ROREAS ergibt sich eine weitere Herausforderung daraus, dass in der Rehabilitationsklinik sehr viele Patienten auf eine Gehhilfe angewiesen sind. Dadurch versagen übliche Personendetektoren, welche auf der Erkennung von nicht verdeckten Beinpaaren basieren.

Im nächsten Absatz werden der Forschungsstand zur laserbasierten Personendetektion aufgearbeitet und der Verbesserungsbedarf begründet. Danach wird in Abschnitt 4.1.2 bis 4.1.4 der eigene Beitrag zur laserbasierten Personendetektion beschrieben und in Abschnitt 4.1.7 experimentell untersucht.

4.1.1. Verfahren zur Personendetektion in Laserdaten

Ein wesentliches Kriterium zur Einordnung der Verfahren zur Personendetektion sind die verwendeten Merkmale. So können Verfahren, welche vorrangig auf Bewegungsmerkmalen basieren, und Verfahren, welche stärker die geometrische Gestalt von Personen berücksichtigen, unterschieden werden.

Die Arbeiten, bei welchen eine Ebene durch mehrere stationäre Laserscanner erfasst wird (Abb. 4.1 erste Spalte), basieren vor allem auf Bewegungsmerkmalen. So wird in [Fod, 2002] durch die verteilten Laserscanner ein Hintergrundmodell der statischen Umgebung erzeugt. Alle Messpunkte, welche nicht diesem Hintergrundmodell entsprechen, werden zu Clustern zusammengefasst, wobei Messpunkte dann zu einem Cluster gehören, wenn sie eine geschlossene Oberfläche bilden. Diese Cluster werden mittels Kalman-Filter getrackt, ohne dass deren Geometrie berücksichtigt wird. Im Gegensatz zu [Fod, 2002] sind die Laserscanner in [Zhao, 2005] nicht in Hüfthöhe, sondern in 20 cm Höhe angebracht, wodurch die gegenseitige Verdeckung der Personen reduziert werden soll und Störungen durch Arme und beispielsweise Handtaschen entfallen. Die Vordergrundpunkte werden auch hier geclustert und als einzelne Beinhypothesen getrackt. Jeweils zwei Beinhypothesen werden unter Berücksichtigung des menschlichen Gangmodells zu einer Personenhypothese zusammengefasst. In [Glas, 2007] wird die Erscheinung der Arme im Gegensatz dazu nicht als Störung aufgefasst, sondern die in 90 cm Höhe gescannten Personen werden durch drei Kreise modelliert. Einen großen für den Torso und zwei kleinere für die Arme. Mittels

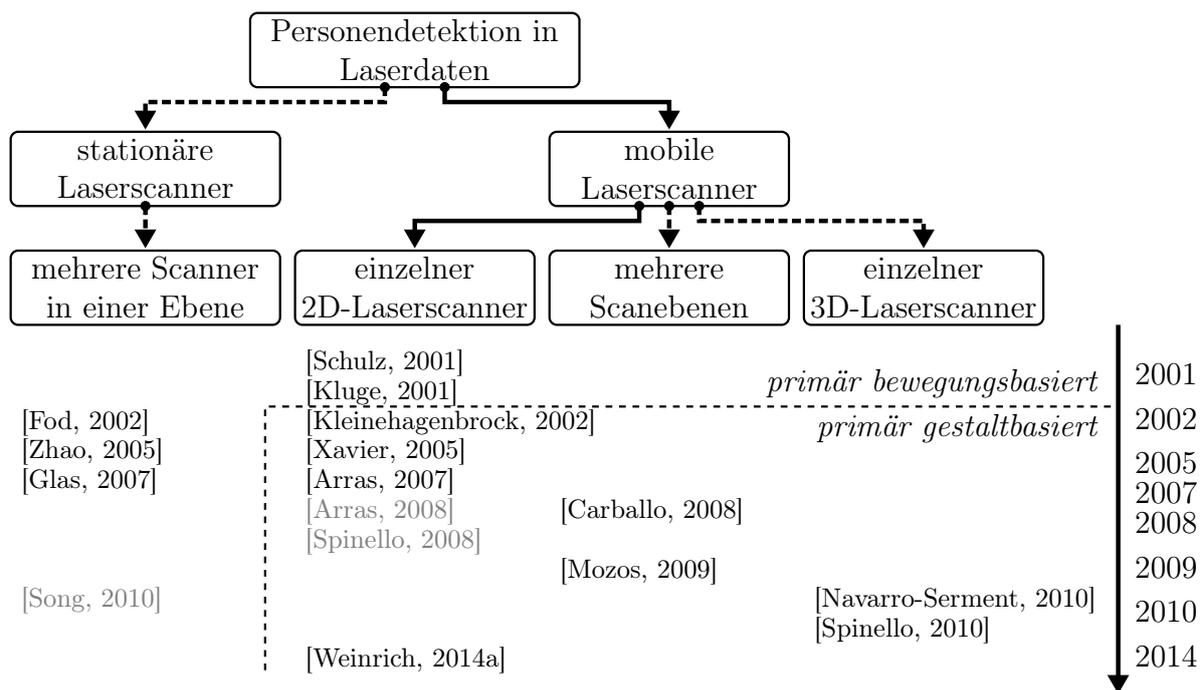


Abbildung 4.1. Arbeiten zu Personendetektion in Laserdaten: Darstellung neuartiger Verfahren (schwarz) und darauf aufbauender Arbeiten (grau) zur Personendetektion in verschiedenen Laseranordnungen nach Jahren

Partikelfilter werden die Positionen und Geschwindigkeiten der Personen getrackt, wobei während der Wichtung der Partikel auch ein Formmodell der relativen Positionen und Durchmesser der drei Kreise verwendet wird. Durch Optimierung der Modellparameter erfolgt auch eine Schätzung der Oberkörperorientierung der Personen.

Die beschriebenen Verfahren funktionieren aufgrund der Verwendung eines Hintergrundmodells jedoch nur bei stationären Laserscannern. Um in den Tiefendaten eines mobilen 2D-Laserscanners (Abb. 4.1 zweite Spalte) potentielle Personenpositionen zu identifizieren, werden stattdessen in [Schulz, 2003] die lokalen Minima im Distanzprofil der Laserscans verwendet. Da diese Minima jedoch durch beliebige Objekte hervorgerufen werden, ist ein Partikelfilter so angepasst worden, dass nur dynamische Objekte getrackt werden. Um dies zu erreichen wird zur Wichtung der Partikel auch eine Differenzkarte der lokalen Umgebung des Roboters verwendet. Diese Karte zeigt Veränderungen der lokalen Umgebung des Roboters zwischen zwei aufeinander folgenden Scans, wobei die Eigenbewegung des Roboters mittels Scan Matching ausgeglichen wird. In [Kluge, 2001] werden die betrachteten Objekte nicht anhand lokaler Minima im Distanzprofil identifiziert, sondern jegliche Tiefensprünge in den Tiefendaten zur Segmentierung der Laserscans verwendet. Zusätzlich wird jedes Segment in konvexe Objekte unterteilt. Zum Matching der Objekte

in aufeinander folgenden Scans werden keine Kalman- oder Partikelfilter, sondern bipartite Graphen verwendet, welche auch die Unterscheidung von statischen und dynamischen Objekten ermöglichen.

Die bisher beschriebenen Verfahren detektieren Personen nur dann, wenn sie sich einmal im Erfassungsbereich des Laserscanners bewegt haben. Darüber hinaus werden Personen nicht explizit von anderen dynamischen Objekten unterschieden. Im Gegensatz dazu werden in [Kleinehagenbrock, 2002] die Segmente der Laserscans anhand nicht näher spezifizierter Schwellwerte als Bein oder Nicht-Bein klassifiziert und die detektierten Beine werden zu Personen gruppiert. Auch in [Xavier, 2005] wird die geometrische Form von Beinen zur Detektion verwendet. So werden in den Tiefendaten Linien, Bögen und Kreise detektiert. Wenn ein Kreis einen Durchmesser zwischen 10 cm und 25 cm hat, wird dieser als Bein klassifiziert. Allerdings variiert die Erscheinung eines Beines in Laserdaten relativ stark, und vordefinierte Merkmale wie der Kreisdurchmesser sind nicht sehr robust. Deshalb wird in [Arras, 2007] mittels AdaBoost ein Klassifikator trainiert, dessen Weak Classifier auf verschiedenen Merkmalen wie Linearität, Kreisähnlichkeit etc. basieren.

Um den Informationsgehalt der Laserdaten und somit die Robustheit der Detektoren zu steigern werden in [Carballo, 2008] und [Mozos, 2009] mehrere Laserscanner (Abb. 4.1, dritte Spalte) verwendet, um verschiedene horizontale Ebenen zu erfassen. In [Mozos, 2009] wird zur Erzeugung von Personenhypothesen, ausgehend von der Detektion einzelner Körperteile in den verschiedenen Ebenen, ein gelerntes Formmodell über die geometrischen Relationen der einzelnen Körperteile verwendet. Statt mehrerer Laserscanner in verschiedenen Ebenen, werden in [Navarro-Serment, 2010, Spinello, 2010] 3D-Laserscanner (Abb. 4.1, vierte Spalte) verwendet. In [Navarro-Serment, 2010] werden mittels PCA [Pearson, 1901] die Hauptkomponenten der zu klassifizierenden Punktwolken ermittelt, und dann wird die Punktwolke auf die Ebene der ersten und zweiten Hauptkomponente, sowie auf die Ebene der ersten und dritten Hauptkomponente projiziert. Die Histogramme über die Punktdichten in diesen beiden Ebenen werden dann mittels Support Vector Machine (SVM) [Cortes, 1995] klassifiziert. In [Spinello, 2010] wird die 3D-Punktwolke als eine Menge von horizontalen Scanebenen aufgefasst. Jede Ebene wird mittels dem später auch in dieser Arbeit verwendeten Jump Distance Verfahren segmentiert, und mehrere vertikal übereinander liegende Segmente werden zu 15 cm hohen dreidimensionalen Segmenten zusammengefasst. Jedes dieser Segmente wird dann anhand von 17 Merkmalen durch einen mit AdaBoost trainierten Klassifikator klassifiziert. Ähnlich zu [Mozos, 2009] wird ein Personenmodell über die relative Lage der einzelnen Segmente verwendet, um bestimmte 3D-Positionen als potentielle Personenmittelpunkte zu unterstützen.

Zur Steigerung der Detektionsleistung der Verfahren, werden sie häufig mit kamerabasierten Detektoren kombiniert. So wird der Lasertracker von [Zhao, 2005] in [Cui, 2005] und [Song, 2010] verwendet, und auch [Arras, 2007] wird in [Spinello, 2008] um einen kamerabasierten Detektor erweitert.

Fazit

Aufgrund der Rahmenbedingungen bzgl. der Experimentierplattform (Abschnitt 2.4) kommen für die laserbasierte Personendetektion nur Verfahren in Betracht, welche auf einzelnen mobilen 2D-Laserscannern basieren. Da in dieser Arbeit auch unbewegte Personen detektiert werden sollen, scheiden die bewegungsbasierten Verfahren aus. Ferner wurden in diesem Kapitel verschiedene Verfahren beschrieben, welche bereits einen Tracker oder auch einen kamerabasierten Detektor enthalten. Im Hinblick auf eine modulare Struktur soll in diesem Abschnitt jedoch nur das Detektionsmodul betrachtet werden. Tracker und weitere Detektionsmodule werden später in diesem Kapitel behandelt. Unter diesen Bedingungen ist die Arbeit von [Arras, 2007], welche auf einem gelernten Beinmodell basiert, besonders viel versprechend. Allerdings wird in dieser angenommen, dass die Beine einer Person einzeln im Scan erscheinen. Bereits in [Scheutz, 2004] und [Topp, 2005] wird jedoch darauf hingewiesen, dass sich die Beine einer Person häufig gegenseitig verdecken oder berühren. Wie schon erwähnt kommt es beim Anwendungsszenario ROREAS häufig zu Verdeckungen durch Gehhilfen. Dieser Herausforderung wird sich in dieser Arbeit angenommen, indem verschiedene Erweiterungen und konzeptionelle Veränderungen an [Arras, 2007] vorgenommen werden. Die Beschreibung dieses Verfahrens, wie es in [Weinrich, 2014a] veröffentlicht wurde, erfolgt in den nächsten Abschnitten mit Bezug auf das Referenzverfahren [Arras, 2007].

4.1.2. Struktur des laserbasierten Personendetektors

Da der im Rahmen dieser Arbeit entwickelte laserbasierte Personendetektor auf dem Verfahren von [Arras, 2007] basiert, entsprechen sich beide Verfahren auch in deren groben Struktur, welche in Abb. 4.2 dargestellt ist. An der Segmentierung wurden keine Veränderungen vorgenommen. Die Merkmale, welche von jedem Segment extrahiert werden, unterscheiden sich jedoch grundlegend. Um neben der binären Klassifikation von *Person* und *keine Person* auch noch die Gehhilfen *Rollator* und *Rollstuhl* unterscheiden zu können, wird nicht nur ein binärer Klassifikator verwendet, sondern es werden drei binäre Klassifikatoren in einem Entscheidungsbaum zur Mehrklassenunterscheidung angeordnet.

Da im Gegensatz zu [Arras, 2007] nicht nur einzelne Beine, sondern ganze Personen detektiert werden sollen, erfolgt nach der Klassifikation noch eine Hypothesengenerierung. Im Folgenden werden diese Verarbeitungsschritte genauer erläutert.

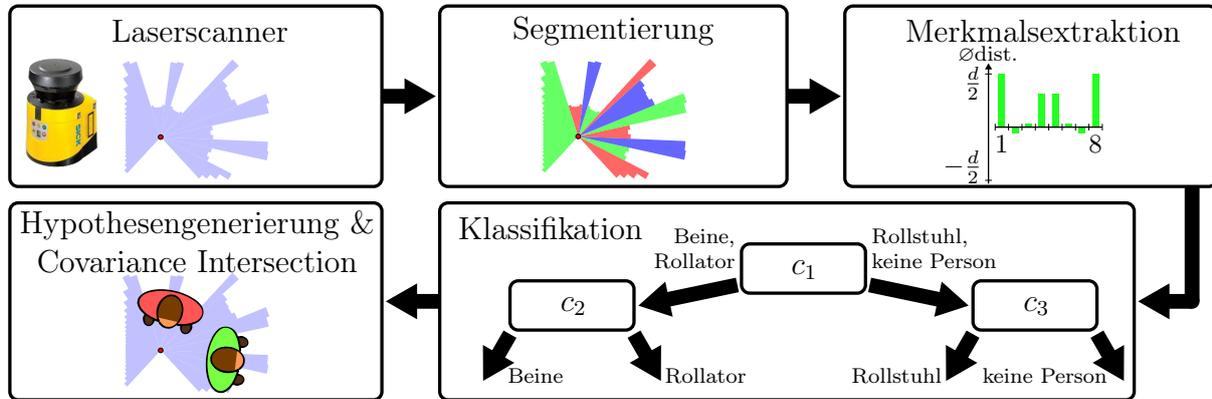


Abbildung 4.2. Verarbeitungsschritte: Verarbeitungsschritte des im Rahmen dieser Arbeit entwickelten laserbasierten Personendetektors. (Bildquelle Laserscanner: www.sick.de)

4.1.3. Segmentierung

Zweck der Segmentierung ist es, die potentiellen Positionen, an welchen sich die zu detektierenden Objekte befinden können, einzuschränken. Es wird angenommen, dass bei zwei benachbarten Laserstrahlen, welche von verschiedenen Objekten reflektiert werden, verschiedene Tiefenwerte gemessen werden. Dieser Sprung in den Tiefendaten wird als Jump Distance bezeichnet. Iterativ werden alle Laserstrahlen bzgl. deren Jump Distance zum vorherigen Strahl untersucht. Wenn diese unterhalb eines Schwellwerts von wenigen Zentimetern ist, wird der aktuelle Strahl dem Segment des vorherigen Strahls zugeordnet. Andernfalls wird dieser Strahl einem neuen Segment zugeordnet. Die nachfolgenden Verarbeitungsschritte werden für jedes Segment des Laserscans durchgeführt, welches aus mindestens drei Laserscans besteht.

4.1.4. Merkmalsextraktion

Anforderungen an die Merkmalsextraktion von Laserdaten

Im Folgenden wird dargelegt, aus welchen Gründen die Merkmale zur Objektdetektion in Laserdaten segmentierungsunabhängig, entfernungsinvariant und generisch sein sollten.

Danach werden die in dieser Arbeit entwickelten Merkmale vorgestellt, welche diese drei Kriterien erfüllen.

Segmentierungsunabhängig Wie im vorigen Abschnitt beschrieben, wird davon ausgegangen, dass ein Segment nur die reflektierten Strahlen eines einzelnen Objektes enthält. Allerdings werden einzelne Objekte häufig in mehrere Segmente unterteilt. So werden die beiden Beine einer Person häufig zwei verschiedenen Segmenten zugeordnet (Abb. 4.3a), und ein Rollstuhl wird in eine sehr große Anzahl an Segmenten unterteilt (Abb. 4.3e). Würden die Merkmale, wie in [Arras, 2007], segmentweise extrahiert, beschrieben die Merkmale nur einen Teil der Erscheinung des übersegmentierten Objektes, und die Detektionsqualität wäre beeinträchtigt. Um dies zu vermeiden, wird in [Spinello, 2008] eine zusätzliche Gruppierung der Segmente basierend auf einer Delaunay-Triangulation durchgeführt. Gerade bei den Gehhilfen führt jedoch auch diese wegen der großen Distanzsprünge zu keinem zufriedenstellenden Ergebnis. Deshalb wird eine Lösung angestrebt, bei welcher der Extraktionsbereich vollkommen unabhängig von der Segmentierung ist und stattdessen von den Ausmaßen der zu detektierenden Objekte abhängt.

Entfernungsinvariant Wie bei vielen anderen Sensoren reduziert sich auch bei Laserdaten die räumliche Auflösung, mit der ein bestimmtes Objekt abgetastet wird, je weiter das Objekt von dem Laserscanner entfernt ist.

Damit eine bestimmte Ansicht eines Objektes unabhängig von der Distanz zum Sensor zu ähnlichen Merkmalen führt, wird bei Bilddaten häufig eine Gauß-Pyramide berechnet. Durch diese kann für Objekte, welche in geringer Distanz zur Kamera wahrgenommen werden, bereits vor der Merkmalsextraktion eine Reduktion der räumlichen Auflösung erreicht werden, so dass alle Objekte so erscheinen, als wären sie in großer Distanz zur Kamera aufgenommen worden. Die hochfrequenten Bildinformationen werden bereits vor der Merkmalsextraktion eliminiert, und die Entfernungsinvarianz ist unabhängig von den extrahierten Merkmalen gegeben. Da jedoch bei Bilddaten die Tiefeninformationen nicht explizit gegeben sind, werden alle potentiellen Entfernungen untersucht, indem die Detektion auf den entsprechenden Pyramidenebenen durchgeführt wird.

Dies ist bei Laserdaten nicht notwendig, da die Objektdistanzen explizit gemessen werden. Die Merkmale können direkt aus den kartesischen Koordinaten der abgetasteten Punkte berechnet werden. So werden in [Arras, 2007] 14 Merkmale wie beispielsweise Kreisförmigkeit, Radius und Standardabweichung extrahiert. Die extrahierten Merkmale sind zwar entfernungsinvariant, stehen jedoch im Konflikt mit der nächsten Anforderung und sollen deshalb ersetzt werden.

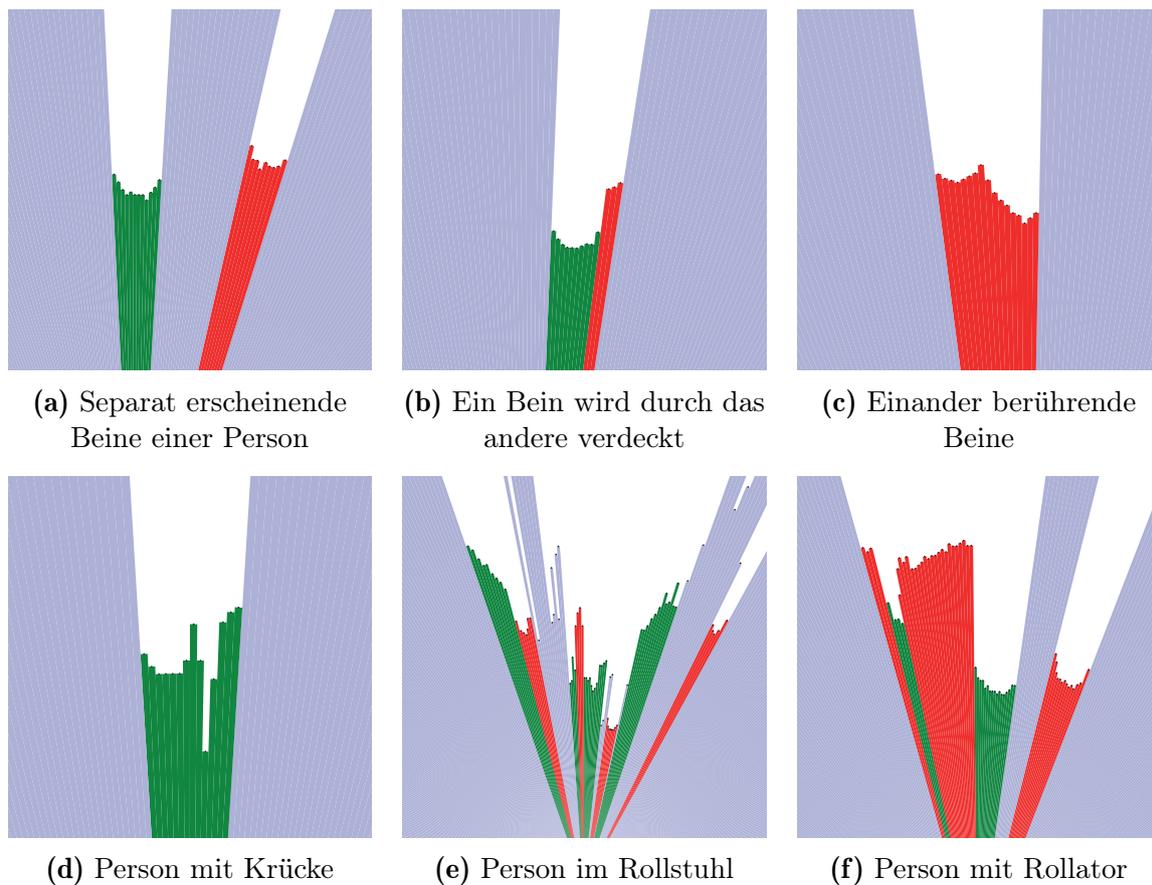


Abbildung 4.3. Erscheinung von Personen in Tiefendaten: Ausschnitte der Tiefendaten, welche Personen (mit Gehhilfen) zeigen. Segmente, welche zum Hintergrund gehören oder aus weniger als drei Laserscans bestehen sind blau dargestellt. Die einzelnen Segmente, welche von Personen (oder Gehhilfen) reflektiert werden, sind abwechselnd grün und rot hervorgehoben, damit benachbarte Segmente unterschieden werden können. (Übernommen aus [Weinrich, 2014a])

Generisch Die in [Arras, 2007] verwendeten Merkmale sind für die Detektion von einzelnen Beinen konzipiert. Bei den im Rahmen dieser Arbeit aufgenommenen Daten von Personen ohne Gehhilfen erscheinen die Beinpaare jedoch nur in 35% der Laserdaten als zwei einzelne Beine. In den meisten Fällen verdeckt ein Bein das andere, oder beide Beine berühren sich. Darüber hinaus sollen auch Personen mit Gehhilfen detektiert werden, deren Erscheinung komplexer ist und große Varianz aufweist. Aus diesem Grund sollten die Merkmale nicht auf die Detektion einzelner, spezifischer Objekte zugeschnitten sein. Denn solche Merkmale verwerfen häufig Informationen, welche zur Detektion anderer Objekte benötigt werden. Stattdessen sollen durch die Merkmale möglichst viele Tiefeninformationen erhalten bleiben, zumindest in dem Umfang, wie dies mit der Forderung nach Entfernungsinvarianz vereinbar ist.

Generic Distance-Invariant Features (GDIF) zur Personendetektion

Die im Rahmen dieser Arbeit entwickelten Generic Distance-Invariant Features (GDIF) [Weinrich, 2014a] sind direkt zur Erfüllung der beschriebenen Anforderungen konzipiert. Im Gegensatz zu den Merkmalen aus [Arras, 2007] ist die Ausdehnung des Bereiches, aus welchem die Merkmale extrahiert werden, unabhängig von der Segmentierung. Stattdessen hat dieser Bereich eine konstante Größe, welche so gewählt wird, dass die zu detektierenden Objekte (Personen, Gehhilfen, ...) immer vollständig erfasst werden. Abbildung 4.4 zeigt, die konstante Breite w und Tiefe d dieses Bereiches. Die Segmentierung wird nur zur Positionierung dieses Extraktionsbereiches genutzt. Eine Übersegmentierung würde somit nur die Anzahl der Positionen, an welchen Merkmale extrahiert werden, steigern. Die innerhalb des Extraktionsbereiches extrahierten Merkmale sind jedoch unabhängig von der Segmentierung.

Wie Abbildung 4.4 zeigt, ist der Extraktionsbereich definiert durch die Basislinie l_j , welche die konstante Breite w des Extraktionsbereiches bestimmt. Die Basislinie verläuft immer orthogonal zu der Geraden zwischen dem Scanner und ihrem Zentrum C_j . Die Basislinien l_j werden für jedes Segment S_j so platziert, dass der erste Abtastpunkt des Segments mit dem Versatz o auf der Basislinie liegt.

Die Basislinie ist in n gleich große Liniensegmente unterteilt (Abbildung 4.5). Diese Liniensegmente dienen dazu, die Laserstrahlen des Extraktionsbereiches in n Gruppen zu unterteilen, indem jeder Laserstrahl dem Liniensegment zugeordnet wird, welches er schneidet oder schneiden würde, wenn er nicht vorher reflektiert würde. Jedes Merkmal wird für jede der n Gruppen, basierend auf dem Abstand der Messpunkte zu den Schnittpunkten der entsprechenden Strahlen mit der Basislinie, berechnet. Der Abstand wird dabei auf den Wertebereich $[-\frac{d}{2}, \frac{d}{2}]$ begrenzt. Damit entspricht d der Tiefe des Extraktionsbereiches und sollte mindestens doppelt so groß sein, wie die maximale Ausdehnung der zu detektierenden Objekte.

Ein besonders wichtiges Merkmal ist der durchschnittliche Abstand der Lasermesspunkte. Wie die Abbildungen 4.5b und 4.5d veranschaulichen, wird durch dieses Merkmal die Distanzinvarianz erreicht. Der Grund dafür ist, dass bei der Lichtlaufzeitmessung mit moduliertem Licht (wie auch bei dem Pulsverfahren) bei einem Laserstrahl, welcher in großer Distanz durch ein Objekt reflektiert wird, die durchschnittliche Entfernung innerhalb des Strahls gemessen wird. Diese Durchschnittsmessung wird bei nahen Objekten durch die Durchschnittsberechnung der Strahlen einer Gruppe nachgebildet.

Weitere Merkmale einer Gruppe von Laserstrahlen, wie der minimale oder maximale Abstand oder die Standardabweichung etc., können genutzt werden, um den Informa-

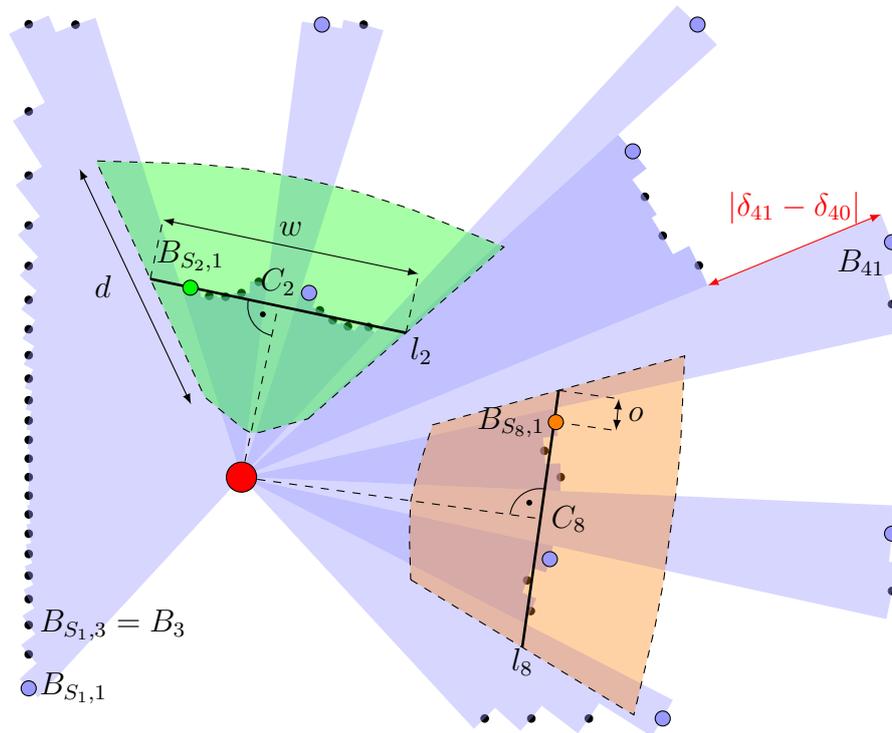
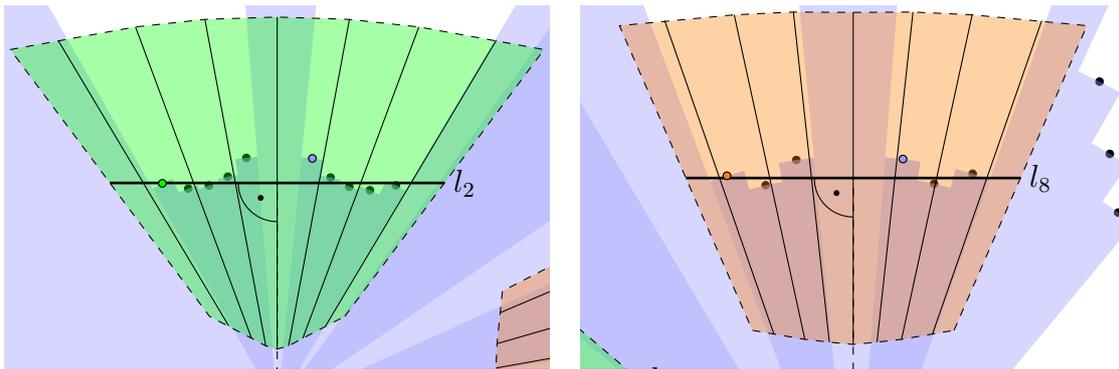


Abbildung 4.4. Merkmalsextraktion aus Tiefendaten : Schematische Darstellung der Tiefendaten (hellblau), eines Laserscanners (roter Kreis) in einem rechteckigen Raum. In den Tiefendaten sind die Beine von zwei Personen zu erkennen. Die Jump Distance, auf welcher die Segmentierung beruht, ist für den Strahl B_{41} exemplarisch gezeigt. Der erste Strahl $B_{S_j,1}$ jedes Segments S_j ist durch einen farbigen Punkt hervorgehoben. Jeder dieser Punkte dient zur Positionierung des Bereiches, aus welchem Merkmale extrahiert werden. In der Darstellung sind jedoch nur die beiden Extraktionsbereiche (grün und orangefarben) dargestellt, deren Merkmale als Person klassifiziert werden sollen. (Übernommen aus [Weinrich, 2014a] und angepasst)

tionsgehalt bei nahen Objekten zu steigern. Bei entfernten Objekten, bei welchen pro Liniensegment nur eine Messung vorliegt oder gar zwischen benachbarten Messungen interpoliert werden muss, steigt der Informationsgehalt durch diese Merkmale nicht.

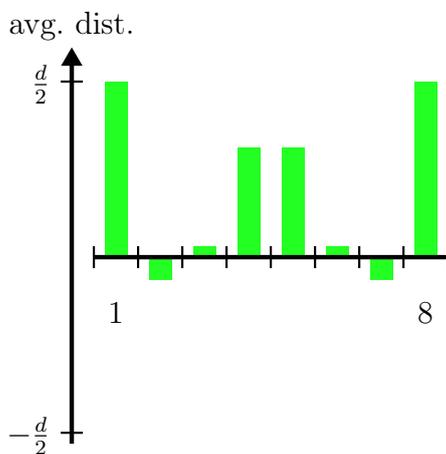
4.1.5. Klassifikation

Sowohl in [Arras, 2007] als auch in [Spinello, 2008] wird ein mittels AdaBoost [Freund, 1995] trainierter Klassifikator eingesetzt. Um den Berechnungsaufwand zu reduzieren, wird in [Spinello, 2008] eine Kaskade von AdaBoost Klassifikatoren verwendet, bei welcher viele Hypothesen bereits in den niedrigen Kaskadenstufen verworfen werden können. Als

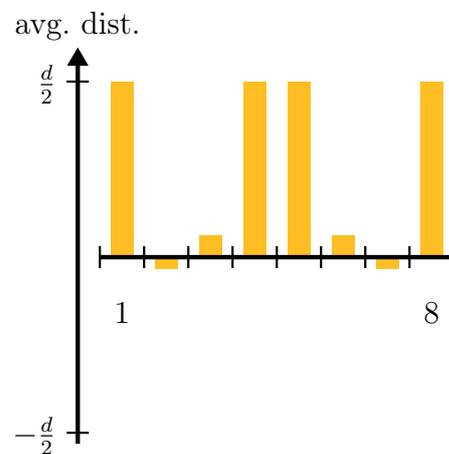


(a) Bereich, aus welchem Merkmale einer nah am Scanner stehenden Person extrahiert werden

(b) Bereich, aus welchem Merkmale einer weiter entfernten Person extrahiert werden



(c) Merkmalsvektor einer nah am Sensor stehenden Person



(d) Merkmalsvektor einer weiter entfernten Person

Abbildung 4.5. Extrahierte Merkmale am Beispiel der durchschnittlichen Distanz: Bereiche, aus welchen die Merkmale einer (a) nah am Scanner stehenden Personen und einer (b) weiter entfernten Person (siehe Abbildung 4.4) extrahiert werden. Beispielhaft ist in (c) und (d) die durchschnittliche Distanz zur Basislinie für $n = 8$ Liniensegmente visualisiert. (Übernommen aus [Weinrich, 2014a])

Weak Classifier dienen in beiden Verfahren einfache Schwellwerte bzgl. der einzelnen eindimensionalen Merkmale. In [Spinello, 2008] werden zum Finden des Schwellwerts SVMs eingesetzt.

In dieser Arbeit werden als Weak Classifier hingegen binäre Entscheidungsbäume [Breiman, 1984] eingesetzt. Diese sind mächtiger als die in [Arras, 2007] und [Spinello, 2008] verwendeten Schwellwertklassifikatoren. Ein binärer Entscheidungsbaum der Tiefe 1, auch Stub genannt, entspricht einer Schwellwertklassifikation. In [Arras, 2007] wurden durch

AdaBoost 10 Stubs zu einem binären Klassifikator vereint. In dieser Arbeit werden weitaus weniger rechenaufwändige Merkmale verwendet und dafür komplexere Klassifikatoren untersucht. In den Experimenten werden zur binären Klassifikation bis zu 50 Entscheidungs-bäume der Tiefe 10 zu einem Strong Classifier kombiniert.

Ziel dieser Arbeit ist jedoch nicht nur die Detektion von Personen, sondern auch die Unterscheidung ihrer Gehhilfen. Deshalb bilden drei der zuvor beschriebenen binären Strong Classifier c_i die Knoten eines binären Entscheidungsbaumes, um vier Klassen (Beine, Rollator, Rollstuhl, keine Person) voneinander zu unterscheiden (Abb. 4.2).

4.1.6. Hypothesengenerierung

Falls die extrahierten Merkmale als Person (mit Gehhilfe) klassifiziert werden, wird eine Personenhypothese am Zentrum der Basislinie C_j generiert.

Da in [Arras, 2007] einzelne Beine detektiert werden, müssten diese zusätzlich noch zu Beinpaaren zusammengefasst werden, um nur eine Personenhypothese pro Person zu erzeugen. Theoretisch ist dieser Bearbeitungsschritt bei den GDIF nicht notwendig, da beide Beine im Extraktionsbereich der Merkmale erfasst und direkt als Person detektiert werden sollten. Praktisch werden jedoch auch bei GDIF einzelne Beine als Person klassifiziert, weil manchmal nur ein Bein einer Person sichtbar ist (Abb. 4.3b) und auch diese Personen detektiert werden sollen. Wenn die Beine einer Person separat erscheinen (Abb. 4.3a) wird der Extraktionsbereich der Merkmale zuerst bei dem ersten Laserscan des linken Beins positioniert. Dabei werden Merkmale beider Beine extrahiert und sollten als Person klassifiziert werden. Danach wird der Extraktionsbereich beim ersten Laserscan des rechten Beines positioniert. Falls beim Trainingsdatensatz des Klassifikators aufgrund von Verdeckung häufig nur einzelne Beine einer Person sichtbar waren (Abb. 4.3b), wird auch dieser Extraktionsbereich als Person klassifiziert werden. Um Mehrfachdetektionen einer Person zu vermeiden, werden Personendetektionen in geringem räumlichen Abstand mittels Covariance Intersection [Julier, 1997] zusammengefasst.

4.1.7. Experimente

Datensätze

Zur Evaluation werden Datensätze aus drei verschiedenen Einsatzumgebungen verwendet (Tab. 4.1). Der Datensatz STATIC¹ ist als Beispieldatensatz zum Verfahren [Spinello,

¹<http://www.informatik.uni-freiburg.de/~spinello/people2D.html>

2008], welches auf [Arras, 2007] basiert, veröffentlicht. Er wurde mit einem stationären Laser aufgenommen, während sich einige Personen in der Nähe des Roboters bewegen. Die Varianz des Hintergrundes ist entsprechend gering (Abb. 4.6a), und der Hintergrund der Trainings- und Testdaten unterscheidet sich nicht. Deshalb wird dieser Datensatz auch nicht zum Training eines Klassifikators empfohlen, weil dieser nicht ausreichend generalisieren würde, um auch in komplexeren Umgebungen gute Ergebnisse zu liefern. Dennoch wird dieser Datensatz in dieser Arbeit verwendet, um zu zeigen, dass das vorgestellte Verfahren auch auf fremden Daten überzeugt. Die Datensätze HOME und REHA wurden am Fachgebiet NIKR im Rahmen der Projekte SERROGA² und ROREAS aufgenommen. Zur Aufnahme der Hintergrunddaten für die Trainings- und Testdatensätze haben sich die Roboter durch verschiedene Bereiche der Einsatzumgebungen bewegt. Der HOME Datensatz stammt aus einer häuslichen Umgebung (Abb. 4.6b) und wurde von einem Laserscanner in 23 cm Höhe aufgenommen. Dieser Datensatz zeichnet sich durch eine besonders hohe Varianz in den Hintergrunddaten aus. Die REHA Daten wurden in dem Rehabilitationszentrum aufgenommen. Die Herausforderung dieses Datensatzes besteht vorrangig in der Detektion und Unterscheidung von Personen mit ihren Gehhilfen. Da der Laserscanner die Personen in einer Höhe von 40 cm erfasst, haben die Beine einer Person auch beim Gehen eine geringe Distanz zueinander und somit steigt auch der Anteil sich berührender oder verdeckender Beine (Tab. 4.1). Die Trainingsdatensätze von HOME und REHA enthalten jeweils 1.250 Laserscans mit Objekten jeder Klasse. Somit besteht der Testdatensatz von HOME aus 2.500 Scans der Klassen *Beine* und *keine Person*. Der Testdatensatz von REHA besteht aus 5.000 Scans, da jeweils 1.250 Scans für die Klassen *Rollator* und *Rollstuhl* hinzu kommen.



Abbildung 4.6. Einsatzumgebungen: Ansichten der Einsatzumgebungen in welchen die drei Datensätze aufgenommen wurden. Bildquellen: <http://www.informatik.uni-freiburg.de/~spinello/people2D.html> (STATIC), [Weinrich, 2014a] (HOME, ROREAS)

²Informationen zum Projekt „Service-Robotik für die Gesundheitsassistenz“ im häuslichen Umfeld in Kapitel 1.2.4

Tabelle 4.1. Datensätze: Eigenschaften der zum Training und Test verwendeten Datensätze.

		STATIC	HOME	REHA
Laser-scanner	Öffnungswinkel	180°	270°	270°
	Winkelauflösung	0,5°	0,5°	0,5°
	Einbauhöhe	?	23 cm	40 cm
Tiefen-scans	Insgesamt	38.994	24.249	30.582
	Trainingsdatensatz	19.497 (50%)	21.749 (~90%)	25.582 (~84%)
	Testdatensatz	19.497 (50%)	2.500 (~10%)	5.000 (~16%)
	Personenanzahl	?	7	11
Ansichten von Personen	Person ohne Gehhilfe	? ¹	18.022	13.503
	Beine eindeutig separierbar	?	10.570 (59%)	4.790 (35%)
	Beine gegenseitig verdeckt	?	3.092 (17%)	3.769 (28%)
	Beine berühren sich	?	4.360 (24%)	4.944 (37%)
	Person im Rollstuhl	0	0	5.093
	Person mit Rollator	0	0	4.219

¹ Anzahl nicht bekannt, da nicht Personen, sondern die einzelnen Laserstrahlen gelabelt sind

Detektoren

Um die Leistungsfähigkeit der vorgestellten GDIF Merkmale zu evaluieren, werden diese mit zwei alternativen Merkmalssets verglichen:

ARRAS: Eine Reimplementierung der 14 Merkmale von [Arras, 2007]

SPINELLO: Die Open Source Implementierung der 16 Merkmale aus [Spinello, 2008].

Gegenüber ARRAS wurden drei der 14 Merkmale durch fünf andere ersetzt.

GDIF: Die Merkmale, welche in Absatz 4.1.4 beschrieben sind.

Da sich die Klassifikatoren in [Arras, 2007] und [Spinello, 2008] von dem in dieser Arbeit verwendeten Klassifikator unterscheiden, werden die nachfolgenden drei Klassifikatoren über allen drei Merkmalssets untersucht:

10-01: Ein mit AdaBoost trainierter Klassifikator bestehend aus zehn „schwachen Klassifikatoren“. Jeder Weak Classifier fällt eine Schwellwertentscheidung bzgl. eines der Merkmale und entspricht somit einem Stump, also einem binären Entscheidungsbaum, welcher aus nur einem einzigen Knoten besteht.

50-01: Klassifikator, welcher 50 Stumps zu einem Strong Classifier kombiniert.

50-10: Klassifikator, welcher 50 binäre Entscheidungsbäume mit einer maximalen Tiefe von zehn Knoten kombiniert.

Durch Kombination der drei Merkmalsextraktoren mit den drei Klassifikationstypen ergeben sich insgesamt neun Verfahren, welche im Folgenden auf den drei verschiedenen Datensätzen evaluiert werden. Die Bezeichnung der Verfahren ergibt sich aus der Verknüpfung des Merkmalsbezeichners mit dem Bezeichner für die Klassifikatoren. So entspricht ARRAS-10-01 der Kombination in [Arras, 2007], SPINELLO-50-01 entspricht [Spinello, 2008] und GDIF-50-10 dem hier vorgestellten Verfahren. Es sei angemerkt, dass [Arras, 2007] und [Spinello, 2008] ausschließlich zur Unterscheidung von *Person* und *keine Person* entwickelt wurden. Um darüber hinaus die Gehhilfen des REHA Datensatzes unterscheiden zu können, werden drei Klassifikatoren des jeweiligen Typs in einem binären Entscheidungsbaum entsprechend Abb. 4.2 verwendet. Da die Datensätze von Laserscannern in verschiedenen Höhen aufgenommen wurden (Tab. 4.1) und somit auch die Erscheinung der Personen zwischen den Datensätzen variiert, wurden für alle drei Datensätze verschiedene Klassifikatoren trainiert. Die verwendeten Parameter sind im Anhang Abschnitt D.1 aufgeführt.

Detektionsqualität

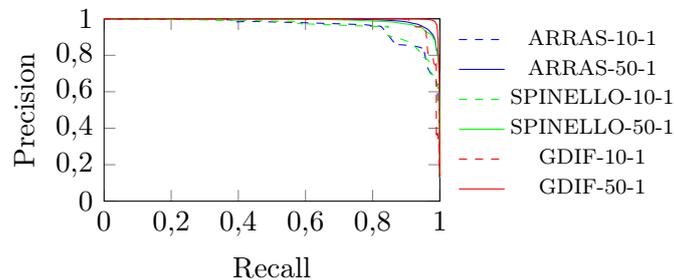


Abbildung 4.7. Evaluation mit STATIC Datensatz: Precision-Recall Kurve für verschiedene Kombinationen von Merkmalsextraktoren und Klassifikatoren zur Klassifikation einzelner Laserstrahlen (Übernommen aus [Weinrich, 2014a] und angepasst)

STATIC Das erste Experiment wurde auf dem Datensatz STATIC durchgeführt. Zur Evaluation wurde dasselbe Bewertungsmaß wie in [Spinello, 2008] verwendet. Bei diesem Datensatz ist jeder einzelne Laserstrahl in Abhängigkeit davon, was diesen reflektierte, als *Person* oder *keine Person* gelabelt. Deshalb werden nach der Detektion einer Person alle Laserstrahlen des entsprechenden Segments als Person klassifiziert. Die Precision-Recall-Kurven, welche sich durch Variation der Strong Classifier Schwellwerte θ ergeben, sind in Abb. 4.7 dargestellt. Es wird deutlich, dass unabhängig vom getesteten Klassifikator die Güte der ARRAS und SPINELLO Merkmale vergleichbar ist und die GDIF Merkmale zu einem deutlich besseren Ergebnis führen. Auch wird gezeigt, dass der Klassifikator

50-01 besser klassifiziert als der Klassifikator 10-01. Die Precision-Recall Kurve für den Klassifikator 50-10 wurde nicht visualisiert, da diese bei allen Merkmalen zu einer fast fehlerfreien Klassifikation führt, und die Kurven nicht mehr unterschieden werden können. Der Grund dafür könnte sein, dass der Datensatz trotz seiner großen Anzahl an Laserscans eine sehr geringe Varianz bzgl. der Erscheinung des Hintergrunds beinhaltet.

HOME Das eigentliche Ziel der evaluierten Verfahren ist nicht die Klassifikation von Laserstrahlen, sondern die Detektion von Personen. Deshalb sind im HOME Datensatz im Gegensatz zu STATIC nicht die einzelnen Laserstrahlen gelabelt, sondern zu jedem Scan sind auch die tatsächlichen Positionen der erfassten Personen im Laser-Koordinatensystem gespeichert. Ist der minimale Abstand zwischen der tatsächlichen Position einer Person und der Personenhypothese geringer als 0,7 m, wird dies als korrekte Detektion (True Positive) gewertet. Sollten einer Person mehr als eine Detektion zugeordnet werden, gelten diese als Fehldetektion (False Positive). Die entsprechenden Precision-Recall-Kurven, welche sich durch Variation des Strong Classifier Schwellwerts θ ergeben, sind in Abb. 4.8a dargestellt. Sie zeigen, dass die Merkmale GDIF bei beiden getesteten Klassifikatoren wieder bessere Ergebnisse erreichen als ARRAS und SPINELLO. Der Klassifikator 10-01 wurde nicht visualisiert, um die Unterscheidbarkeit der Kurven zu gewährleisten. Um einen einzelnen Wert für das Gütemaß eines Detektors D zu liefern, wird zusätzlich für die einzelnen Detektoren D der maximale F_1 Wert über den AdaBoost Schwellwert θ berechnet. Der F_1 Score entspricht dem harmonischen Mittel von Precision und Recall:

$$\max_{\theta} F_1 = \max_{\theta} 2 \cdot \frac{\text{precision}(D_{\theta}) \cdot \text{recall}(D_{\theta})}{\text{precision}(D_{\theta}) + \text{recall}(D_{\theta})} \quad (4.1)$$

Der maximale F_1 Score für ARRAS-50-10 ist 0,90. Für SPINELLO-50-10 ist er 0,77 und für GDIF-50-10 ist er 0,97.

REHA Als nächstes wird die Leistungsfähigkeit der GDIF Merkmale zur Personendetektion mit Unterscheidung von Gehhilfen für den REHA Datensatz untersucht. Dazu werden die Klassifikatoren der drei Knoten des Entscheidungsbaumes zur Mehrklassenunterscheidung (Abb. 4.2) einzeln untersucht. Abbildung 4.8b zeigt die Precision-Recall-Kurven für den Klassifikator c_1 zur Separation der vereinigten Klassen *Beine* \cup *Rollator* und von den Klassen *Rollstuhl* \cup *keine Person*. Der beste F_1 Score für ARRAS-50-10 ist 0,84. Für SPINELLO-50-10 ist er 0,78 und für GDIF-50-10 0,98. Die Precision-Recall-Kurven zur Unterscheidung der in c_1 zusammengefassten Klassen *Beine* und *Rollator* sowie *Rollstuhl* und *keine Person* sind in Abb. 4.9 dargestellt.

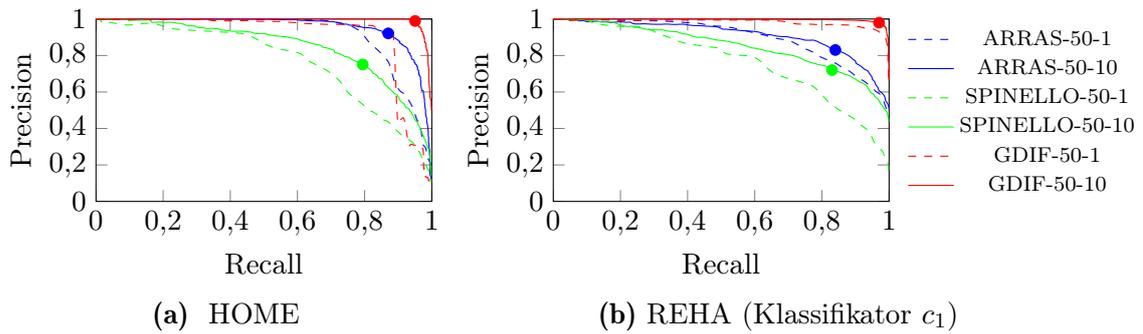


Abbildung 4.8. Evaluation mit HOME und REHA Datensatz: Precision-Recall-Kurven verschiedener Kombinationen von Merkmalsextraktoren und Klassifikatoren zur (a) Detektion von Personen ohne Gehhilfe auf dem *HOME* Datensatz und (b) Separation von $Beine \cup Rollator$ und $Rollstuhl \cup keine Person$ durch Klassifikator c_1 auf dem *REHA* Datensatz.

Abbildung 4.9a zeigt die Unterscheidung von *Beine* und *Rollator* (Klassifikator c_2). Der beste F_1 Score für ARRAS-50-10 ist 0,84. Für SPINELLO-50-10 0,71 und für GDIF-50-10 0,92. Die schlechteren Werte lassen sich dadurch erklären, dass bei Personen, welche einen Rollator schieben und von hinten durch den Laserscanner erfasst werden, beide Beine vollkommen sichtbar sind, jedoch der Rollator großteils verdeckt wird.

Die Precision-Recall-Kurven zur Unterscheidung der Klassen *Rollstuhl* und *keine Person* (Klassifikator c_3) sind in Abb. 4.9b abgebildet. Hier macht der maximale F_1 Score die Leistungsfähigkeit von GDIF-50-10 (0,98) gegenüber ARRAS-50-10 (0,82) und SPINELLO-50-10 (0,75) besonders deutlich.

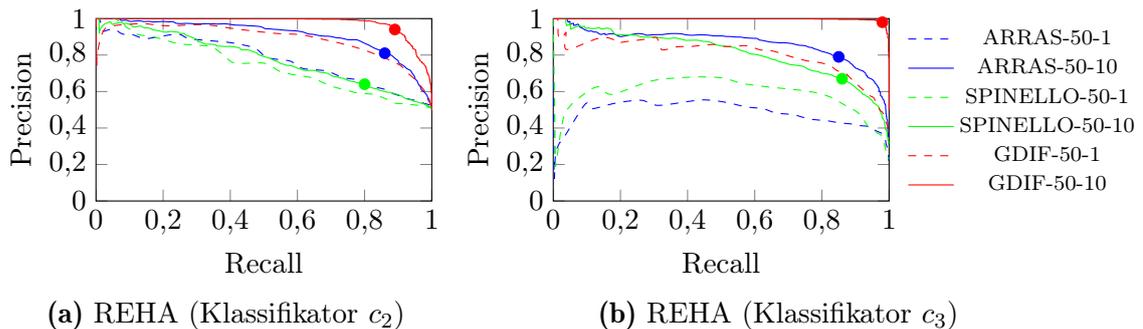


Abbildung 4.9. Weitere Evaluation auf REHA Datensatz: Precision-Recall-Kurven verschiedener Kombinationen von Merkmalsextraktoren und Klassifikatoren zur (a) Unterscheidung von *Beine* und *Rollator* durch Klassifikator c_2 und (b) Separation von *Rollstuhl* und *keine Person* durch Klassifikator c_3 auf dem *REHA* Datensatz. (Übernommen aus [Weinrich, 2014a] und angepasst)

Mittels der Precision-Recall-Kurven kann die Leistungsfähigkeit der verschiedenen Merkmalsextraktoren verglichen werden, und der maximale F_1 Score stellt ein objektives Op-

timierungskriterium für den Arbeitspunkt der einzelnen Klassifikatoren dar. Tabelle 4.2 zeigt die Leistungsfähigkeit der resultierenden Entscheidungsbäume zur Mehrklassenunterscheidung für ARRAS-50-10 und GDIF-50-10 Klassifikatoren. Der Klassifikator 50-10 war bei allen binären Klassifikationen der Leistungsfähigste. Die SPINELLO Merkmale werden in der Konfusionsmatrix nicht dargestellt, da diese ohnehin zu weniger guten Ergebnissen führen als ARRAS.

Tabelle 4.2. Evaluation der Mehrklassenentscheidung: Konfusionsmatrix der Mehrklassenentscheidungsbaume mit **GDIF-50-10** (fett) und *ARRAS-50-10* auf dem REHA Datensatz

		Ground-Truth						Fehldetektionen auf 100 Scans	
		<i>Beine</i>		<i>Rollator</i>		<i>Rollstuhl</i>			
Klassi- fikation	<i>Beine</i>	92%	85%	13%	21%	3%	7%	0.56	7.84
	<i>Rollator</i>	7%	4%	82%	54%	5%	9%	0.36	1.42
	<i>Rollstuhl</i>	0%	2%	5%	12%	91%	70%	0.52	4.12
	<i>keine Person</i>	1%	9%	0%	13%	1%	14%		

Insgesamt werden durch Verwendung der GDIF Merkmale 99.1% der Personen unabhängig von deren Gehhilfe detektiert, und bei 88% der Personen wird auch die Gehhilfe korrekt klassifiziert. Im Gegensatz dazu werden bei Verwendung der ARRAS Merkmale nur 88% der Personen detektiert und bei gerade einmal 70% der Personen wird die Gehhilfe korrekt klassifiziert. Durch die Verwendung der GDIF Merkmale konnte die Anzahl an Fehldetektionen pro 100 Laserscans gegenüber ARRAS von 13.38 auf 1.44 gesenkt werden.

Berechnungsaufwand

Neben der Detektionsqualität ist in der mobilen Robotik vor allem der Berechnungsaufwand der Detektoren relevant. Die reine Extraktion der GDIF Merkmale benötigt $65 \cdot 10^3$ CPU-Zyklen, wohingegen für die Extraktion der ARRAS Merkmale durchschnittlich ca. $180 \cdot 10^3$ Zyklen benötigt werden. Das entspricht einer fast 2.8-fachen Beschleunigung. Die Beschleunigung gegenüber der Open Source Implementierung der noch aufwändigeren SPINELLO Merkmale ist noch größer. Das gesamte Verfahren zur Personendetektion mit dem Mehrklassenentscheidungsbaum (GDIF-50-10) benötigt durchschnittlich bei einem 270° Laserscan $7,075 \cdot 10^3$ CPU-Zyklen. Das entspricht ca. 2,5 ms auf einem 2,8 GHz CPU. Somit benötigt die laserbasierte Personendetektion mit einer 10 Hz Aktualisierungsrate nur 2.5% der Rechenleistung eines einzelnen 2,8 GHz CPU-Kerns.

4.1.8. Zusammenfassung und Ausblick

Die im Rahmen dieser Arbeit entwickelten GDIF Merkmale zeigen gegenüber den Merkmalen aus [Arras, 2007], welche im Jahr 2015 den De-facto-Standard darstellen, eine 2.8 fache Reduktion des Berechnungsaufwandes. Darüber hinaus erlauben sie bei der Verwendung des gleichen Klassifikators eine Steigerung der Personendetektion von 88% auf 99.1%. Zusätzlich wird bei 88% der Detektionen die korrekte Gehhilfe aus drei möglichen Klassen erkannt. Die Anzahl an Fehldetektion wurde durch die GDIF Merkmale auf 1.44 Fehldetektionen pro 100 Laserscans gesenkt. Das entspricht bei einer Detektionsrate von 10 Hz durchschnittlich etwa einer Fehldetektion in sieben Sekunden. Abschnitt 4.3 präsentiert neben der Beschreibung des Trackers auch eine Erklärung dafür, warum nur ein Bruchteil dieser Fehldetektionen zu einer falschen Personenhypothese führen.

Der Fokus dieser Arbeit in Bezug auf die laserbasierte Personendetektion liegt auf der Verbesserung der Merkmale. Die binäre Klassifikation wurde durch einen zweischichtigen Entscheidungsbaum so erweitert, dass eine Unterscheidung der Gehhilfen möglich ist. Zur Verbesserung der Klassifikationsergebnisse könnte dieser Entscheidungsbaum durch einen Entscheidungsbaum ersetzt werden, bei welchem sich die Struktur aus den Trainingsdaten ergibt. Ein solches Verfahren zur kamerabasierten Unterscheidung von Oberkörperorientierungen wird im nächsten Absatz beschrieben. In Bezug auf die Evaluation wären weitere Experimente interessant, in welchen explizit untersucht wird, wie Personengruppen die Detektionsqualität beeinflussen.

4.2. Kamerabasierte Personendetektion mit Orientierungsschätzung

Zusätzlich zu der laserbasierten Personendetektion soll ein kamerabasierter Detektor eingesetzt werden, um die Sicherheit der getrackten Hypothesen zu steigern. Darüber soll basierend auf den Kamerabildern die Oberkörperorientierung der Personen geschätzt werden. Das Wissen über die Orientierung der Oberkörperpose ist eine wichtige Voraussetzung zur Verbesserung der Interaktion zwischen Menschen und Robotern. So kann im Rahmen der Verhaltensweise zur *gezielten Kontaktaufnahme* anhand der Oberkörperorientierung abgeschätzt werden, ob eine Person den Roboter wahrgenommen haben kann oder gar an einer Interaktion mit dem Roboter interessiert ist. Außerdem hängt im Kontext der *respektvollen Navigation* der Personal Space der Personen von deren Oberkörperorientierung ab [Pacchierotti, 2005].

Im Vergleich zur laserbasierten Personendetektion werden kamerabasierte Detektoren häufiger auch über die Robotik hinaus in Bereichen wie Überwachung und Fahrerassistenzsystemen eingesetzt. Kameras sind eine ausgereifte, vielfältig eingesetzte Technologie und im Allgemeinen verhältnismäßig günstig. Die Bilder haben einen hohen Informationsgehalt, und im Gegensatz zu den Laserdaten ist die kamerabasierte Wahrnehmung dem menschlichen Sehen sehr ähnlich. Aus diesen Gründen ist die kamerabasierte Personendetektion bereits intensiv erforscht. Im nächsten Absatz wird der aktuelle Forschungsstand in Bezug auf dessen Relevanz für diese Arbeit aufgearbeitet.

Eine wesentliche Herausforderung der kamerabasierten Personendetektion resultiert aus der vielfältigen Erscheinung von Personen. Diese wird insbesondere durch unterschiedliche Kleidung, Beleuchtung und die beweglichen Gelenke von Personen hervorgerufen. Zusätzlich kann auch der Hintergrund der Bilder stark variieren. Um dieser Herausforderung zu begegnen, ist ein umfangreicher Trainingsdatensatz notwendig, welcher die Varianz der Erscheinung von Personen im Merkmalsraum ausreichend abdeckt. In Absatz 4.2.3 wird ein Verfahren zum automatischen Labeln umfangreicher Trainingsdatensätze beschrieben. Damit die Personendaten im Labor aufgenommen werden können, wird deren Hintergrund durch Hintergründe eines externen Hintergrunddatensatzes ersetzt (Abb. 4.10).

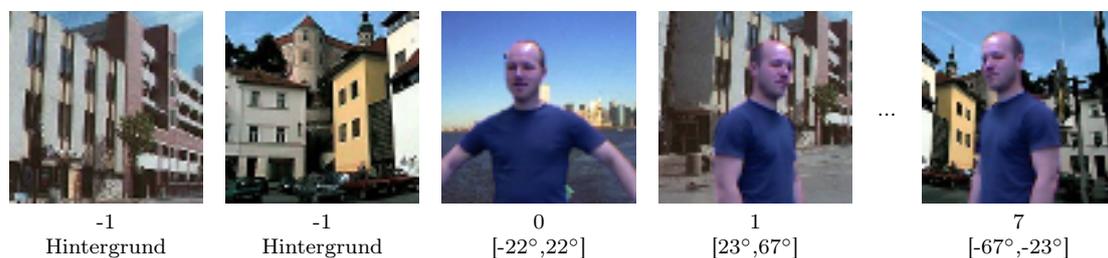


Abbildung 4.10. Trainingsdaten: Beispielbilder des Trainingsdatensatzes mit den zugehörigen Klassenlabels und dem Bereich der Oberkörperorientierungen, welche durch das Label repräsentiert werden. (Darstellung der aus [Weinrich, 2012] übernommenen Beispielbilder in tatsächlich verwendeter Auflösung von 72×72 Pixeln)

Neben der reinen Detektion von Personen ist in dieser Arbeit die Oberkörperorientierung von besonderem Interesse. In Bezug auf die Anwendungsszenarien wird davon ausgegangen, dass die Personen sitzen oder stehen und bzgl. der Orientierung nur die Drehung um die vertikale Achse relevant ist. Diese Orientierung ermöglicht es abzuschätzen, worauf die Aufmerksamkeit einer Person gerichtet sein kann, ob sie den Roboter sehen kann, wie ihr Personal Space orientiert ist, wohin sie sich bewegen möchte, uvm.. Zwar ließe sich die Oberkörperorientierung bei sich bewegenden Personen auch aus deren getrackter Bewegungsspur ableiten, insbesondere im ROREAS Szenario ist jedoch auch die Oberkörperorientierung von stehenden Personen von Bedeutung. In Abschnitt 4.2.4 wird ein diskriminatives Verfahren beschrieben, welches nicht nur Personen in beliebiger Orientierung

zum Roboter detektiert, sondern zusätzlich acht verschiedene Orientierungsklassen effizient unterscheidet. Jede Orientierungsklasse umfasst folglich 45° . Diese Winkelauflösung ist einerseits ausreichend für die Anwendungen, und andererseits werden die Experimente (Abschnitt 4.2.5) zeigen, dass sich acht Orientierungsklassen noch effizient unterscheiden lassen. Zusammen mit der Hintergrundklasse werden somit neun Zielklassen (Abb. 4.10) unterschieden.

4.2.1. Forschungsstand

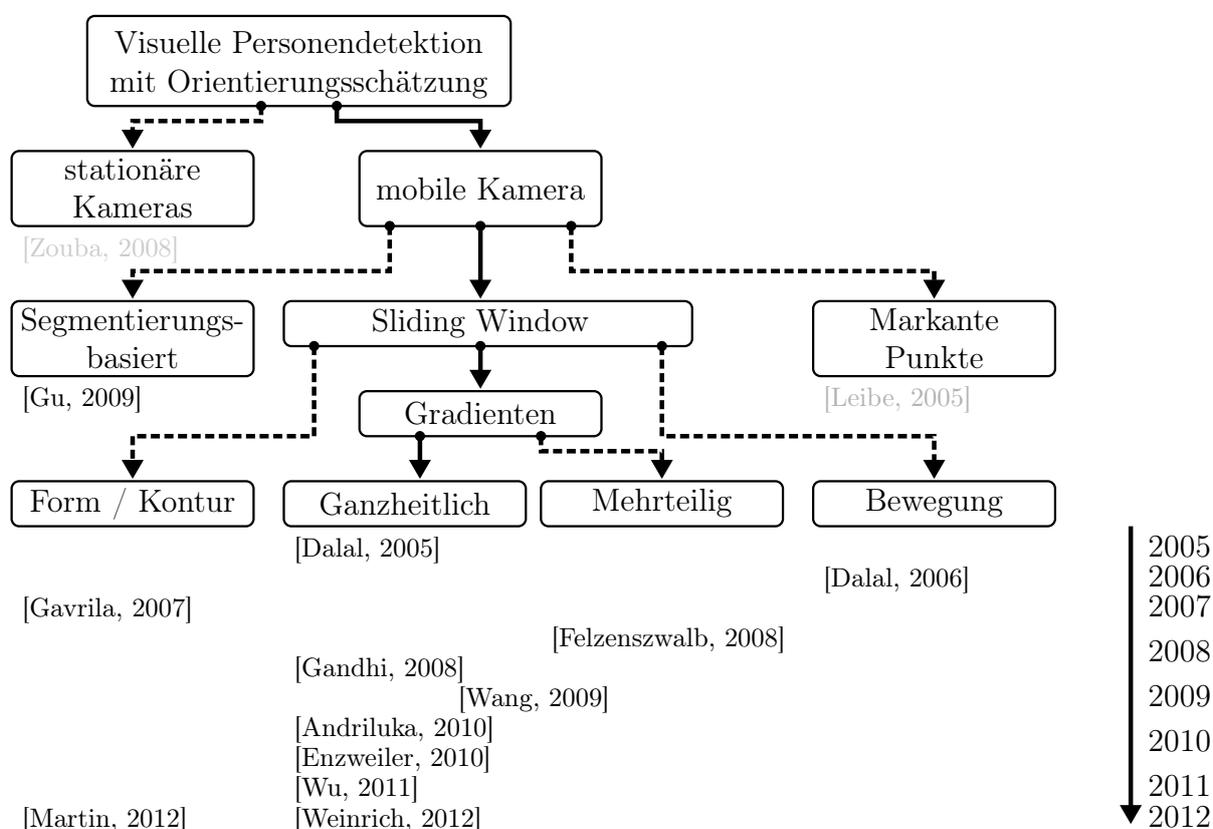


Abbildung 4.11. Arbeiten zur kamerabasierten Personendetektion mit Orientierungsschätzung: Systematische Einordnung von Verfahren zur kamerabasierten Personendetektion mit Orientierungsschätzung nach notwendiger Hardware, Methodik und Jahr der Veröffentlichung

Da der Forschungsstand zur kamerabasierten Personendetektion sehr umfangreich ist, liegt der Fokus in der nachfolgenden Übersicht bei Verfahren, welche Personen in einzelnen Bildern detektieren und zusätzlich die Orientierung des menschlichen Oberkörpers schätzen. Aufwändigere Verfahren zur Schätzung von (3D-)Gelenkstellungen werden hier

nicht berücksichtigt, da diese Verfahren in Absatz 5.1 behandelt werden. Außerdem werden nur Verfahren berücksichtigt, welche auch Personen in geringer Auflösung (Personenhöhe ca. 120 Pixel) detektieren können.

Personendetektion

Erstes Unterscheidungskriterium der Personendetektoren ist die Auswahl der Bildregionen, in welchen sich möglicherweise eine Person befindet (Abb. 4.11). Bei stationären Kameras wird häufig ein Hintergrundmodell gelernt [Zouba, 2008]. So braucht für die Vordergrundregionen nur noch ermittelt werden, ob es sich um Personen oder andere dynamische Objekte handelt. Diese Klassifikation basiert meist auf einer pixelgenauen Silhouette der Vordergrundobjekte.

Bei mobilen Kameras ist diese implizite Detektion von dynamischen Objekten jedoch nicht möglich. Stattdessen werden segmentierungsbasierte Verfahren [Gu, 2009] oder markante Punkte [Leibe, 2005] eingesetzt. Da diese beiden Verfahren jedoch weniger gut bei geringer Auflösung funktionieren [Dollar, 2012], werden hauptsächlich Sliding-Window-Verfahren eingesetzt, um das Detektionsproblem in ein Klassifikationsproblem zu überführen. Farbe, wie beispielsweise Hautfarbe, wird in neueren Arbeiten kaum noch zur Personendetektion eingesetzt. Stattdessen beschreiben die aus den Detektionsfenstern extrahierten Merkmale meist die Form bzw. Konturen [Gavrila, 2007] oder Bewegung [Dalal, 2006] innerhalb des Fensters. Als besonders leistungsfähig erwiesen sich Verfahren, welche auf Gradienten, insbesondere den Histogrammen über Gradientenorientierungen (Histograms of Oriented Gradients - HOG) [Dalal, 2005], basieren. Da bei vielen zu detektierenden Objekten die relative Lage einzelner Teile variiert, werden neben Verfahren, bei denen die Merkmale das Objekt als Ganzes beschreiben, auch Verfahren unterschieden, bei welchen einzelne Teile der Objekte detektiert werden [Felzenszwalb, 2008]. In [Wang, 2009] werden sowohl ganzheitliche als auch Teildetektoren basierend auf HOG- und LBP- (Local Binary Pattern) Merkmalen eingesetzt, um auch Personen, welche teilweise verdeckt werden, besser detektieren zu können. Neben HOG und LBP werden auch weitere gradientenbasierte Merkmale, wie CENTRIST [Wu, 2011], verwendet oder verschiedene Merkmale miteinander kombiniert [Wojek, 2008].

Orientierungsschätzung

Um die Oberkörperorientierung zu ermitteln, detektieren verschiedene Verfahren erst die Personen und schätzen danach die Orientierung in einem weiteren Verarbeitungsschritt

[Gandhi, 2008]. Auch in [Martin, 2012] werden erst HOG-Merkmale zur groben Personendetektion verwendet und danach wird mittels Active Shape Models [Cootes, 1995] die Kontur der Person modelliert. Mittels einer Hauptkomponentenanalyse (PCA) [Pearson, 1901] bzw. Unabhängigkeitsanalyse (ICA) [Comon, 1994] wurde eine geringe Anzahl an Parametern ermittelt, mit welchen das Modell an eine beobachtete Erscheinung angepasst werden kann. Um ausgehend von den optimierten Modellparametern auf die Orientierung einer Person zu schließen, wird eine Support Vector Machine [Cortes, 1995] verwendet.

Im Gegensatz dazu wird in den Sliding-Window-Verfahren [Andriluka, 2010] und [Enzweiler, 2010] die Personendetektion und die Orientierungsschätzung in einem Verarbeitungsschritt durchgeführt. Dadurch wird nicht nur erreicht, dass der zusätzliche Berechnungsaufwand zur Schätzung der Oberkörperorientierung gering gehalten wird, sondern auch die orientierungsunabhängige Detektion wird verbessert. Bei beiden Verfahren wird für jedes Detektionsfenster ein orientierungsabhängiger Detektor je Orientierungsklasse angewendet. In einem weiteren Schritt werden die realwertigen Ausgabewerte der einzelnen Detektoren zu der prädizierten Orientierungsklasse bzw. einem kontinuierlichen Winkel verrechnet. Als Detektoren werden in [Andriluka, 2010] Pictorial Structures [Felzenszwalb, 2005] verwendet, wobei zusätzlich zu den acht Detektoren für die acht Orientierungsklassen noch drei weitere eingesetzt werden. In [Enzweiler, 2010] werden verschiedene Detektoren untersucht, wobei mit HOG-Detektoren [Dalal, 2005] die besten Ergebnisse erreicht wurden. Es sei angemerkt, dass die orientierungsabhängigen HOG-Detektoren die gleichen HOG-Merkmale verwenden. Die berechnungsintensive Merkmalsextraktion braucht somit nur einmal durchgeführt zu werden. Dennoch müssen für jedes Detektionsfenster mindestens so viele binäre Klassifikationen wie Orientierungsklassen berechnet werden. Danach werden die Ergebnisse der einzelnen Detektoren in [Enzweiler, 2010] genutzt, um mittels einer Gaußschen Mischverteilung (GMM) die Wahrscheinlichkeitsverteilung über die Orientierungsklassen zu approximieren. Der Modus dieser Mischverteilung wird nach [Carreira-Perpinan, 2000] berechnet, um einen kontinuierlichen Orientierungswinkel zu erhalten. In [Andriluka, 2010] werden verschiedene Verfahren untersucht, um aus den Ausgabewerten der acht Detektoren die resultierende Orientierungsklasse zu ermitteln. Das Verfahren mit den besten Klassifikationsergebnissen ist vergleichbar mit einer lineare 1-versus-Rest (1-v-R) SVM.

Unterscheidung der Orientierungsklassen basierend auf Detektorausgaben gegenüber einer Unterscheidung direkt im Merkmalsraum

Die Ermittlung der Orientierung findet in [Enzweiler, 2010] und [Andriluka, 2010] nicht direkt im Merkmalsraum statt, sondern auf Basis der kontinuierlichen Ausgaben der orien-

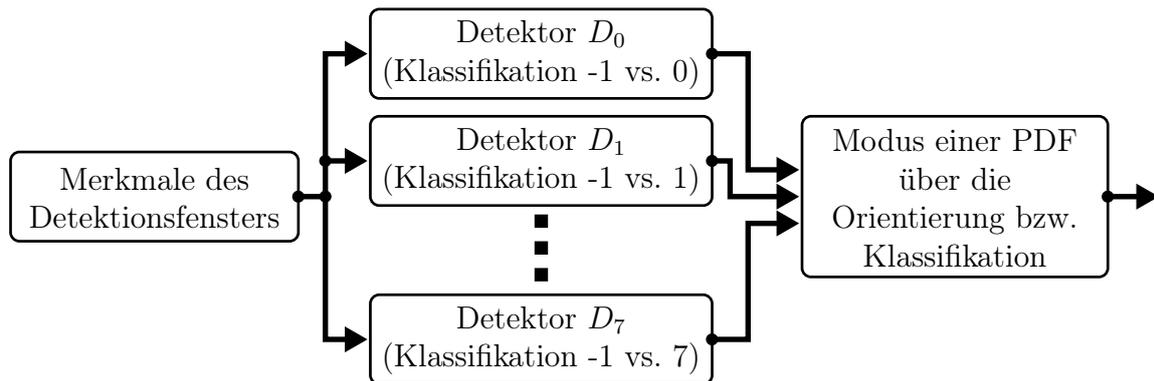


Abbildung 4.12. Grundprinzip der Referenzverfahren [Enzweiler, 2010] und [Andriluka, 2010] zur Orientierungsschätzung: Für jedes Detektionsfenster wird pro Orientierungs-klasse (0 – 7) ein Detektor (und in [Andriluka, 2010] noch drei zusätzliche Detektoren) angewendet. Anschließend werden die kontinuierlichen Ausgabewerte der Detektoren zur Ermittlung des Modus der Wahrscheinlichkeitsdichtefunktion (PDF) über die Orientierung [Enzweiler, 2010] bzw. zur Klassifikation bzgl. der acht Orientierungsklassen [Andriluka, 2010] verwendet.

tierungsabhängigen Detektoren (Abb. 4.12). In [Enzweiler, 2010] wird sogar explizit davon abgeraten, eine 1-v-1 bzw. 1-v-R SVM zu verwenden, bei welcher die Orientierungsklassen und die Hintergrundklasse gleichermaßen im Merkmalsraum separiert werden. Dies wird damit begründet, dass dafür ähnliche Trainingsdaten von denselben Personen in unterschiedlichen Orientierungen benötigt würden³.

Bei einer Unterscheidung der Orientierungsklassen direkt im Merkmalsraum stünden jedoch mehr Informationen zur Unterscheidung der einzelnen Orientierungsklassen zur Verfügung. Um diesen Vorteil nutzen zu können, ist es neben den beschriebenen Anforderungen an die Trainingsdaten erforderlich, dass der Klassifikator den unterschiedlichen Datenverteilungen der Hintergrundklasse und der Orientierungsklassen entsprechen kann. Tabelle 4.3 zeigt die Eigenschaften verschiedener im Rahmen dieser Arbeit untersuchter Methoden zur Mehrklassenunterscheidung mittels SVMs. Eine Beschreibung der verglichenen Methoden erfolgt im Anhang D.2. Die Angaben bzgl. des Rechenaufwandes und der Separationsfähigkeit stellen einen relativen Vergleich der Verfahren auf den in dieser Arbeit verwendeten Daten dar. Insbesondere bei der nichtlinearen SVM kann der Rechenaufwand und die Separationsfähigkeit bei bestimmten Datensätzen abweichen. Viel versprechend ist das Verfahren von [Osman, 2007], bei welchem lineare SVMs als binäre Entscheider in einem Entscheidungsbaum eingesetzt werden. Ein wesentlicher Verarbeitungsschritt des Trainings eines Entscheidungsbaumes ist die Gruppierung der zu trennen-

³Dieser Forderung wird durch der in dieser Arbeit entwickelten Datenaufnahme (Abstanz 4.2.3) entsprochen.

den Zielklassen zu zwei Klassen, welche in einem Knoten des Baumes separiert werden. In [Osman, 2007] wird in diesem Zusammenhang davon ausgegangen, dass sich zwei Klassen gut linear voneinander separieren lassen, wenn die Schwerpunkte ihrer Datenverteilungen einen großen Abstand haben. Dies trifft jedoch nur zu, wenn es sich um konvexe Datenverteilungen handelt. Bei dem in dieser Arbeit verwendeten Datensatz ist dies nicht gegeben, und so muss eine andere Methode zur Gruppierung angewendet werden.

Tabelle 4.3. Mehrklassenunterscheidung mit SVMs: Verschiedene Verfahren zur Mehrklassenunterscheidung mittels SVMs und deren notwendige Anzahl an binären Klassifikationen relativ zur Anzahl der unterschiedenen Klassen C , sowie qualitative Aussagen bzgl. Rechenaufwand und Separationsfähigkeit der Daten zur Oberkörperunterscheidung

Binärer Klassifikator	Mehrklassenunterscheidung	#Bin. Klassifikationen	Rechenaufwand	Separationsfähigkeit
lin. SVM	1-v-R [Vapnik, 1998]	C	---	+
lin. SVM	1-v-1 [Kreßel, 1999]	$\frac{C^2-C}{2}$	----	++
nichtlin. SVM	1-v-R [Vapnik, 1998]	C	-----	+++
lin. SVM	DAGSVM [Platt, 2000]	$C - 1$	--	++
lin. SVM	SVMBDT [Osman, 2007]	$\geq \log_2(C)$	-	+++

Fazit

HOG-Merkmale werden erfolgreich zur Personendetektion verwendet und können wie in [Enzweiler, 2010] zur Unterscheidung von Oberkörperorientierungen eingesetzt werden. In [Andriluka, 2010] und [Enzweiler, 2010] werden für jedes Detektionsfenster mindestens so viele Detektoren bzw. binäre Klassifikatoren berechnet, wie Orientierungsklassen unterschieden werden. Die Bestimmung der resultierenden Orientierungsklasse findet jedoch nicht im Merkmalsraum statt, sondern auf Basis der kontinuierlichen Ausgaben dieser Detektoren bzw. Klassifikatoren. Im Rahmen dieser Arbeit soll untersucht werden, ob durch die Separation der Orientierungsklassen im Merkmalsraum eine bessere Klassifikation möglich ist. Daraus ergeben sich sowohl Anforderungen an die Datenaufnahme als auch an den Klassifikator, welcher die verschiedenen Datenverteilungen von Hintergrund- und Orientierungsklassen separieren muss. Deshalb soll aufbauend auf [Osman, 2007] ein Verfahren zum Training von Entscheidungsbäumen mit linearen SVMs als binäre Entscheider entwickelt werden.

4.2.2. Struktur des kamerabasierten Personendetektors

Die Verarbeitungsschritte des in dieser Arbeit verwendeten Personendetektors unterscheiden sich von [Dalal, 2005] vorrangig in der Klassifikation, da hier neun statt zwei verschiedene Klassen effizient unterschieden werden sollen. Für jedes einzelne Kamerabild wird eine Auflösungspyramide berechnet, und von den Auflösungsstufen werden mittels Sliding-Window-Verfahren potentielle Personenausschnitte generiert. Von jedem dieser Ausschnitte werden die HOG-Merkmale extrahiert, und diese werden durch einen Entscheidungsbaum mit SVMs als binäre Entscheider in den Knoten bzgl. der neun Zielklassen klassifiziert. Als einheitliche Schnittstelle zum Tracker (Abschnitt 4.3) soll auch dieser Detektor 3D-Hypothesen liefern. Deshalb werden die als Person klassifizierten Detektionsfenster unter Nutzung der extrinsischen und intrinsischen Kameraparameter in 3D-Hypothesen umgerechnet.

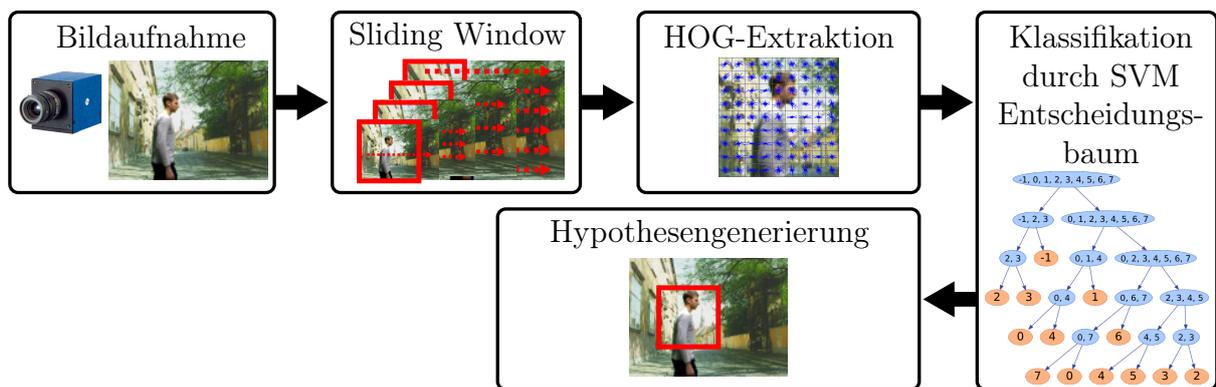


Abbildung 4.13. Verarbeitungsschritte: Verarbeitungsschritte des kamerabasierten Personendetektors. (Bildquelle Kamera: www.svs-vistek.com)

4.2.3. Generierung von Trainingsdaten

Bei der Personendetektion mit Orientierungsschätzung haben die Trainingsdaten wesentlichen Einfluss auf die Leistungsfähigkeit des resultierenden Detektors. Nachfolgend werden verschiedene Eigenschaften der Trainingsdaten, wie sie in dieser Arbeit verwendet werden, erläutert. Anschließend wird das Verfahren zur automatischen Aufnahme dieser Daten beschrieben.

Label Jedes Bild des Trainingsdatensatzes ist mit einem Label versehen. Dieses ist -1, wenn das Bild keine Person zeigt und ansonsten 0 - 7, je nach Oberkörperorientie-

rung in 45° Intervallen (Abb. 4.10). Zusätzlich ist zu jedem Bild auch der kontinuierliche Orientierungswinkel gespeichert. Dieser wird jedoch in dieser Arbeit nur zur Evaluation genutzt.

Bildausschnitt Die Trainingsbilder sind quadratisch. Die Bilder, welche eine Person zeigen, bilden nur den Oberkörper der Person ab, damit in der Anwendungsphase auch Personen, welche sich dicht am Roboter befinden und von denen nur der Oberkörper sichtbar ist, detektiert werden können. Außerdem werden die Bildausschnitte so gewählt, dass sich die Wirbelsäule etwa in der horizontalen Bildmitte befindet und auch die vertikalen Kopf- und Hüftpositionen nur leicht⁴ variieren. Im Gegensatz zu vielen anderen Datensätzen, bei welchen die Bildausschnitte eine Bounding Box um den gesamten Oberkörper bilden würden, sind ausgestreckte Arme hier nicht vollständig im Bildausschnitt enthalten. Dafür hat der Torso immer eine ähnliche Position im Bild, wodurch die Klassifikation erleichtert werden soll.

Hintergrund Um die Oberkörperorientierung erfassen zu können, werden die Daten in einer Laborumgebung aufgenommen. Damit die Personenbilder nicht alle den gleichen Hintergrund zeigen und möglicherweise anhand des Hintergrundes erkannt würden, wird die Greenscreen-Technik angewendet. Auf diese Weise wird der Hintergrund der im Labor aufgenommenen Bilder durch verschiedene Hintergründe aus einem abwechslungsreichen Hintergrunddatensatz ersetzt.

Diese Kombination von Hintergrunddaten und Personendaten hat den Vorteil, dass der Informationsgehalt der Trainingsdaten erheblich erhöht wird, weil in dem Datensatz sehr viele Kombinationen aus Person und Hintergrund enthalten sind. Gerade der Übergang zwischen Person und Hintergrund ist bei den verwendeten HOG-Merkmalen für die Detektion besonders wichtig. [Dalal, 2005]

Datenaufnahme

Wie zuvor beschrieben muss zur Auswahl des Bildausschnittes der Trainingsdaten die Position des Kopfes, der Hüfte und der Wirbelsäule aller aufgenommenen Personen erfasst werden. Darüber hinaus werden zum Labeln der Trainingsdaten bzgl. der Oberkörperorientierung die Schulterpositionen benötigt. Um diese Daten automatisch zu erhalten, wird das Skeletal Tracking der OpenNITM [OpenNI, 2011] zusammen mit der KinectTM

⁴Die Position des Bildausschnitts relativ zu Kopf und Hüfte variiert innerhalb der Hälfte des Wertes, um welchen das Sliding Window in der Anwendungsphase verschoben wird.

Tiefenkamera verwendet. Durch die Verwendung der KinectTM ist der Erfassungsbereich der Personen während der Aufnahme der Trainingsdaten auf 5 m beschränkt. Der Detektionsbereich in der Anwendungsphase wird dadurch jedoch nicht begrenzt. Vielmehr hängt der Detektionsbereich in der Anwendungsphase von der Auflösung der verwendeten Kamera ab. Personen können bis zu der Entfernung detektiert werden, bis zu welcher deren Oberkörper mit derselben Auflösung (72 x 72 Pixel) erfasst wird, wie die verwendeten Bildausschnitte der Trainingsdaten.

Um den Hintergrund der im Labor aufgenommenen Bilder durch Hintergrundbilder des INRIA Datensatzes [Dalal, 2005] ersetzen zu können, wurde zuerst eine Hintergrundsegmentierung im HSI-Farbraum und anschließend der GrabCut-Algorithmus [Rother, 2004] zur pixelgenauen Segmentierung durchgeführt (Abb. 4.14). Im Gegensatz zu den Trainingsdaten wurde der Hintergrund der Testdaten unverändert gelassen. Andernfalls würde bei der Evaluation nicht bemerkt, ob das Verfahren nur funktionieren würde, wenn der Hintergrund ersetzt wird. Die Hintergrundbilder des Testdatensatzes wurden aus der Caltech Hintergrunddatenbank [Fergus, 2003] entnommen.

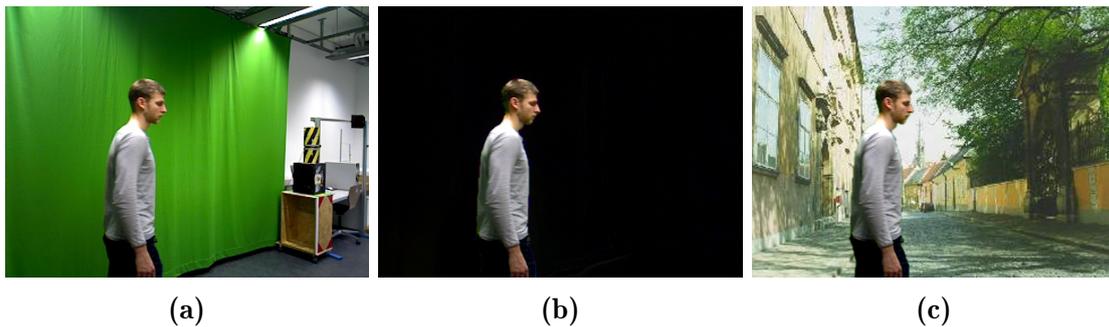


Abbildung 4.14. Ersetzung des Hintergrundes: In den im Labor aufgenommenen Bildern (a) wird mittels Hintergrundsubtraktion die Person segmentiert (b), um dann den Hintergrund durch Bilder des INRIA Datensatzes [Dalal, 2005] zu ersetzen (c).

4.2.4. Mehrklassenunterscheidung mittels SVM Tree

Zur effizienten Klassifikation der HOG-Merkmale bzgl. der neun Zielklassen wird ein Entscheidungsbaum mit linearen SVMs als binäre Entscheider eingesetzt. Die Topologie des SVM Tree zur Klassifikation des Oberkörperdatensatzes HOG-UB (Abschnitt D.4) ist zur Veranschaulichung bereits in Abbildung 4.15 gezeigt.

Der SVM Tree kombiniert die effiziente Klassifikation hochdimensionaler Merkmalsräume mittels linearer SVMs mit den Vorteilen der Entscheidungsbäume bzgl. effizienter Mehrklassenentscheidungen (Abschnitt 4.2.1) und der Korrektur von Separationsfehlern

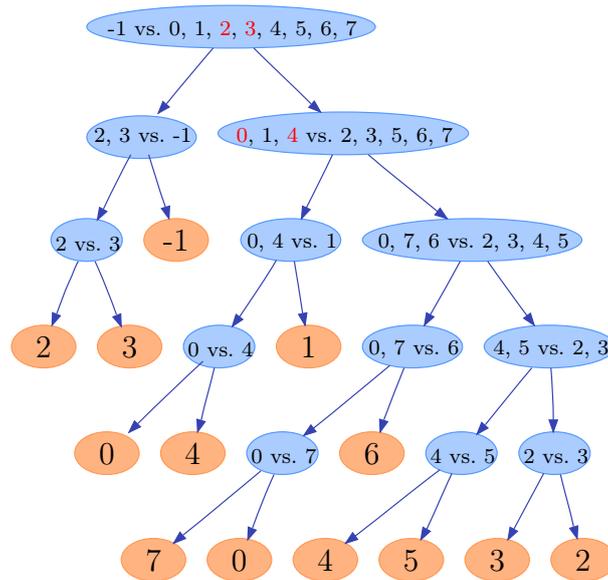


Abbildung 4.15. Topologie des Entscheidungsbaumes zur Personendetektion und Orientierungsschätzung: Struktur des Baumes, welcher auf dem, im Rahmen dieser Arbeit erzeugten, HOG-UB Datensatz trainiert wurde. Die Beschriftung der blauen Knoten gibt an, welche Klassen durch die SVM getrennt werden sollten. Die rot hervorgehobenen Klassen lassen sich nicht ausreichend gut durch die entsprechende SVM trennen. Dieser lokale Klassifikationsfehler führt dazu, dass diese Klassen durch beide Kindknoten behandelt werden. Die Zahlen in den orangefarbenen Blättern sind deren wahrscheinlichste Klasse.

in tieferen Baumebenen. Der SVM Tree (Abbildung 4.15) zeigt, dass zur Klassifikation eines Samples bzgl. der neun Zielklassen maximal fünf lineare SVMs angewendet werden, wohingegen bei einer 1-v-R SVM immer neun SVMs einzusetzen wären. Außerdem zeigt der Baum, dass die lokalen Separationsfehler bzgl. der rot hervorgehobenen Klassen in tieferen Baumebenen durch weitere SVMs ausgeglichen werden.

Um einen gegebenen Merkmalsvektor zu klassifizieren, nutzt der Wurzelknoten des Entscheidungsbaumes eine lineare SVM, um zu entscheiden, in welchem Halbraum das gegebene Sample liegt und welcher seiner beiden Kindknoten entsprechend für die weitere Verarbeitung zuständig ist. Dies wird rekursiv für den Kindknoten wiederholt bis ein Blatt erreicht ist (Abb. 4.15). Für jedes Blatt gibt eine Multinomialverteilung über die Klassenlabel für jede Klasse c die Wahrscheinlichkeit P_c an, dass das Sample zu der jeweiligen Klasse gehört. Abbildung 4.16 zeigt beispielhaft einen Entscheidungsbaum mit der entsprechenden Separation des Merkmalsraumes. Dargestellt ist die Separation von 3 Klassen in einem 2D-Merkmalsraum, denn dies ist anschaulicher als die tatsächliche Separation von 9 Klassen innerhalb eines Merkmalsraumes mit 2304 Dimensionen. Im Folgenden werden diese neun Klassen als Zielklassen bezeichnet, um sie von den binären Klassen einer einzelnen SVM unterscheiden zu können.

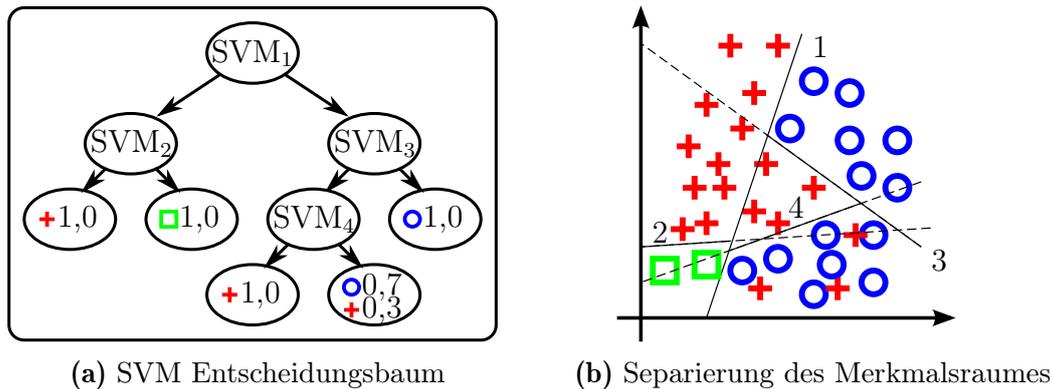


Abbildung 4.16. SVM Entscheidungsbaum zur Mehrklassenunterscheidung: (a) Struktur von SVMs in einer Baumarchitektur zur Klassifikation von drei Klassen (Kreuz, Rechteck, Kreis) und (b) die geometrische Darstellung einer entsprechenden Separation eines 2D Merkmalsraumes.

Ähnlich der Anwendung eines SVM Tree zur Klassifikation, arbeitet auch das im Rahmen dieser Arbeit entwickelte Verfahren zum Training des SVM Tree rekursiv. Initial wird entsprechend Algorithmus 1 der Wurzelknoten trainiert und danach werden die Datensätze zum Training der Kindknoten aufgeteilt, bis eine ausreichende Separation der Daten erreicht wird. Um die SVM eines Knotens zu trainieren, müssen alle Samples der Trainingsmenge S_T dieses Knotens zu einer der binären Klassen -1 und 1 zugeordnet werden. Dies wird im Folgenden als Grouping bezeichnet. Neben dem Grouping gibt es noch weitere Parameter, welche optimiert werden, um eine möglichst geeignete SVM für einen Knoten zu finden. Diese Parameter sind im Anhang in Abschnitt D.3 beschrieben. Wenn eine geeignete SVM gefunden wurde, wird die Trainingsmenge S_T entsprechend der trainierten SVM aufgeteilt und die beiden Teilmengen S_T^{-1} und S_T^1 werden an die beiden Kindknoten übergeben. Diese Aufteilung von S_T in S_T^{-1} und S_T^1 erfolgt mittels der trainierten SVM, wie sie später in der Anwendungsphase auch genutzt wird. Die Separation kann somit bei Klassifikationsfehlern durch die SVM von der gewünschten Aufteilung entsprechend des Groupings abweichen.

Dieser Vorgang wird für die Kindknoten rekursiv wiederholt bis ein bestimmter Anteil $\max_{c=1\dots C} P_c = \max_{c=1\dots C} \frac{|S_{T,c}|}{|S_T|} > p \in (0, 1]$ der Trainingsmenge S_T zu einer einzigen Zielklasse c gehört. Darüber hinaus wird ein Validierungsdatensatz S_V verwendet, um zu entscheiden, wann ein Knoten zu einem Blatt wird. Um Überanpassung zu vermeiden, wird ein Knoten auch zu einem Blatt, wenn eine weitere Separation nicht zu einer besseren Trennung des Validierungsdatensatzes führt.

Das Kriterium zur Bewertung der SVM eines Knotens während der oben angesprochenen Optimierung ist der Information Gain $G(\text{SVM})$ auf dem Validierungsdatensatz S_V . Der

```

Input: Trainingsdatensatz  $S_T$ , Validierungsdatensatz  $S_V$ , Abbruchkriterium  $p$ 
1  $\mathbf{P} \leftarrow$  Häufigkeitsverteilung über Klassenlabel in  $S_T$ ;
2  $\text{SVM} \leftarrow 0$ ;  $N^{-1} \leftarrow 0$ ;  $N^1 \leftarrow 0$ ;
3 if  $\max_{c=1\dots C} P_c \leq p$  then
   // Training der SVM wie im Anhang in Kapitel D.3 beschrieben
4  $\text{SVM} \leftarrow \text{getOptimizedSVM}(S_T, S_V)$  ;
   // Aufteilung von Training- und Validierungsdatensatz
5  $S_T^{-1} \leftarrow \bigcup_{s \in S_T} \text{SVM}(s) = -1$ ;  $S_T^1 \leftarrow \bigcup_{s \in S_T} \text{SVM}(s) = 1$ ;
6  $S_V^{-1} \leftarrow \bigcup_{s \in S_V} \text{SVM}(s) = -1$ ;  $S_V^1 \leftarrow \bigcup_{s \in S_V} \text{SVM}(s) = 1$ ;
   // Rekursives Training der Kindknoten nach Alg. 1
7  $N^{-1} \leftarrow \text{trainTree}(S_T^{-1}, S_V^{-1}, p)$ ;  $N^1 \leftarrow \text{trainTree}(S_T^1, S_V^1, p)$ ;
8 end

```

Algorithmus 1 trainTree: Algorithmus zum Training eines (Teil-)baumes

Gain beschreibt wie stark die Entropie der durch die SVM separierten Validierungsdaten $H(S_V^{-1})$ und $H(S_V^1)$ gegenüber der Entropie der nicht separierten Validierungsdaten $H(S_V)$ sinkt. Der Information Gain wird in dieser Arbeit verwendet, da er sich bereits zum Aufbau von Entscheidungsbäumen durch den ID3 Algorithmus [Quinlan, 1986] bewährt hat:

$$G(\text{SVM}) = H(S) - \frac{|S^{-1}|}{|S|} H(S^{-1}) - \frac{|S^1|}{|S|} H(S^1) \quad (4.2)$$

Die Entropie $H(S)$, $H(S^{-1})$ bzw. $H(S^1)$ wird am Beispiel von $H(S^1)$ wie folgt berechnet:

$$H(S^1) = - \sum_{c=0}^C \frac{|S_c^1|}{|S^1|} \log \frac{|S_c^1|}{|S^1|}, \quad (4.3)$$

wobei S_c^1 die Menge an Samples mit dem Klassenlabel c (in dieser Arbeit $c \in \{-1, 0, \dots, 7\}$) in S^1 ist.

4.2.5. Experimente

Vergleich mit anderen Klassifikatoren Zur Evaluation wird der in dieser Arbeit entwickelte SVM Tree mit verschiedenen alternativen Klassifikatoren, wie sie in Abschnitt D.2 vorgestellt werden, verglichen. Der für die Anwendungsszenarien entscheidende Benchmark wurde auf dem im Rahmen dieser Arbeit erzeugten Datensatz zur

Personendetektion und Oberkörperorientierungsschätzung HOG-UB durchgeführt. Um allgemeinere Aussagen über die Klassifikationsgüte des SVM Tree treffen zu können, wurden weitere Benchmark-Daten auf den in [Chang, 2010] verwendeten, öffentlich verfügbaren Datensätzen erhoben. Für eine detaillierte Beschreibung der Datensätze und des experimentellen Vergleichs sei auf Abschnitt D.4 im Anhang verwiesen. Der Vergleich zeigt die hohe Klassifikationsgüte und den geringen Berechnungsaufwand des hier vorgestellten SVM Tree. In Tabelle 4.4 sind die Experimente zusammengefasst, welche auf dem HOG-UB Datensatz durchgeführt wurden. Es wird deutlich, dass ein reiner Entscheidungsbaum (ID3 Tree) [Quinlan, 1986] die geringste Accuracy erreicht. Die Verwendung von nichtlinearen 1-v-R SVMs in den Blättern eines Entscheidungsbaumes (DTSVM) [Chang, 2010] erhöht zwar die Klassifikationsgüte, jedoch auch den Berechnungsaufwand. Auffällig ist, dass bei dem 2304-dimensionalen Merkmalsraum die Verwendung einer linearen 1-v-R SVM [Vapnik, 1998] eine bessere Klassifikationsgüte als der DTSVM erreicht. Dies könnte dadurch begünstigt sein, dass beim Training des DTSVM nur 10% des Trainingsdatensatzes präsentiert werden konnten. Ein größerer Trainingsdatensatz konnte für das DTSVM-Verfahren nicht verwendet werden, da so bereits der zur Verfügung stehende Arbeitsspeicher mit einer Größe von 128 GB fast vollständig ausgenutzt wurde. Der Berechnungsaufwand der 1-v-R SVM ist erwartungsgemäß gering und es zeigt sich, dass der hochdimensionale Merkmalsraum gut mittels linearer SVMs zerlegt werden kann. Die Verwendung eines SVM Tree steigert die Accuracy noch einmal deutlich von 55% auf 65% und dennoch wird im Vergleich zur 1-v-R SVM weniger als ein Drittel der Rechenzeit benötigt.

Tabelle 4.4. Klassifikationsgüte und Rechenaufwand: Balanced Accuracy (BAC) und CPU-Zyklen verschiedener Verfahren auf dem HOG-UB Datensatz dieser Arbeit

Verfahren \ Kriterium	BAC[%]	CPU-Zyklen
ID3 Tree [Quinlan, 1986]	33.07	$1,0 \cdot 10^{10}$
DTSVM [Chang, 2010]	51.61	$1,3 \cdot 10^{11}$
1-v-R SVM [Vapnik, 1998]	55.23	$8,8 \cdot 10^9$
SVM Tree [Weinrich, 2012]	64.82	$2,6 \cdot 10^9$

Güte der kontinuierlichen Orientierungsschätzung Nach der Anwendung des SVM Tree zur Personendetektion mit Orientierungsschätzung wird ausgehend von der klassifizierten Orientierungsklasse die mittlere Orientierung dieser Klasse für die resultierende Personenhypothese verwendet. Das folgende Experiment bewertet den Fehler zwischen der geschätzten Orientierung und der tatsächlichen Orientierung der Person. Die Aus-

sagekraft dieses Fehlers ist beispielsweise höher, wenn die tatsächliche Orientierung im Grenzbereich zweier Orientierungsklassen liegt. Ein Histogramm über den absoluten Fehler der Orientierungsschätzung ist in Abbildung 4.17 dargestellt. Dieses zeigt, dass der absolute Fehler bei 64% der Orientierungsschätzungen unterhalb von $22,5^\circ$ ist. Es sei angemerkt, dass der halbe Wertebereich einer Klasse $22,5^\circ$ beträgt und somit selbst bei vollkommen korrekter Klassifikation Fehler von $22,5^\circ$ auftreten würden. Das Histogramm zeigt weiter, dass Schätzungen mit einem Fehler von ca. 90° besonders selten vorkommen und die Häufigkeit von 90° bis 180° mit wachsendem Fehler leicht steigt. Dies lässt sich damit erklären, dass sich die Silhouette einer Person gegenüber der Silhouette nach einer Drehung um 180° kaum unterscheidet.

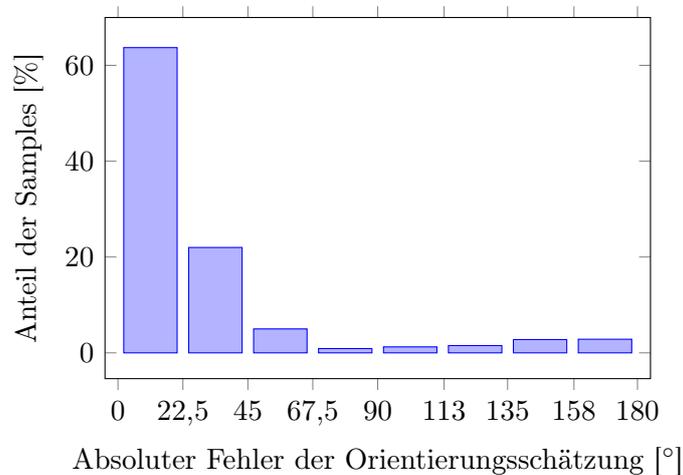


Abbildung 4.17. Histogramm über absoluten Orientierungsfehler des SVM Tree: Histogramm über den absoluten Fehler zwischen geschätzter, diskreter Orientierung und der tatsächlichen Oberkörperorientierung auf dem HOG-UB Datensatz

Die True Positive Rate (TPR) für reine Personendetektion unabhängig von der Oberkörperorientierung beträgt 97,7%, und die False Positive Rate (FPR) ist 0,87%. In Abbildung 4.18 ist das Klassifikationsergebnis eines Beispielbildes dargestellt. Viele Detektionsverfahren wie [Dalal, 2005] wenden beispielsweise den Mean-Shift Algorithmus [Fukunaga, 1975] an, um die Detektionen der unterschiedlichen Ebenen der Auflösungspyramide zu clustern. Dieser Verarbeitungsschritt wäre hier genauso möglich, wird jedoch in dieser Arbeit nicht durch den kamerabasierten Detektor durchgeführt. Stattdessen werden die 3D-Posenhypothesen zu jeder Detektion berechnet und dem zentralen Personentracker (Abschnitt 4.3) übergeben. Die Verrechnung der Mehrfachdetektionen erfolgt im \mathbb{R}^3 durch den Tracker auf Basis der Unsicherheiten der Posenhypothesen.

Die Topologie des Entscheidungsbaumes, welcher auf den HOG-UB Daten trainiert und zur Evaluation verwendet wurde, wurde bereits in Abbildung 4.15 dargestellt. Wie in Abschnitt D.3 beschrieben, werden die Klassen, welche sich im Merkmalsraum sehr ähnlich

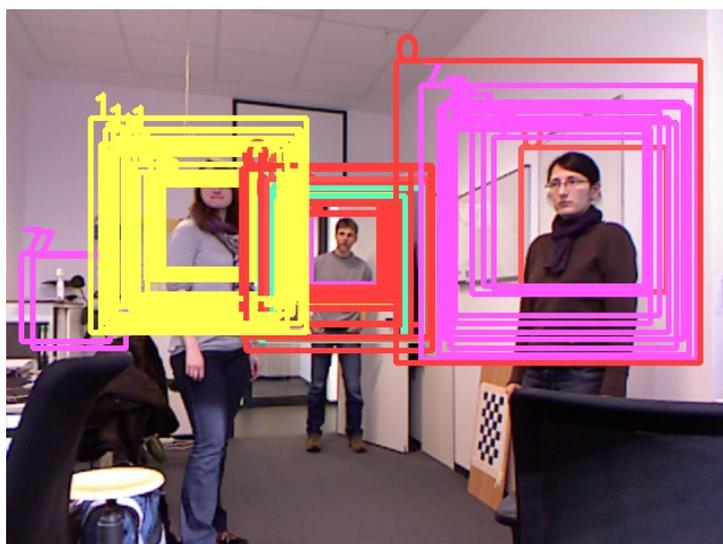


Abbildung 4.18. Beispielhafte Darstellung der Personendetektion mit Orientierungsschätzung: Die Rahmenfarben Gelb (45°), Rot (0°) und Magenta (-45°) zeigen die Orientierungsklassen. Alle Detektionen mit einer Wahrscheinlichkeit über 0,65 sind visualisiert. Da die Hypothesen nicht gruppiert werden, sind alle Hypothesen der unterschiedlichen Ebenen der Auflösungspyramide, welche durch die tatsächlichen Personen hervorgerufen werden, dargestellt.

sind, in den tieferen Ebenen des Entscheidungsbaumes separiert. Erwartungsgemäß zeigt die Baumstruktur, dass sich die Klassen benachbarter Oberkörperorientierung auch im Merkmalsraum ähnlich sind. Ebenso zeigt die Baumstruktur die Ähnlichkeit der Frontalansichten von Personen, welche der Kamera zugewandt (Klasse 0) bzw. abgewandt (Klasse 4) sind.

Die maximale Baumtiefe ist fünf, das bedeutet, es werden maximal fünf lineare 1-v-R SVMs angewendet, um einen Merkmalsvektor zu klassifizieren. Darüber hinaus werden für die korrekte Klassifikation der Hintergrundklasse (-1), welche dem Klassifikator in der Anwendungsphase am häufigsten präsentiert wird, nur zwei 1-v-R SVMs berechnet. Durchschnittlich wurden auf den Testdaten zur Klassifikation eines Merkmalsvektors ca. drei lineare 1-v-R SVMs berechnet. Aus diesem Grund ist der in dieser Arbeit präsentierte Entscheidungsbaum dreimal schneller als eine 1-v-R SVM, welche immer neun binäre SVMs anwendet.

Dadurch erhöht sich die Berechnungszeit zur HOG Merkmalsextraktion, Personendetektion und Orientierungsschätzung gegenüber der Personendetektion ohne Orientierungsschätzung um durchschnittlich 21%. Dies entspricht auf einem 2,8 GHz CPU 824 ms gegenüber 682 ms zur Verarbeitung eines Bildes der Auflösung 640 x 480 Pixel.

Beschränkung des Detektionsraumes Um den Berechnungsaufwand weiter zu reduzieren, wird die Personendetektion nicht für alle Sliding-Window-Positionen der Auflösungspyramide durchgeführt. So werden die extrinsischen und intrinsischen Kameraparameter genutzt, um nur die Sliding-Window-Positionen der Auflösungspyramide zu bearbeiten, an deren entsprechender 3D-Position sich eine Person befinden kann. Nicht behandelt werden dadurch Detektionsfenster, deren entsprechende 3D-Position zu hoch oder zu niedrig über der Bodenebene ist, als dass sich dort eine stehende oder sitzende Person befinden könnte.

Darüber hinaus kann der kamerabasierte Detektor in einem Modus betrieben werden, in welchem diesem 3D-Hypothesen mit räumlicher Unsicherheit übergeben werden, um nur in diesen Bereichen eine Detektion durchzuführen. Dies wird in der Gesamtanwendung genutzt, um den kamerabasierten Detektor mit geringem Rechenaufwand zur Bestätigung der laserbasierten Hypothesen zu verwenden. Genaueres dazu wird in Abschnitt 4.3 beschrieben.

4.2.6. Zusammenfassung und Ausblick

Im Kontext der kamerabasierten Personendetektion wurde das Verfahren von [Dalal, 2005] um eine automatische Erzeugung einheitlicher Trainingsdaten und die Verwendung eines SVM Tree Klassifikators [Weinrich, 2012] erweitert. Dadurch ist eine Schätzung der Oberkörperorientierung von Personen mit einer Winkelauflösung von 45° möglich. Der Fehler der Orientierungsschätzung ist bei 64% der Personen geringer als $22,5^\circ$, und bei 86% der Personen ist er geringer als 45° . Der Berechnungsaufwand für die Detektion mit Orientierungsschätzung hat sich gegenüber der einfachen Detektion um nur 21% erhöht. Dieser Gesamtrechenaufwand konnte jedoch verringert werden, indem die räumlichen Bereiche bei der Personendetektion vernachlässigt werden, in welchen sich keine Personen befinden können. Dadurch ist es möglich, Personen in den vier einzelnen Bildern ($4 \times 752 \times 480$ Pixel) der omnidirektionalen Kamera mit mehr als 2 Hz auf dem lüfterlosen PC mit 1.7 GHz CPU (Abschnitt 2.4) zu detektieren.

Das verwendete Skeletal Tracking der OpenNITM [OpenNI, 2011] liefert bei seitlicher Oberkörperorientierung fehlerhafte Schulterpositionen. Da diese Problematik erst nach der umfangreichen Datenaufnahme festgestellt wurde, wurden die fehlerhaften Daten manuell aus dem Testdatensatz entfernt. Es ist davon auszugehen, dass die Qualität des vorgestellten Verfahrens steigt, wenn auch ein fehlerfreier Trainingsdatensatz verwendet würde. So könnte z. B. das in Kapitel 7 verwendete System aus verteilten Laserscannern (ähnlich [Schenk, 2011]) genutzt werden, um die Orientierung des Oberkörpers zu erfassen.

4.3. Tracking laser- und bildbasierter Hypothesen

In den vorherigen beiden Abschnitten wurden ein kamerabasiertes und ein laserbasiertes Verfahren zur Personendetektion beschrieben. In diesem Abschnitt wird nun geschildert, wie die Personenhypothesen der beiden Detektoren durch einen Tracker fusioniert werden. Da das Tracking von Personen nicht im Fokus dieser Arbeit liegt, sei für eine ausführlichere Beschreibung des verwendeten Trackers auf [Volkhardt, 2013b] verwiesen. Danach wird aufgeführt, wie sich die beiden Detektoren ergänzen, und es werden Details dazu geliefert, wie ein echtzeitfähiges Tracking erreicht wird.

Schnittstelle zum Tracker Der in dieser Arbeit verwendete Tracker [Volkhardt, 2013b] filtert multimodale, asynchrone Objekthypothesen verschiedener Detektoren unter Nutzung eines Kalman-Filters für jede getrackte Hypothese. Eine Hypothese hat sechs Freiheitsgrade, da sie die Position und Orientierung des entsprechenden Objektes im dreidimensionalen Raum beschreibt. Zusätzlich zur 6D Posenhypothese wird auch noch die translatorische Geschwindigkeit der Personen als dreidimensionaler Vektor ausgegeben. Um auch die Unsicherheiten der Hypothesen wiedergeben zu können, wird jede Hypothese durch eine Gaußverteilung repräsentiert. Der Mittelwert der Gaußverteilung repräsentiert somit die Referenzpose eines Objektes, ohne dass die räumliche Ausdehnung oder die relative Lage einzelner Körperteile einer Person wiedergegeben wird. Dementsprechend müssen sich die Hypothesen aller Detektoren auf dieselbe Referenzpose des zu trackenden Objektes beziehen. In dieser Arbeit stellen die Hypothesen die Kopfposition und die Oberkörperorientierung der Personen dar. Damit entsprechen die Hypothesen des kamerabasierten Detektors bereits dieser Referenzpose. Die Hypothesen des laserbasierten Detektors beschreiben jedoch die Beinposition der Personen und müssen entsprechend transformiert werden, bevor sie dem Tracker übergeben werden. Da der laserbasierte Detektor weder die Größe der Personen noch die Oberkörperorientierungen detektiert, ist das Sensormodell der transformierten Hypothesen dieses Detektors bzgl. dieser Freiheitsgrade besonders unsicher (Abb. 4.19).

Intern transformiert der Tracker alle Hypothesen vor der Weiterverarbeitung in ein statisches Koordinatensystem. Dadurch sind die Hypothesen unabhängig von der Bewegung des Roboters und können mit einem linearen Bewegungsmodell getrackt werden. Die Transformation in das statische Koordinatensystem erfordert jedoch, dass die Unsicherheit der Hypothesen, welche aus dem beschriebenen Sensormodell resultieren, um die Lokalisationsunsicherheit der Roboterpose relativ zu dem statischen Koordinatensystem vergrößert werden. Diese Unsicherheitsfortpflanzung ist im Anhang in Absatz D.5 be-

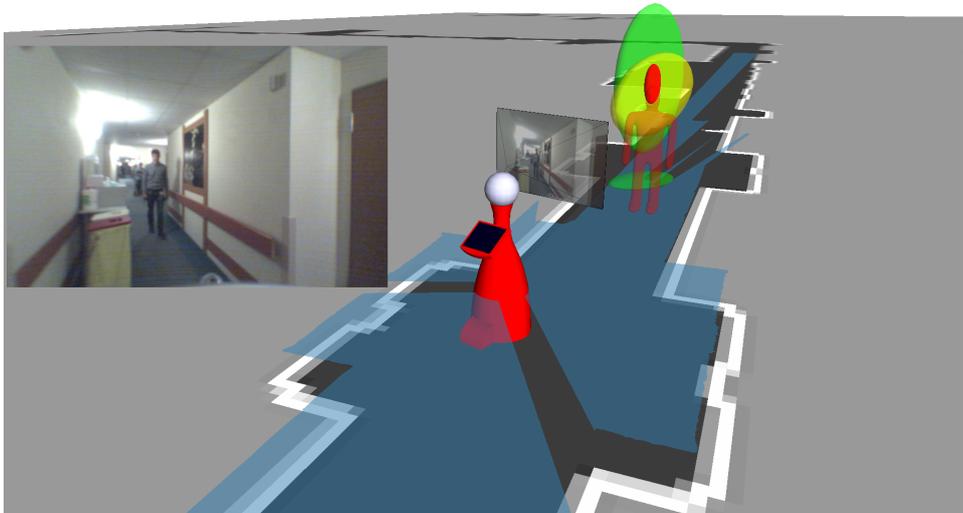


Abbildung 4.19. Hypothesen der einzelnen Detektoren und des Trackers: Visualisierung des Roboters (rot) in der Belegtheitskarte (schwarz-weiß), des Kamerabildes der Frontkamera und des Laserscans (blau) in einer 3D-Ansicht. Das Kamerabild ist links oben noch einmal vergrößert dargestellt. In der 3D-Ansicht ist die laserbasierte Detektion der Person als grüne Kreisfläche und die daraus abgeleitete Hypothese der Kopfposition als grünes Ellipsoid dargestellt. Das gelbe Ellipsoid stellt die Hypothese des kamerabasierten Detektors dar und die getrackte Hypothese ist in rot gezeigt. Zusätzlich ist die wahrscheinlichste Pose der getrackten Person als rote Figur dargestellt, wodurch auch die wahrscheinlichste Orientierung deutlich wird.

schrieben. Der Tracker wird in dieser Arbeit mit einem Kalman-Filter mit siebendimensionalem Zustandsraum verwendet, um die Position (drei Dimensionen), die translatorische Geschwindigkeit (drei Dimensionen) und die Orientierung um die vertikale Achse (eine Dimension) zu tracken. Da nur eine Dimension der dreidimensionalen Orientierung geschätzt wird, beschreiben alle ausgegebenen Hypothesen eine aufrecht stehende bzw. sitzende Person. Es sei angemerkt, dass trotz des intern verwendeten 7D-Zustandsraumes die Ausgaben des Trackers pro Person aus einer 6D-Posenhypothese h_i und einer 3D-Schätzung der translatorischen Geschwindigkeit bestehen.

Eigenschaften der laser- und bildbasierten Hypothesen Wie bereits Abbildung 4.19 zeigt, ergänzen sich die Hypothesen des laserbasierten und des kamerabasierten Detektors sehr gut in Bezug auf die Unsicherheiten der Hypothesen. So ist die Unsicherheit der kamerabasierten Detektionen bzgl. der Distanz zur Kamera relativ groß, wohingegen der laserbasierte Detektor die Distanz sehr genau schätzen kann. Da die Laserscanner die Umgebung jedoch nur in einer Ebene erfassen, ist die laserbasierte Schätzung bzgl. der Größe der Personen besonders unsicher. Diese wird wiederum durch die omnidirektionale

Tabelle 4.5. Eigenschaften von Kameras und Laserscannern und den entsprechenden Detektoren: Gezeigt ist, welche Kriterien die Sensoren bzw. Detektoren vollständig (■), teilweise (■) oder gar nicht (□) erfüllen.

	Kriterium	omnidirektionale Kamera	Laserscanner
Sensorik	Hohe Verfügbarkeit	■	■
	Robustheit bzgl. Beleuchtung	□	■
	Hoher Informationsgehalt	■	■
	Geringe Datenmenge	□	■
	Hohe Aktualisierungsrate	■	■
	Großer horizontaler Öffnungswinkel	■	■
	Großer vertikaler Öffnungswinkel	■	□
Detektor	Geringer Berechnungsaufwand	□	■
	Sicherheit der Distanzschätzung	■	■
	Sicherheit der Höhenschätzung	■	□
	Sicherheit der Detektion	■	■

Kamera wahrgenommen. In Tabelle 4.5 werden noch weitere Eigenschaften der Tiefen- und Bilddaten und der entsprechenden Detektoren gegenübergestellt.

Ein entscheidender Unterschied zwischen dem kamerabasierten und dem laserbasierten Detektor ist der Rechenzeitbedarf. Die kamerabasierte Detektion von Personen wäre bei Verarbeitung aller Detektionsfenster der Auflösungspyramide auf dem zur Verfügung stehenden Rechensystem mit maximal 1.2 Hz möglich. Im Gegensatz dazu belastet der laserbasierte Detektor, selbst bei Verarbeitung aller Laserscans bei 10 Hz, die CPU kaum. Da jedoch beide Detektoren auch Fehldetektionen ausgeben, schätzt der Tracker die Sicherheit der getrackten Hypothesen. Diese reduziert sich über die Zeit, wenn eine Hypothese nicht durch weitere Detektionen bestätigt wird. Außerdem kann nur eine Hypothese, welche von beiden Detektoren bestätigt wurde, einen Sicherheitswert erreichen, ab dem sie als sichere Hypothese gilt. Da die True Positive Rate des laserbasierten Detektors mit 99,1% sehr hoch ist, werden allein durch den laserbasierten Detektor fast alle Personen getrackt. Die Hypothesen werden jedoch durch den Tracker zunächst als unsicher betrachtet, weil der laserbasierte Detektor durchschnittlich alle sieben Sekunden eine Fehldetektion liefert. Die Aufgabe des kamerabasierten Detektors besteht somit vorrangig in der Bestätigung der unsicheren Hypothesen. Deshalb wird der kamerabasierte Detektor in einem Modus betrieben, in welchem nur die Detektionsfenster der Auflösungspyramide klassifiziert werden, deren entsprechende 3D-Positionen sich in der Nähe der unsicheren Hypothesen des Trackers befinden. Dadurch erhöht sich die Aktualisierungsrate des kamerabasierten Detektors für alle vier Kameras auf der Testplattform auf durchschnittlich 2,2 Hz.

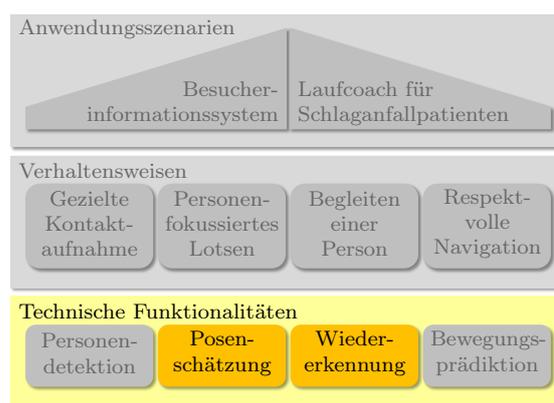
Des Weiteren ist der Erfassungsbereich des kamerabasierten Detektors mit ca. 5 m weitaus geringer als der des laserbasierten Detektors, welcher Personen auch noch in über 10 m Entfernung zum Roboter detektiert. Um auch die in mehr als 5 m Entfernung zum Roboter getrackten Hypothesen, welche nur durch den laserbasierten Detektor gestützt werden, als sicher zu werten, wird eine Heuristik angewendet. Es wird davon ausgegangen, dass es sich bei den laserbasierten Fehldetektionen um statische Objekte handelt. Dementsprechend wird die Detektionssicherheit der Hypothesen des Trackers, welche sich mit mindestens $v_{min} = 0,5 \frac{m}{s} = 1,8 \frac{km}{h}$ bewegen, erhöht. Somit werden Personen, welche sich in 5 m Entfernung zum Roboter befinden, kamera- und laserbasiert detektiert und durch den Tracker als sichere Hypothesen ausgegeben. Personen, welche sich in einer Distanz zwischen 5 m und 10 m zum Roboter bewegen, werden zwar nur durch den laserbasierten Detektor erfasst, werden jedoch aufgrund der Heuristik auch als sichere Hypothesen angesehen. Personen, welche sich in über 10 m Entfernung zum Roboter befinden, und Personen, welche stillstehen und sich in mehr als 5 m Entfernung zum Roboter befinden, werden nicht durch das beschriebene System detektiert.

4.4. Zusammenfassung

Das robuste Tracking von Personen in der Nähe des Roboters ist eine wesentliche Grundlage für eine sozialverträgliche Navigation. In diesem Kapitel wurde ein System beschrieben, mit welchem Personen durch einen mobilen Roboter getrackt werden können. Neben der Position und der translatorischen Geschwindigkeit der Personen liefert dieses System auch noch die Orientierung des Oberkörpers der getrackten Personen und die Information, welche Gehhilfe die Personen verwenden. Zu diesem Zweck werden aus den Laserdaten neuartige Merkmale extrahiert, welche sehr effizient berechnet werden können und zur Detektion und Unterscheidung von Gehhilfen geeignet sind. Die Oberkörperorientierung wird durch den kamerabasierten Detektor mit einer Winkelauflösung von 45° erkannt. Um dies zu erreichen, wurde eine Methode zur Generierung geeigneter Trainingsdaten entwickelt. Außerdem wurde das Verfahren [Dalal, 2005] um einen effizienten SVM Tree zur Mehrklassen-Klassifikation erweitert. Auch wenn die Kombination eines Entscheidungsbaumes mit SVMs nicht neu ist, so konnte durch das im Rahmen dieser Arbeit entwickelte Training eines solchen Klassifikators insbesondere bei hochdimensionalen Merkmalsräumen, wie sie bei der kamerabasierten Personendetektion vorliegen, eine deutliche Verbesserung der Klassifikationsleistung erreicht werden.

5

Erscheinungsbasierte Oberkörpermodellierung und Wiedererkennung



In Kapitel 4 wurden zwei diskriminative Verfahren zur Personendetektion mit Positions- und Orientierungsschätzung sowie das echtzeitfähige Tracking der Posenhypothesen beschrieben. In diesem Kapitel wird untersucht, ob die Posenschätzung der getrackten Hypothesen mittels eines generativen, erscheinungsbasierten Verfahrens verbessert werden kann. Darüber hinaus wird die Eignung des erscheinungsbasierten Modells zur Wiedererkennung von Personen evaluiert. Im Ergebnis wird jedoch deutlich, dass die hier präsentierten Verfahren nicht zum praktischen Einsatz auf dem Roboter kommen. Bei Bedarf kann der Leser dieses Kapitel daher durchaus überspringen, ohne dass sich daraus in den nachfolgenden Kapiteln Verständnisprobleme ergeben.

Zur Posenschätzung und Wiedererkennung wird ein 3D-Formmodell so angepasst, dass dieses möglichst genau die beobachtete Erscheinung der jeweiligen getrackten Person nachbildet. Im Wesentlichen werden die Modellparameter θ gesucht, welche mit maximaler Wahrscheinlichkeit $P(\mathbf{I}|\theta)$ zur Aufnahme des Kamerabildes \mathbf{I} geführt hätten. Zusätzlich zu den getrackten Posenparametern, welche in Kapitel 4 beschrieben sind, umfassen die Parameter θ des erscheinungsbasierten Modells auch noch die Gelenkstellungen der Arme und des Kopfes, sowie Farbmodelle des gesamten Oberkörpers. Durch die Modellierung der Farbe, wird eine Wiedererkennung von Personen ermöglicht. Eine robuste Wiedererkennung ist notwendig, um den aktuellen Nutzer des Roboters, beispielsweise während dieser gelotst oder begleitet wird, von anderen Patienten bzw. Besuchern un-

terscheiden zu können. Die Voraussetzung für eine solche Wiedererkennung ist jedoch eine ausreichend gute Schätzung der Gelenkstellungen, weshalb diese im Fokus dieses Kapitels liegt. Im nächsten Abschnitt wird der aktuelle Forschungsstand bzgl. Verfahren zur kamerabasierten Schätzung von Gelenkstellungen aufgearbeitet. Danach wird das im Rahmen dieser Arbeit entwickelte Verfahren zur ercheinungsbasierten Optimierung der getrackten Posenhypothesen vorgestellt und dessen Evaluation beschrieben. Den Abschluss dieses Kapitels bildet eine kritische Bewertung dieses Verfahrens in Bezug auf die Anwendungsszenarien dieser Arbeit.

5.1. Bildbasierte Schätzung von Gelenkstellungen

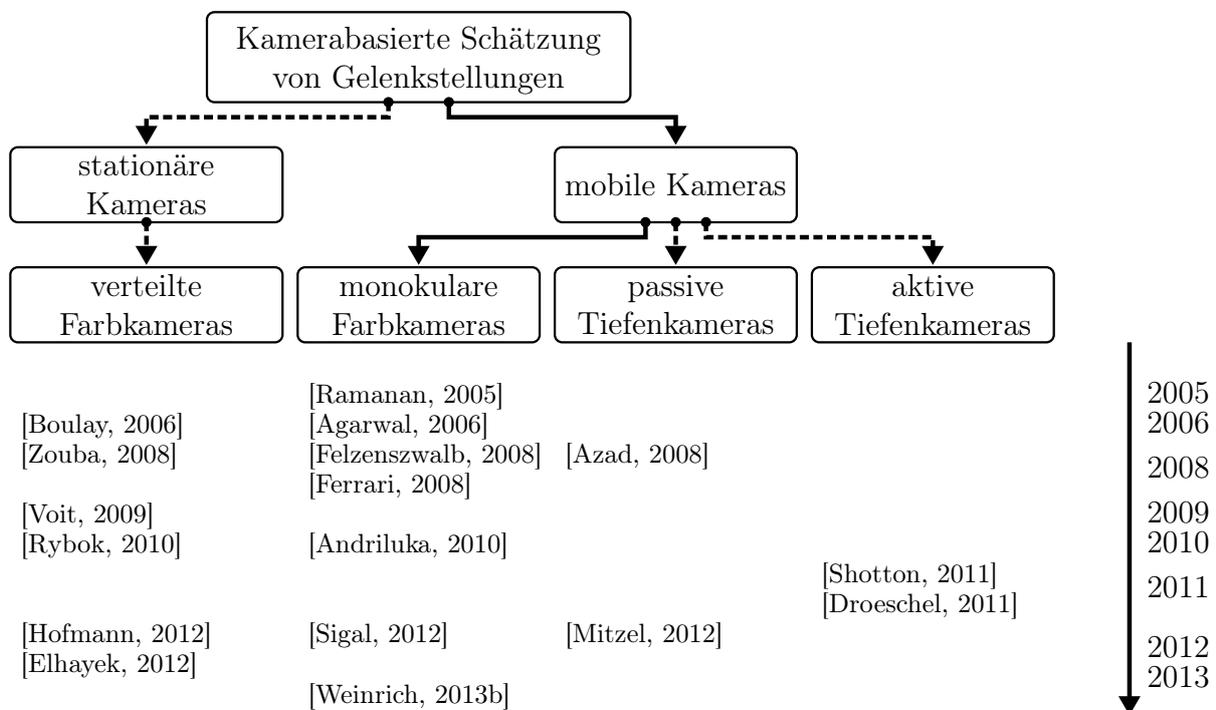


Abbildung 5.1. Arbeiten zu kamerabasierter Personendetektion: Darstellung neuartiger Verfahren (schwarz) und darauf aufbauender Arbeiten (grau) zu kamerabasierter Personendetektion durch verschiedene Kameratypen und -anordnungen nach Jahren

Im Gegensatz zu den Verfahren zur kamerabasierten Personendetektion mit Orientierungsschätzung, welche in Abschnitt 4.2.1 vorgestellt wurden, liegt der Fokus in diesem Abschnitt auf Verfahren zur Schätzung von 3D-Gelenkstellungen. Dies stellt aufgrund der hohen Beweglichkeit der menschlichen Gelenke und vielfältigen Erscheinungen eine große Herausforderung dar. Um Mehrdeutigkeiten aufzulösen, benötigen viele Verfahren mehrere Ansichten einer Person mittels verteilter Kameras [Zouba, 2008, Voit, 2009, Rybok, 2010, Hofmann, 2012, Elhayek, 2012]. Dies ist jedoch auf einem mobilen Roboter

nicht möglich. Eine Alternative, um die Mehrdeutigkeiten mit einer einzelnen Kamera aufzulösen, sind aktive [Shotton, 2011, Droschel, 2011] und passive [Azad, 2008, Mitzel, 2012] Tiefenkameras. Allerdings haben sowohl die passiven Stereokameras als auch aktive Tiefenkameras wie die KinectTM aufgrund des notwendigen Basisabstands Abmessungen, welche es praktisch unmöglich machen, diese in einem omnidirektionalen System auf dem Roboter zu verwenden. Eine einzelne KinectTM hat jedoch für die Anwendungsszenarien dieser Arbeit einen zu geringen Öffnungswinkel. Weniger kritisch, dennoch erwähnenswert im Zusammenhang mit mobilen Robotern, ist der meist höhere Stromverbrauch aktiver Systeme gegenüber passiven Kameras. Außerdem werden aktive Tiefenkameras durch externe Infrarotlichtquellen, wie z. B. Sonnenlicht, gestört.

Deshalb sind für diese Arbeit Verfahren relevant, welche auf monokularen Kamerabildern arbeiten. Dies hat neben den technischen Gründen auch historische Ursachen in der Projektbearbeitung. Die Methoden zur Posenschätzung können in Top-Down (generative) und Bottom-Up Verfahren (diskriminative) eingeteilt werden [Brauer, 2012]. Bei [Ramanan, 2005, Felzenszwalb, 2008, Ferrari, 2008] handelt es sich um drei Bottom-Up Verfahren, bei welchen erst Pictorial Structures zur Detektion einzelner Körperteile genutzt werden, und danach wird mittels Belief Propagation die 2D-Pose der Personen inferiert. Für die im Rahmen dieser Arbeit entwickelten Verhaltensweisen ist jedoch die 2D-Pose der Personen im Bild nicht ausreichend. Die 3D-Posen der Personen sind erforderlich, weil der Abstand zu den Personen und deren Orientierung zum Roboter wesentlichen Einfluss auf alle vier in dieser Arbeit bearbeiteten Verhaltensweisen zur Förderung einer sozialverträglichen Navigation haben.

Bei [Agarwal, 2006] handelt es sich um ein entsprechendes diskriminatives, einzelbildbasiertes Verfahren zur Schätzung der 3D-Pose mittels Regression. Allerdings wird die Silhouette der Person als Merkmal verwendet. Dadurch kommt es zu Mehrdeutigkeiten, und die Eigenbewegung des Roboters würde die Hintergrundsubtraktion erschweren. Darüber hinaus ist es bei den diskriminativen Verfahren erforderlich, dass zum Training des Klassifikators bzw. Regressors die Merkmale verschiedenster Posen der Personen vorliegen, um eine ausreichende Generalisierung zu erreichen. Auch die Fehler bei der Merkmalsextraktion, wie bspw. bei der Bildsegmentierung in [Agarwal, 2006], müsste der Trainingsdatensatz abdecken. Damit sind die diskriminativen Verfahren meist effizient, jedoch weniger geeignet, um hochdimensionale Posen zu schätzen und werden deshalb eher zur 2D Posenschätzung angewendet.

Bei generativen Verfahren wird beabsichtigt, die Bildentstehung zu modellieren. Der Zweck ist jedoch weniger die Erzeugung realistischer Bilder gelenkiger Personen, als die Ermittlung der Wahrscheinlichkeit, dass ein Bild bei gegebener Posenhypothese entstan-

den sein kann. Die menschliche Pose wird meist durch den Parametersatz eines kinematischen Baumes beschrieben, welcher die Torsopose als Wurzelknoten und die Gelenkwinkel der Körperteile relativ zum übergeordneten Knoten spezifiziert. Solche Modelle können beliebige Posen beschreiben und auch physikalische Einschränkungen berücksichtigen.

Die Vielzahl der Gelenke erfordert eine Optimierung in einem sehr hochdimensionalen Raum, und die Verfahren sind meist stark abhängig von der Initialisierung der Parameter. Deshalb werden auch hybride Verfahren [Andriluka, 2010] zur 3D-Posenschätzung eingesetzt. Die diskriminative Komponente dient vorrangig zur Einschränkung des Suchraumes bzw. der Initialisierung des generativen Verfahrens. So wird in [Andriluka, 2010] ausgehend von den diskriminativen 2D Körperteildetektionen auf die Wahrscheinlichkeit der 3D-Posen geschlossen. Bei dieser 2D-zu-3D-Dimensionserhöhung gibt es jedoch sehr viele Mehrdeutigkeiten. Deshalb wird in [Andriluka, 2010] zur Einschränkung des Suchraumes ein hierarchical Gaussian Process Latent Variable Model (hGPLVM) [Lawrence, 2007] verwendet, um die a-priori-Wahrscheinlichkeit über mögliche Posen (die Kombination der Gelenkstellungen) und über die Zeit (Abfolge von Posen), z. B. bei einer Gehbewegung, zu repräsentieren. Die Mehrdeutigkeiten werden folglich aufgelöst, indem alle Gelenkstellungen innerhalb einer Bildsequenz gesamtheitlich betrachtet werden.

Allerdings sind viele Verfahren, wie auch [Andriluka, 2010], für die Offline-Auswertung von Bildsequenzen konzipiert und nicht echtzeitfähig. Sie lassen sich deshalb nicht in dieser Arbeit anwenden. Für die Anwendungsszenarien dieser Arbeit ist es nicht erforderlich, dass die 3D-Posen für jedes einzelne Kamerabild berechnet werden. Stattdessen sind Verfahren interessant, welche auch funktionieren, wenn bei nur geringer verfügbarer Rechenleistung (z. B. infolge anderer rechenintensiver Prozesse auf dem Roboter) nicht jedes Kamerabild verarbeitet werden kann, sondern mehrere Bilder übersprungen werden müssen, bis wieder die Bearbeitung eines Bildes möglich ist. Während einzelne Bilder übersprungen werden, stehen noch die getrackten Hypothesen (Kapitel 4) ohne Gelenkstellungen zur Verfügung. Mittels eines generativen Verfahrens soll letztlich bei einzelnen Bildern die Unsicherheiten der getrackten Hypothesen verringert und die Gelenkstellungen der Oberarme geschätzt werden. Für die Interesseschätzung und die Wiedererkennung reicht es aus, wenn dies mit geringer Aktualisierungsrate durchgeführt wird. Für die erscheinungsbasierte Schätzung der Oberkörperpose und der Gelenkstellungen der Arme sollen die durchgängig getrackten Hypothesen zur Initialisierung dienen, und die optimierten Modellparameter können danach für die Interesseschätzung etc. verwendet werden. Dafür ist es jedoch erforderlich, dass sich aus den Modellparametern die Oberkörperpose und die Gelenkstellungen der Arme als kinematischer Baum extrahieren lassen und umgekehrt.

5.2. Appearance Model zur Hypothesenoptimierung

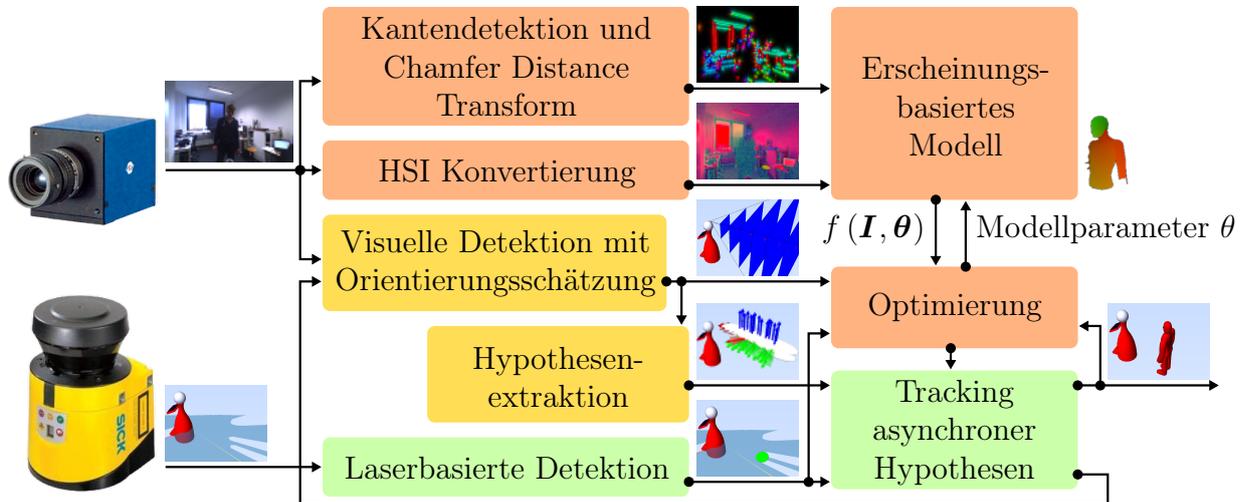


Abbildung 5.2. Einordnung der erscheinungsbasierten Optimierung: Der asynchrone Multi-Hypothesen-Tracker und der laserbasierte Personendetektor (grün) haben sehr geringe Rechenanforderungen und eine hohe Aktualisierungsrate, um kontinuierlich mehrere Personen zu tracken. Der kamerabasierte Detektor (gelb) arbeitet mit geringerer Aktualisierungsrate. Er dient zur Bestätigung der unsicheren Hypothesen des Trackers und gibt den getrackten Hypothesen eine Orientierung. Die verbleibenden Module (orangefarben) werden zur erscheinungsbasierten Optimierung der Posenhypothesen und zum Adaptieren von personenspezifischen Farbmodellen zur Wiedererkennung von Personen verwendet.

Der in Kapitel 4 beschriebene Tracker liefert für jede getrackte Person i eine Normalverteilung $h_i(t)$ im 6D-Posenraum (siehe Abschnitt D.5). Zur Optimierung jeder Posenhypothese wird die Particle Swarm Optimization (PSO) [Kennedy, 1995] angewendet. Die Partikel, welche die Modellparameter θ des dreidimensionalen Oberkörpermodells repräsentieren, werden entsprechen der Normalverteilungen der Hypothese $h_i(t)$ initialisiert. Um die Vielfältigkeit der menschlichen Erscheinung zu modellieren, hat dieses Modell 14 Freiheitsgrade (DOF). Die meisten Modellparameter θ werden somit nicht durch die getrackten Hypothesen $h_i(t)$ initialisiert. Theoretisch könnte der Tracker um weitere Dimensionen erweitert werden. Die Aktualisierungsrate, mit der dieses Verfahren die optimierten Hypothesen liefern wird, ist jedoch im Verhältnis zu der Unsicherheit des Bewegungsmodells bzgl. dieser Freiheitsgrade relativ gering. Der Aufwand zum Tracking dieser DOF wäre nicht verhältnismäßig, weshalb zur Initialisierung des Modells bzgl. dieser Freiheitsgrade eine a-priori-Verteilung verwendet wird.

Das Appearance Model repräsentiert eine Funktion $f(I, \theta)$, welche angibt, wie gut das Modell bei Parametrierung θ mit dem Bild I übereinstimmt. Diese Matching-Funktion

$f(\mathbf{I}, \boldsymbol{\theta})$ stellt das Optimierungskriterium der PSO dar. Sie korreliert mit der bedingten Wahrscheinlichkeit $P(\mathbf{I}|\boldsymbol{\theta})$, dass das Bild \mathbf{I} aufgenommen werden kann, wenn sich eine Person mit gegebener Pose $\boldsymbol{\theta}$ vor der Kamera befindet.

Im nächsten Abschnitt werden die Modellparameter des ercheinungsbasierten Modells spezifiziert, und es wird beschrieben, wie die Matching-Funktion $f(\mathbf{I}, \boldsymbol{\theta})$ berechnet wird.

Erscheinungsbasiertes Modell

Das ercheinungsbasierte Modell basiert auf einem 3D-Modell eines durchschnittlichen, geschlechtsunspezifischen Oberkörpers (Abb. 5.3b). Die 14 veränderlichen Freiheitsgrade des Modells sind die Torsoposition (3 DOF) und Oberkörperorientierung (1 DOF), die Kopfdrehung und Neigung (2 DOF), die Gelenkstellung beider Schultern (6 DOF) und die Beugung beider Ellenbogen (2 DOF). Zusätzlich können mehrere Farbmodelle für verschiedene Personen gewählt werden. Das Oberkörpermodell wurde mit MakeHumanTM [Bastioni, 2008] erzeugt. Es hat keine Hände, und mehrere Freiheitsgrade (DOF) des Modells, wie Geschlecht, Alter, Muskelmasse, Gewicht, Brustumfang etc., sind unveränderbar, da ein komplexeres 3D Modell zu einer aufwändigeren Optimierung führen würde.

Zur Berechnung von $f(\mathbf{I}, \boldsymbol{\theta})$ werden die nachfolgend beschriebenen Merkmale des Bildes \mathbf{I} ausgewertet. Die meisten Merkmale werden mittels spezieller Shader Programme auf der Grafikkarte (GPU) berechnet, da diese gut für die Verarbeitung des 3D-Modells geeignet ist.

Kanten Die Varianz der Bildgradienten ist, im Gegensatz zur Struktur und der Farbe menschlicher Kleidung, weitaus geringer, und die Bildgradienten werden wesentlich durch die menschliche Pose beeinflusst. Außerdem zeigen Detektionsverfahren wie [Dalal, 2005] die Relevanz dieser Merkmale. Das Kantenmodell vergleicht die erwarteten Orientierungen der Gradienten \mathbf{O}^{mod} des 3D-Modells pixelweise mit den Gradientenorientierungen des aufgenommenen Bildes \mathbf{O}^{obs} . Der Fehler zwischen beobachteter Gradientenorientierung und erwarteter Gradientenorientierung wird mittels einer Gauß-Funktion in einen Wert für die Übereinstimmung umgerechnet. Die entsprechenden Magnituden der Modellgradienten \mathbf{M}^{mod} und der Bildgradienten \mathbf{M}^{obs} fließen als Gewichte in die Verrechnung der Übereinstimmung aller Pixel $f_{Edg}(\mathbf{I}, \boldsymbol{\theta})$ ein. Die erwarteten Gradientenorientierungen \mathbf{O}^{mod} und Magnituden \mathbf{M}^{mod} (Abb. 5.4c) werden durch spezielle Vertex- und Pixel-Shader auf der GPU berechnet. Die Funktionsweise dieser Shader und die Verrechnung mit den Magnituden ist im Anhang E.1 erklärt.



(a) Frontaler Bildausschnitt der omnidirektionalen Kamera



(b) Gelenkiges 3D-Modell



(c) Invers gerenderte Textur der Aufnahme



(d) Modell mit aufgenommener Textur



(e) Durchschnittsfarbe eines personenspezifischen Modells

Abbildung 5.3. Erscheinungsbasiertes Modell und inverses Rendern: Beim typischen Rendern wird ausgehend von Informationen über eine Szene ein möglichst realistisches Bild erzeugt. Im Gegensatz dazu ist das Ziel des inversen Renderns, Szeneninformationen aus einem realen Bild (a) zu extrahieren. So kann unter Verwendung eines gelenkigen 3D-Modells (b) die Textur der Person (c) berechnet werden. Also die Textur, mit welchem das 3D-Modell beim typischen Rendern wie die Person in dem aufgenommenen Bild erscheinen würde (vgl. (a) und (d)). Die invers gerenderte Textur (c) einer Person zeigt nur Teilflächen des menschlichen Oberkörpers. Werden über die Zeit mehrere Ansichten der Person beobachtet, können die Texturen zur Adaption eines vollständigen Oberkörpermodells (e) verwendet werden. Dieses Modell wird zur Wiedererkennung der Person genutzt.

Um die tatsächlichen Gradienten aus dem aufgenommenen Bild zu extrahieren, werden einfache 2×2 Roberts-Operatoren zur Detektion horizontaler und vertikaler Kanten angewendet. Der Kantendetektor (Abb. 5.2) berechnet, ausgehend von dem aufgenommenen Bild I , ein Bild mit den Gradientenorientierungen O^{obs} und eines mit den Magnituden M^{obs} ähnlich zu [Dalal, 2005]. Um den Einfluss des Pixelrauschens zu reduzieren und relevante Kanten zu verstärken, wird ein nicht-linearer Filter auf dem Magnitudenbild

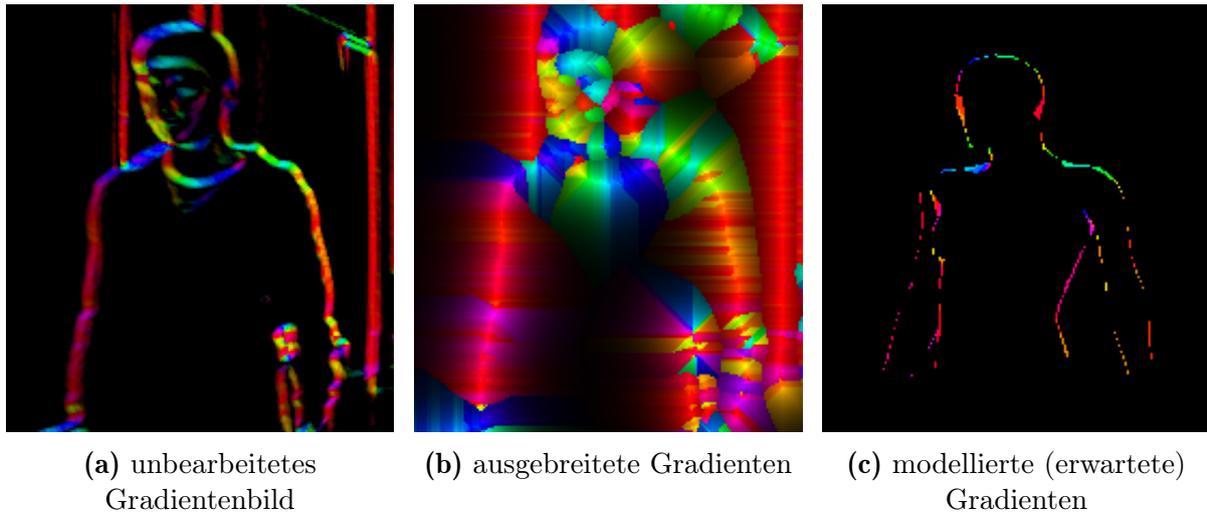


Abbildung 5.4. Kanten: Die Gradienten des aufgenommenen Bildes (a) werden basierend auf der Chamfer Distance Transformation ausgebreitet (b). Das erhöht die Robustheit bzgl. Abweichungen der modellierten Gradienten (c). In allen drei Bildern ist die Gradientenorientierung durch die Farbe und die Magnitude durch die Intensität codiert.

\mathcal{M}^{obs} angewendet, welcher niedrige Werte unterdrückt und höhere Werte verstärkt.

Um den später beschriebenen Optimierungsvorgang zu unterstützen, ist es wichtig, dass die Matching-Funktion $f_{Edg}(\mathbf{I}, \boldsymbol{\theta})$ möglichst glatt ist. Dies wird erreicht, indem die Magnituden und Gradienten ähnlich der Chamfer-Distanz-Transformation [Bailey, 2004] räumlich ausgebreitet werden, wobei die Gradientenorientierung von der höchsten Magnitude in der Umgebung des Pixels übernommen wird. Dieser Algorithmus erlaubt es, die Kanteninformationen bei konstantem Berechnungsaufwand beliebig weit auszubreiten. Abbildung 5.4b zeigt das resultierende Gradientenbild, welches zum Matching mit den erwarteten Kanten (Abb. 5.4c) verwendet wird. Durch diese Glättung des Magnitudenbildes wird auch die Matching-Funktion $f_{Edg}(\mathbf{I}, \boldsymbol{\theta})$ geglättet.

Segmentierung Das zuvor beschriebene Kantenmodell wird beeinflusst durch stark strukturierte Kleidung bzw. Hintergrund, weil die Konturen der Person nicht von deren Struktur oder der Struktur des Hintergrundes unterschieden werden kann. Deshalb bewertet dieses Modell $f_{SEG}(\mathbf{I}, \boldsymbol{\theta})$ die Vordergrund/Hintergrund Segmentierung des aufgenommenen Bildes \mathbf{I} entsprechend der Modellparameter $\boldsymbol{\theta}$, basierend auf der Farbverteilung beider Segmente (Abb. 5.5).

Dazu werden zwei 2D Farbhistogramme (über Farbton und Sättigung) über den Vordergrundausschnitt, welcher entsprechend $\boldsymbol{\theta}$ den Oberkörper zeigt, und über einen Hintergrundausschnitt, welcher einen Saum um diesen Ausschnitt darstellt, berechnet und ba-

sierend auf der Bhattacharyya-Distanz [Bhattacharyya, 1946] verglichen. Die Matching-Funktion $f_{SEG}(\mathbf{I}, \boldsymbol{\theta}) = 1,0 - BD(p_F(H, S), p_B(H, S))$ erreicht hohe Werte, wenn die Modellparameter $\boldsymbol{\theta}$ dazu führen, dass sich die Farbverteilung des Oberkörpers von dem Hintergrund abhebt. Da die Funktion auf den Farbhistogrammen ganzer Bildbereiche basiert, führen kleinere Änderungen der Modellparameter $\boldsymbol{\theta}$ auch zu kleinen Änderungen der Matching-Funktion $f_{Seg}(\mathbf{I}, \boldsymbol{\theta})$, und diese ist entsprechend glatt.

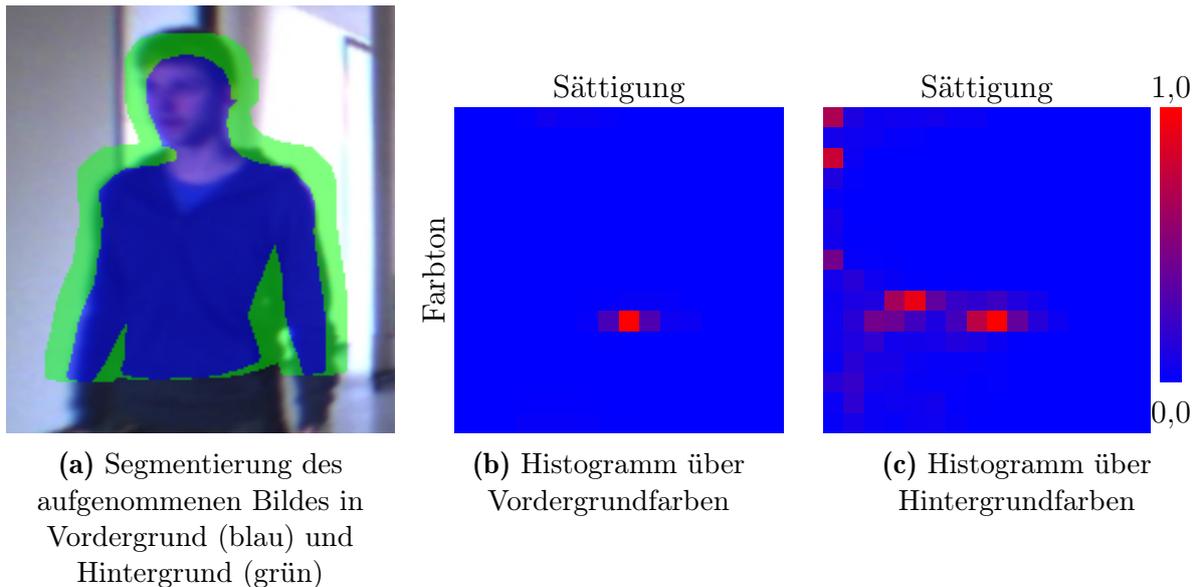


Abbildung 5.5. Segmentierung: Das appearancebasierte Modell wird zur Segmentierung des aufgenommenen Bildes verwendet (a). Tendenziell ist die Anpassung des Modells umso besser, je stärker sich das Histogramm über die Vordergrundfarben (b) von dem Histogramm über die Hintergrundfarben (c) unterscheidet.

Farbe Im Gegensatz zu den beiden zuvor beschriebenen Merkmalen, ist die Farbe ein Merkmal, bzgl. welchem sich die Oberkörper verschiedener Personen meist unterscheiden. Aus diesem Grund werden in dieser Arbeit die Farben des Oberkörpers von Personen durch Texturen¹ modelliert und zur Wiedererkennung von Personen eingesetzt.

Zur personenunabhängigen Optimierung der Modellparameter wird ein universelles Farbmodell \mathbf{C}_0 eingesetzt. Darüber hinaus werden zur Wiedererkennung von Personen personenspezifische Farbmodelle $\mathbf{C}_{m>0}$ verwendet. Bevor die Verwendung der verschiedenen Farbmodelle genauer erklärt wird, erfolgt eine Erläuterung der Matching-Funktion $f_{Col}(\mathbf{I}, \boldsymbol{\theta}, \mathbf{C}_m)$ für ein beliebiges Farbmodell \mathbf{C}_m .

¹Im Kontext dieser Arbeit ist Textur als Begriff aus der Computergrafik, für Bilder auf einer 3D-Oberfläche, zu verstehen.

Ein wesentlicher Verarbeitungsschritt zur Berechnung von $f_{Col}(\mathbf{I}, \boldsymbol{\theta}, \mathbf{C}_m)$ ist die Ermittlung der Oberkörpertextur, ausgehend von dem Kamerabild \mathbf{I} und den Modellparametern $\boldsymbol{\theta}$.

Diese Berechnung erfolgt mittels inversem Rendern. Das typische Rendern hat das Ziel ausgehend von einem 3D-Modell, einer Textur und anderen Merkmalen einer Szene ein möglichst realistisches Bild zu erzeugen. Im Gegensatz dazu, liegt hier das real aufgenommene Bild \mathbf{I} bereits vor. Unter Nutzung des 3D-Modells, entsprechend der angenommenen Modellparameter $\boldsymbol{\theta}$, wird ein Attribut der Szene, die Oberkörpertextur, berechnet. Es wird somit die gesuchte Textur bestimmt, welche bei gegebenen Parametern $\boldsymbol{\theta}$ zu dem gegebenen Bild \mathbf{I} geführt hätte. Das Ergebnis für das inverse Rendern eines Kamerabildes (Abb. 5.3a) und eines 3D-Modells (Abb. 5.3b), ist beispielhaft in Abbildung 5.3c gezeigt. Eine genauere Erklärung, wie das inverse Rendering auf der Grafikkarte mit den vorhandenen Rendermethoden durchgeführt wird, ist im Anhang E.2 beschrieben.

Ein Farbmodell \mathbf{C}_m wird repräsentiert durch eine Gauß-Verteilung im HSI-Farbraum für jedes Pixel der Oberkörpertextur. Die Mittelwertfarben eines solchen Farbmodells sind in Abbildung 5.3e gezeigt. Die Parameter der Farbverteilung werden online mittels der Maximum-a-posteriori-Methode (MAP) geschätzt. Wenn sich eine getrackte Person vor der Kamera bewegt und so verschiedene Bereiche der Oberfläche ihres Oberkörpers bei unterschiedlichen Beleuchtungen sichtbar werden, steigt die Qualität des Farbmodells. Die Übereinstimmung der aktuell aufgenommenen Textur mit einem Farbmodell wird bestimmt, indem über alle sichtbaren Pixel der Textur die durchschnittliche Likelihood, dass die beobachtete Farbe zum Modell gehört, berechnet wird.

Initial wird das universelle Farbmodell \mathbf{C}_0 zur Optimierung der Hypothesen verwendet. Danach werden die optimierten Modellparameter $\boldsymbol{\theta}$ verwendet, um das universelle Modell \mathbf{C}_0 anzupassen und als ein personenspezifisches Modell $\mathbf{C}_{m>0}$ zu speichern. Um einer durchgängig getrackten Hypothese immer dasselbe personenspezifische Farbmodell m zuzuordnen, wird eine Zuordnung der Track-IDs i des Trackers auf die Personen-IDs m gespeichert. Wenn der Tracker eine neue Track-ID i ausgibt, wird wieder das universelle Farbmodell \mathbf{C}_0 zur Optimierung verwendet. Danach wird unter Verwendung der optimierten Parameter $\boldsymbol{\theta}$ geprüft, ob eines der personenspezifischen Farbmodelle $\mathbf{C}_{m>0}$ eine höhere Wahrscheinlichkeit als das universelle Modell \mathbf{C}_0 erreicht. Dies würde als einfache ansichtsbasierte Wiedererkennung gewertet. Die neue Track-ID würde der gleichen Personen-ID zugeordnet und das spezifische Farbmodell würde weiter adaptiert. Falls jedoch das universelle Farbmodell die höchste Wahrscheinlichkeit erreicht, wird dieses adaptiert und ein neues spezifisches Farbmodell erzeugt. Außerdem würde der neuen Track-ID die Personen-ID des neuen Modells zugeordnet.

Oberkörperdetektionen Wie in Kapitel 4 beschrieben, basiert das Tracking der Personen auf kamera- und laserbasierten Personendetektionen. In diesem Absatz wird beschrieben, wie die diskreten Detektionen des kamerabasierten Oberkörperdetektors auch die erscheinungsbasierte Optimierung unterstützen.

Um die Matching-Funktion $f_{HOG}(\mathbf{I}, \boldsymbol{\theta})$ zu berechnen, werden die Modellparameter $\boldsymbol{\theta}$ in die Auflösungspyramide mit den Detektionsergebnissen des kamerabasierten Detektors (Abschnitt 4.2) transformiert und dann mittels Interpolation aus den Wahrscheinlichkeiten der benachbarten Oberkörperorientierungen, Pyramidenschichten sowie horizontaler und vertikaler Oberkörperposition berechnet.

Laserbasierte Detektionen Wie schon die zuvor beschriebenen Oberkörperdetektionen, werden auch die laserbasierten Detektionen (Abschnitt 4.1) während der Optimierung verwendet. Die Matching-Funktion $f_{Leg}(\mathbf{S}, \boldsymbol{\theta})$ ergibt sich direkt aus der Wahrscheinlichkeit der Gauß-Verteilungen der laserbasierten Torsohypothesen an der Torsoposition, welche sich aus $\boldsymbol{\theta}$ ergibt.

Die zuvor beschriebenen Matching-Funktionen werden mittels Gamma-Operator, welcher insbesondere aus der Fuzzy Logik bekannt ist, zu einer Gesamtfunktion $f(\mathbf{I}, \mathbf{S}, \boldsymbol{\theta})$ kombiniert. Es ist ein Kompromiss aus dem Produkt und dem gewichteten Mittelwert der Teilfunktionen:

$$f(\mathbf{I}, \mathbf{S}, \boldsymbol{\theta}) = \gamma \left(\sqrt[5]{\prod_{M \in \{\text{Edg, Seg, Col, HOG, Leg}\}} \omega_M f_M(\mathbf{I}, \mathbf{S}, \boldsymbol{\theta})} \right) + (1 - \gamma) \left(\frac{1}{5} \sum_{M \in \{\text{Edg, Seg, Col, HOG, Leg}\}} \omega_M f_M(\mathbf{I}, \mathbf{S}, \boldsymbol{\theta}) \right) \quad (5.1)$$

Der verwendete Wert für γ und den Gewichtungsfaktor ω_M für jedes der fünf Modelle ist im Anhang in Abschnitt E.3 spezifiziert.

Für jede getrackte Hypothese wird die entsprechende Parameterkonfiguration $\boldsymbol{\theta}$ mittels Particle Swarm Optimization (PSO) [Kennedy, 1995] optimiert. Dabei wird die Matching-Funktion $f(\mathbf{I}, \mathbf{S}, \boldsymbol{\theta})$ über dem Parameterraum Θ als Optimierungskriterium verwendet. Der Partikelschwarm besteht aus 20 Partikeln. Jedes repräsentiert eine 14-dimensionale Parameterkonfiguration $\boldsymbol{\theta}$. Der Partikelschwarm wird entsprechend der Gaußverteilung der entsprechenden Hypothese initialisiert. Die zehn Parameter der Gelenkstellungen, welche nicht durch die Hypothese beschrieben werden (Kopfdrehung und -neigung sowie

Schulter und Ellenbogen), werden entsprechend vordefinierter Wahrscheinlichkeitsverteilungen initialisiert. Auch die Geschwindigkeitsvektoren der Partikel werden basierend auf vordefinierten Wahrscheinlichkeiten initialisiert. Dann wird die PSO für maximal 20 Iterationen durchgeführt.

5.3. Experimente

Die Eigenschaften der Matching-Funktion $f(\mathbf{I}, \mathbf{S}, \boldsymbol{\theta})$ haben wesentlichen Einfluss auf den Erfolg der Optimierung und sind deshalb im Anhang in Abschnitt E.3 aufgearbeitet. Die in dieser Arbeit relevanten Parameter des Oberkörpermodells sind die Oberkörperposition und -orientierung. Die übrigen Parameter sind in dieser Arbeit nicht relevant, müssen jedoch dennoch optimiert werden, da die Gelenkstellungen der Schultern und Ellenbogen einen hohen Einfluss auf die Erscheinung einer Person haben und ohne die Anpassung dieser Gelenke auch die Oberkörperposition und -orientierung nicht geschätzt werden könnte.

Im Folgenden soll die Qualität der Oberkörperposenschätzung und Wiedererkennung evaluiert werden. Während der Aufnahme der Evaluationsdaten haben sich drei Personen wiederholt durch einen Evaluationsbereich vor dem Roboter bewegt (Abb. 5.6). Zur Erhebung von Ground-Truth-Daten wurden die 2D-Positionen der Personen mit einem externen System aus verteilten Laserscannern (basierend auf [Schenk, 2011]) getrackt. Die Größe jeder Person wurde einmalig manuell gemessen. Da sich die Probanden im Evaluationsbereich immer nur in Richtung der Oberkörperorientierung bewegen durften, lässt sich die Oberkörperorientierung für die Referenzmessung direkt aus der Bewegungsrichtung ableiten.

Zu Beginn des Experiments waren die verwendeten Farbmodelle wie folgt initialisiert. Das universelle Farbmodell \mathbf{C}_0 wurde auf Basis der Erscheinung von fünf verschiedenen Personen, welche nicht in dem Evaluationsdatensatz enthalten waren, gelernt. Außerdem existierten zwei personenspezifische Farbmodelle \mathbf{C}_1 und \mathbf{C}_2 , welche für zwei Personen, die auch nicht im Evaluationsdatensatz enthalten waren, gelernt wurden. Dadurch soll getestet werden, ob die drei Probanden tatsächlich als unbekannte Personen erkannt werden und nicht einem der beiden personenspezifischen Modelle zugeordnet werden. Die Testpersonen haben den Detektionsbereich mehrmals betreten. Beim ersten Mal sollten sie als unbekannte Person klassifiziert werden und danach sollten sie wiedererkannt werden.

Während des Experiments wurden die Personen durch den, im vorigen Kapitel beschriebenen, Tracker getrackt und das erscheinungsbasierte Verfahren wurde zur Optimierung

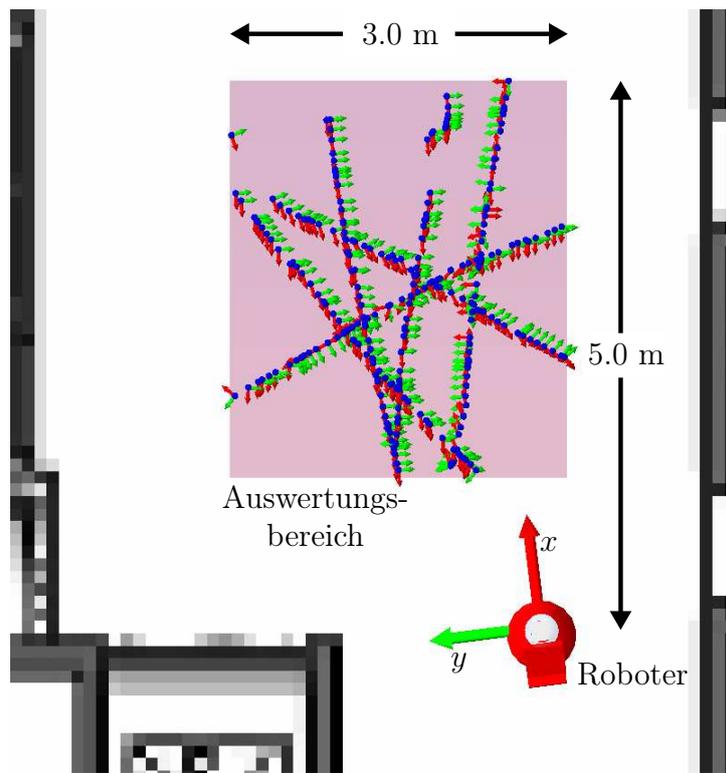


Abbildung 5.6. Versuchsaufbau: Versuchsaufbau während des Experiments aus der Vogelperspektive. Die getrackten und optimierten Posenhypothesen sind durch dreifarbige Achsen dargestellt.

der Posenhypothesen und zur Wiedererkennung eingesetzt. Nach dem Experiment wurden die aufgenommenen Evaluationsdaten (Laser, Kamera, Ground-Truth) verwendet, um den Fehler des Trackings ohne die erscheinungsbasierte Optimierung und Wiedererkennung zu bestimmen.

Die Steigerung der Leistungsfähigkeit des Trackers durch die erscheinungsbasierte Optimierung ist in Tabelle 5.1 dargestellt. Unter anderem werden zur Evaluation zwei Multiple Object Tracking (MOT) Leistungsmetriken [Bernardin, 2008] verwendet. Die MOT Precision (MOTP) bewertet den durchschnittlichen Positionsfehler, und die MOT Accuracy (MOTA) bewertet die Genauigkeit und Konsistenz des Trackers:

$$MOTA = 1 - \frac{\sum_k Miss_k + FP_k + MME_k}{\sum_k G_k} \quad (5.2)$$

Dabei ist $Miss_k$ die Anzahl an nicht detektierten Personen in Evaluationsschritt k und FP_k ist die Anzahl an fälschlich erzeugten Hypothesen. Zur Bewertung der Korrespondenz zwischen Ground-Truth Posen und getrackten Hypothesen wird die euklidische Distanz

mit einem Schwellwert von 0.6 m verwendet. Der Mismatch Error MME_k gibt an, wie oft die Personen-IDs falsch zugeordnet werden, und G_k bezeichnet die Anzahl der möglichen Personen-IDs zum Evaluationsschritt k .

Tabelle 5.1. Tracking Qualität ohne bzw. mit Optimierung & Wiedererkennung: Der mittlere Mismatch Error \overline{MME} bewertet die Wiedererkennung. Er gibt an, wie oft einer Person eine falsche ID zugeordnet wurde (ideal 0,0). In die Multi-Object-Tracking Accuracy (MOTA) fließen zusätzlich nicht detektierte oder fälschlich detektierte Personen sowie die tatsächliche Anzahl an Personen ein (ideal 1,0). Die MOT Precision (MOTP) bewertet den mittleren Fehler der Positionsschätzung, welcher noch einmal genauer bzgl. der einzelnen Dimensionen $\overline{\Delta_x}, \overline{\Delta_y}, \overline{\Delta_z}$ angegeben wird (ideal 0,0). Darüber hinaus ist noch der mittlere Orientierungsfehler $\overline{\Delta_\alpha}$ (ideal 0,0) gegeben.

	Tracking ohne erscheinungs- basierte Optimierung & Wiedererkennung	Tracking mit erscheinungs- basierter Optimierung & Wiedererkennung
# Evaluationen	355	355
\overline{MME}	$1,72 \cdot 10^{-2}$	0,0
MOTA	0,97	0,99
MOTP [cm]	18,7	17,3
$\overline{\Delta_x}$ [cm]	$10,5 \pm 0,71$	$10,2 \pm 0,68$
$\overline{\Delta_y}$ [cm]	$11,5 \pm 0,64$	$11,3 \pm 0,63$
$\overline{\Delta_z}$ [cm]	$6,35 \pm 0,02$	$3,02 \pm 0,03$
$\overline{\Delta_\alpha}$ [°]	$24,64 \pm 7,68$	$22,34 \pm 7,57$

Tabelle 5.1 zeigt, dass mittels des erscheinungsbasierten Modells eine Wiedererkennung von Personen möglich ist und die Posenschätzung leicht verbessert wird. Abgesehen von dem Fehler bzgl. der Schätzung der Kopfhöhe Δ_z , sind diese Verbesserungen jedoch minimal. Dies lässt sich zum einen damit erklären, dass die Parameter relativ stark von der Kalibrierung der intrinsischen und extrinsischen Kameraparameter abhängen. Außerdem hat der Körperbau der Personen relativ großen Einfluss auf die Matching-Funktion. Da sich der Körperbau und die optimierten Parameter θ überlagern, werden selbst geringe Unterschiede zwischen dem Körperbau der Probanden und dem Formmodell durch fehlerhafte Modellparameter θ ausgeglichen, um letztlich eine ähnliche Erscheinung zu erreichen. Für die Wiedererkennung ist jedoch nicht entscheidend, dass die korrekten Parameter gefunden werden, sondern dass die Parameter in Kombination mit dem Farbmodell die Erscheinung wiedergeben. Deshalb wird trotz der geringen Verbesserung der Posenschätzung eine gute Wiedererkennung erreicht, was sich im Mean Missmatch Error \overline{MME} und einer Steigerung der Multi-Object Tracking Accuracy (MOTA) zeigt.

Die Experimente zur erscheinungsbasierten Optimierung wurden auf einem Intel® Core™ i7 CPU mit 2.8 GHz und einer Nvidia® GeForce GTX 470 Grafikkarte durchgeführt.

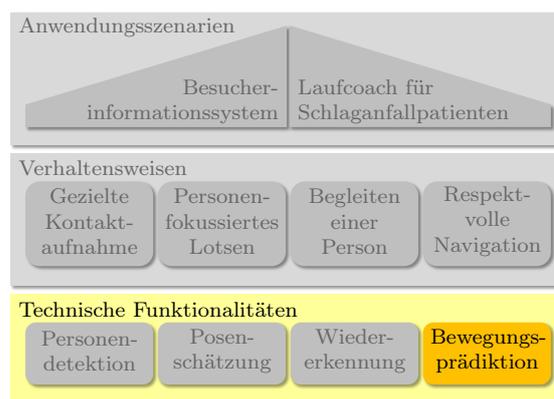
Die Berechnungszeit für die ercheinungsbasierte Optimierung einer Hypothese beträgt durchschnittlich 1.07 Sekunden. Dies ist der Grund, weshalb die Optimierung auf Einzelbildern basiert und unabhängig vom eigentlichen Tracking ist. Außerdem ist für die ercheinungsbasierte Optimierung eine Grafikkarte notwendig, welche die Implementierung eigener Shader erlaubt. Da eine solche GPU nicht auf dem Roboter vorhanden ist, wurden die notwendigen Daten per WLAN auf einen PC übertragen, dort berechnet und die Ergebnisse zurück an den Roboter übermittelt.

5.4. Kritische Bewertung der Ergebnisse

Die Experimente haben gezeigt, dass sich durch die ercheinungsbasierte Optimierung die Posenschätzung nur leicht verbessern lässt. Dies ist möglicherweise im vernachlässigten Einfluss der Oberkörperform begründet. In Bezug auf diese Arbeit ist außerdem entscheidend, dass sich das Verfahren nicht direkt auf den onboard-Rechnern des Robotersystems berechnen lässt, weil die Grafikkarten dafür nicht geeignet sind. Deshalb müssen die Daten per WLAN auf einen externen PC übertragen werden. Da dies jedoch nicht den Rahmenbedingungen entspricht (Abschnitt 2.4), welche sich erst später aus den Anwendungsszenarien ergaben, wurde kein weiterer Aufwand bzgl. der Verbesserung der Optimierung betrieben. Zur Wiedererkennung wird stattdessen ein auf [Eisenbach, 2012] aufbauendes Verfahren eingesetzt, bei welchem der Bereich der Person, von welcher die Merkmale extrahiert werden, vernachlässigt wird. Stattdessen wird mehr Aufwand bzgl. der Auswahl der Merkmale für die jeweilige Person betrieben.

6

Prädiktion des menschlichen Ausweichverhaltens



Die sozialverträgliche Navigation eines Roboters ist insbesondere dadurch gekennzeichnet, dass das Navigationsverhalten an die Personen in der Umgebung des Roboters angepasst ist, indem z. B. deren Personal Space [Hall, 1963] respektiert wird. In den letzten beiden Kapiteln wurden dementsprechend die in dieser Arbeit verwendeten Verfahren zur Detektion und zum Tracking von Personen beschrieben. Um die zukünftige Trajektorie des Roboters planen zu können, ist es darüber hinaus notwendig, auch die zukünftigen Aufenthaltswahrscheinlichkeiten der Personen zu prädictieren. In dieser Arbeit ist dies insbesondere für das reaktive Ausweichen vor Personen (Abschnitt 3.3.1) notwendig. Der Planungs- und Prädiktionshorizont sollte der Robotergeschwindigkeit angemessen sein, um eine gleichmäßige und vorausschauende Fahrweise gewährleisten zu können. So kann erreicht werden, dass die menschlichen Interaktionspartner nicht durch unvorhersehbare Bewegungsänderungen überrascht werden. Da die menschlichen Bewegungsmuster von der Erscheinung und dem Verhalten des Roboters abhängen [Walters, 2011, Hüttenrauch, 2006], dient das hier vorgestellte Verfahren zum on-line Lernen der Bewegungen der Personen in der Nähe des spezifischen Roboters, während der Roboter im Einsatz ist. Bei der Verhaltensweise *respektvolle Navigation* ist ein vorausschauendes Verhalten besonders wichtig. Dies zeigt sich auch bei der nachfolgenden Aufarbeitung ähnlicher Arbeiten zur Bewegungsprädiktion. Deshalb werden die Experimente auch am Beispiel dieser Verhaltensweise durchgeführt, obwohl auch noch andere Einsatzgebiete für die Prädiktion denkbar sind.

6.1. Prädiktion von Personenbewegungen

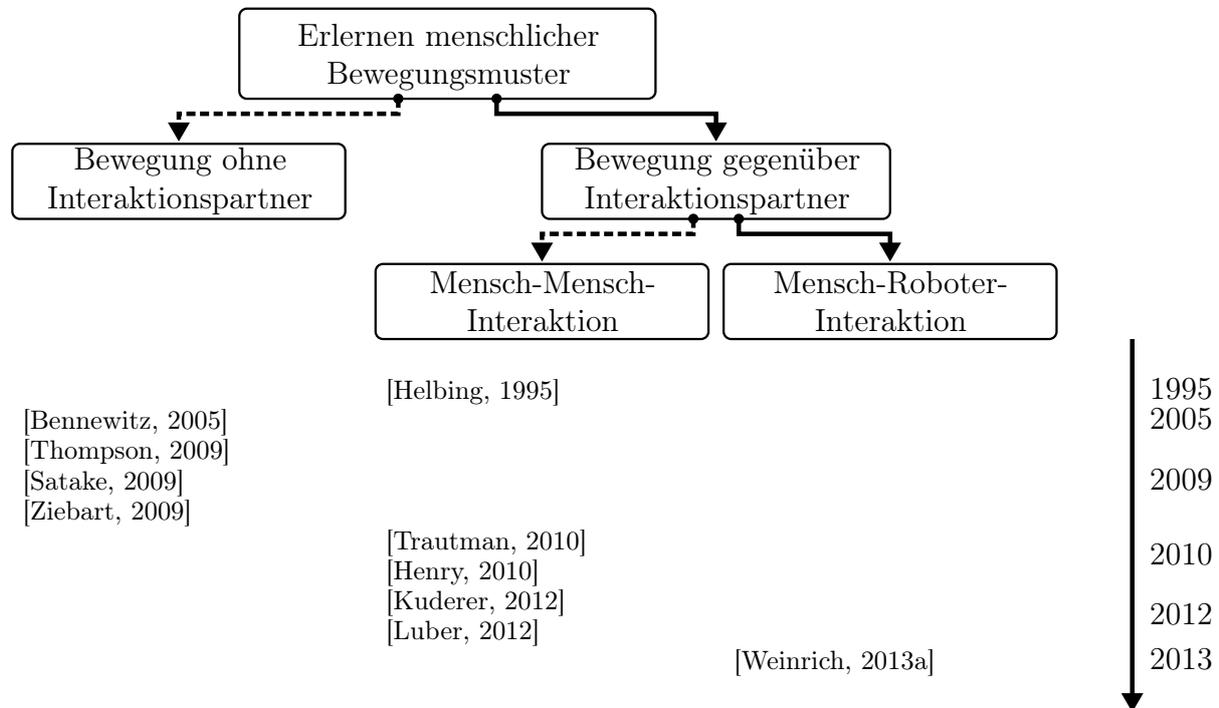


Abbildung 6.1. Arbeiten zur Prädiktion menschlicher Aufenthaltswahrscheinlichkeiten:
Darstellung von Verfahren zur Bewegungsprädiktion von Personen nach Jahren

Verschiedene Verfahren zum Ausweichen vor Personen [Bennewitz, 2005, Thompson, 2009] oder zum Heranfahren an Personen [Satake, 2009] mit mobilen Robotern berücksichtigen während der Pfadplanung typische menschliche Bewegungsmuster. In [Bennewitz, 2005] werden beispielsweise stationäre Laserscanner verwendet, um die Bewegungsmuster von Personen in einer Büroumgebung zu erfassen. Ein Hidden Markov Model wird auf diesen Daten trainiert und zur Prädiktion zukünftiger Positionen der getrackten Personen genutzt. Die prädizierten Trajektorien werden während der globalen Pfadplanung mittels dem A*-Algorithmus auf einer dreidimensionalen Kostenkarte über Raum und Zeit verwendet.

Um die Generalisierung gegenüber Veränderungen der Einsatzumgebung zu verbessern, werden in [Ziebart, 2009] die zielgerichteten Tajektorien der Fußgänger mittels Inverse Reinforcement Learning modelliert. Die gelernten Kostenfunktionen sollen so auch für verschiedene Einsatzumgebungen anwendbar sein. Die Prädiktion der Personenpositionen über die Zeit werden in eine zeitunabhängige Kostenkarte überführt, welche die Grundlage für den D*-Pfadplanungsalgorithmus darstellt. Allerdings wird für dieses Verfahren die

a-priori-Wahrscheinlichkeitsverteilung über globale Zielpositionen der Personen benötigt. Deshalb werden auch bei diesem Verfahren die Trajektorien der Personen mit statischen Laserscannern getrackt.

Bei den zuvor aufgeführten Verfahren fließen die prädizierten Trajektorien der Personen in die globale Pfadplanung des Roboters bis zum Ziel ein. Dementsprechend groß ist auch der zeitliche Planungshorizont für die Personenbewegungen, und es ist notwendig, dass die Personen auch in großer Entfernung zum Roboter durch externe Tracker wahrgenommen werden. Die Intention dabei ist, dass den Personen weiträumig ausgewichen wird und deren Bewegungen nach Möglichkeit nicht durch die Anwesenheit des Roboters beeinflusst werden. Wie in Abbildung 6.1 dargestellt, unterscheiden sich diese Verfahren, bei welchen die Anwesenheit des Roboters auf die Personenbewegung vernachlässigt wird, von den Verfahren zur kurzfristigen Bewegungsprädiktion in der Berücksichtigung von Interaktionspartnern.

In stark belebten Einsatzumgebungen ist es für den Roboter häufig unmöglich, sein Ziel zu erreichen, ohne dabei die Bewegung der anwesenden Personen zu beeinflussen. In [Trautman, 2010] wird dies als Freezing Robot Problem bezeichnet. Die vorgeschlagene Lösung ist eine direkte Interaktion zwischen dem Roboter und den Menschen, bei welcher sich der Roboter sozialverträglich verhält, während sich die Interaktionspartner gegenseitig ausweichen. Dementsprechend wird in [Kuderer, 2012] eine Maximum Entropie Lernmethode [Ziebart, 2009] verwendet, um das gemeinsame Verhalten mehrerer Menschen zu modellieren. Dieses Modell wird verwendet, um ausgehend von dem Zustand des Roboters und den Menschen in seiner Umgebung, deren Trajektorien zu prädizieren. Danach folgt der Roboter der Trajektorie, welche für ihn selbst prädiziert wurde und verhält sich somit, wie es ein Mensch in seiner Situation tun würde. Allerdings setzen beide Verfahren [Ziebart, 2009, Kuderer, 2012] voraus, dass auch die Zielpositionen der Menschen bekannt sind. Außerdem wird in [Trautman, 2010] und [Kuderer, 2012] selbst angemerkt, dass sich ein Mensch gegenüber anderen Menschen anders verhält als gegenüber Robotern. Somit sind Modelle, welche von der Mensch-Mensch-Interaktion gelernt wurden, wie das Social Forces Model [Helbing, 1995], Relative Motion Prototypes [Luber, 2012] oder das Inverse Reinforcement Learning [Henry, 2010], nicht direkt auf die Mensch-Roboter-Interaktion übertragbar. Deshalb wird in dieser Arbeit ein Verfahren angestrebt, bei welchem die Trainingsdaten aufgenommen werden, während der Roboter tatsächlich anwesend ist. Dies hat auch den Vorteil, dass der Roboter die Trainingsdaten selbst aufnehmen kann und die Einsatzumgebung nicht mit einem externen Trackingsystem ausgestattet werden muss. Da nur das lokale (Ausweich-)verhalten der Personen prädiziert wird, reduziert sich auch der zeitliche Prädiktionshorizont, und somit reduzieren sich auch die Unsicherheiten der Prädiktion. Für den Roboter ergeben sich somit zum einen mehr Freiräume zur

Navigation, weil berücksichtigt wird, dass ihm die Personen ausweichen. Zum anderen werden weniger Freiräume durch langfristige und somit räumlich unsichere Prädiktionen eingenommen.

6.2. On-line Lernen menschlicher Bewegungsmuster

Das nachfolgend vorgestellte Verfahren wird verwendet, um die Aufenthaltswahrscheinlichkeit der Personen, welche in der Nähe des Roboters beobachtet und getrackt werden (Abschnitt 4.3), über einen Zeitraum von wenigen Sekunden zu prädizieren. Die beobachteten Trajektorien der getrackten Personen werden außerdem verwendet, um die zukünftige Prädiktion on-line zu verbessern.

Mittels des on-line Lernens der Personenbewegung wird der Herausforderung begegnet, dass die Bewegung der Personen in der Nähe des Roboters bedeutend durch dessen Erscheinung und Verhalten beeinflusst wird [Walters, 2011]. Deshalb sollen die Bewegungsmuster der Personen gelernt werden, während der Roboter tatsächlich mit den Personen interagiert. Ebenso kann die Prädiktion angepasst werden, wenn sich das Verhalten der Person ändert, weil sich diese beispielsweise an den Roboter gewöhnen oder auf eine Veränderung des Roboterhaltens reagieren.

Um die Bewegung der zu einem bestimmten Zeitpunkt t getrackten Personenhypothesen $\mathbf{H}(t)$ zu prädizieren, wird für jede einzelne Hypothese $\mathbf{h}_i(t)$ ein Zustandsvektor $\mathbf{s}_i(t) = (pos(\mathbf{h}_i, t), vel(\mathbf{h}_i, t), \mathbf{e}(t))$ gebildet (Abb. 6.2). Dieser beschreibt die 2D Position $pos(\mathbf{h}_i, t)$ und den 2D Geschwindigkeitsvektor $vel(\mathbf{h}_i, t)$ der Hypothese $\mathbf{h}_i(t)$ relativ zur Roboterposition $pos(\mathbf{r}, t)$. Zusätzlich beschreibt $\mathbf{e}(t) \in \mathbb{R}^3$ die Hindernissituation um den Roboter. Die drei Skalare in $\mathbf{e}(t)$ kodieren den durchschnittlichen Abstand zwischen dem Roboter und den Hindernissen in seiner Umgebung innerhalb von drei 90°-Abschnitten (links, vorne, rechts) des 270°-Laserscans. Jeder Zustand $\mathbf{s}_i(t)$ des 7D-Zustandsraumes \mathbf{S} wird mittels einer on-line Clusterfunktion $cluster : \mathbb{R}^7 \rightarrow \{1, \dots, C\}$ einem von C Clustern zugeordnet. Wie Abbildung 6.2 zeigt, wird für jedes Cluster c_j eine Belief Verteilung der Aufenthaltswahrscheinlichkeit der Person über Raum und Zeit gespeichert. Die Aufenthaltswahrscheinlichkeiten werden durch eine Menge von Grid Maps $\mathbf{M}_1^{c_j}, \dots, \mathbf{M}_T^{c_j}$ repräsentiert. Die Zahlenwerte der Zellen der Karten $\mathbf{M}_\tau^{c_j}$ speichern den Belief $Bel(c_j, \Delta t_\tau, x, y)$ für Personenposition (x, y) relativ zur aktuellen Roboterposition $pos(\mathbf{r}, t)$ während des zukünftigen Zeitintervalls Δt_τ .

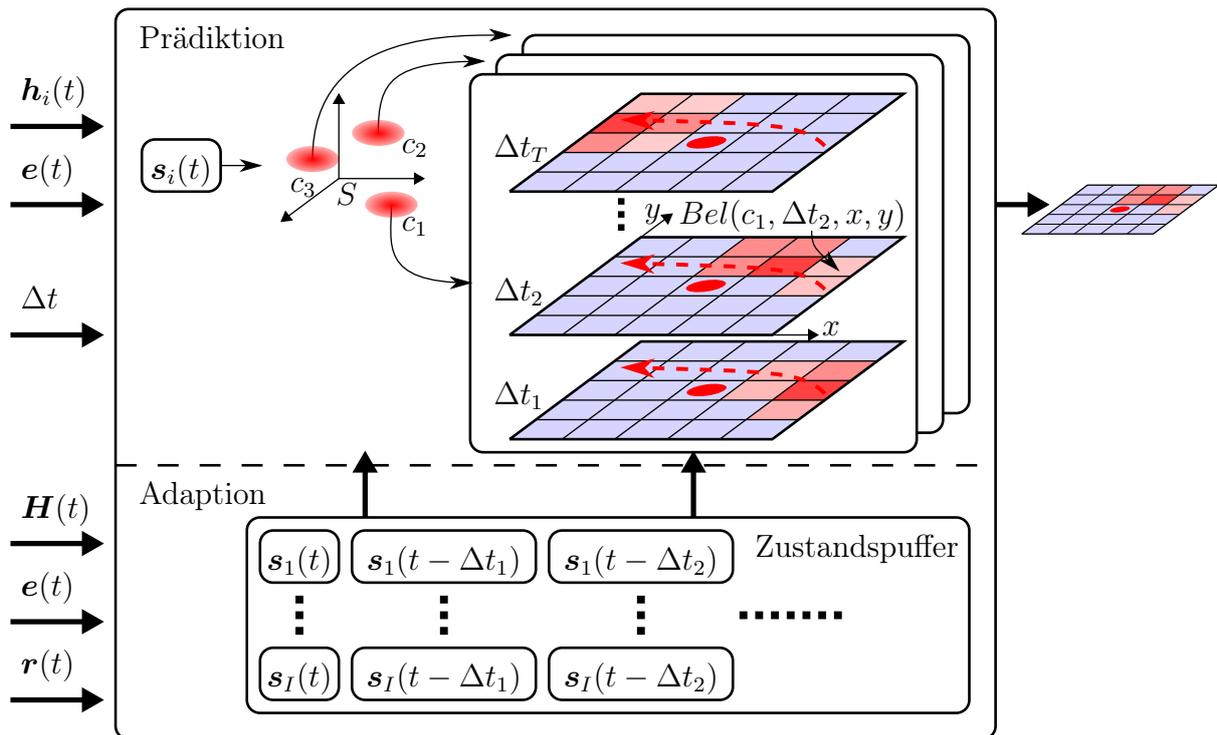


Abbildung 6.2. Belief über zukünftigen Aufenthalt der Personen: Jedes Cluster c_j innerhalb des 7D-Zustandsraumes S (hier nur 3D dargestellt) verweist auf eine diskrete, multimodale Belief Distribution über die zukünftigen Personenpositionen für mehrere Prädiktionsintervalle Δt . Jede Gridzelle repräsentiert eine relative Position zur initialen Roboterposition $r(t)$ zum Prädiktionszeitpunkt t , welche mit einer roten Ellipse markiert ist. Eine Gridzelle mit geringem Belief ist hellblau, und ein hoher Belief ist rot dargestellt. Die gestrichelten Pfeile zeigen die Bewegung des maximalen Beliefs über die Zeit.

Adaption der Aufenthaltswahrscheinlichkeiten Während der Roboter im Einsatz ist und mit Personen interagiert, wird jede getrackte Hypothese $h_i(t)$ in den Zustandsraum S überführt. Jeder Zustand $s_i(t)$ wird genutzt, um die Position des entsprechenden Clusters $c_j = cluster(s_i(t))$ entsprechend einer Implementierung des k-Means-Algorithmus [MacQueen, 1967] für sequentielle Daten zu adaptieren, wobei die normalisierte euklidische Distanz¹ als Distanzmaß verwendet wird. Außerdem wird die aktuelle Personenposition $pos(h_i, t)$ verwendet, um die Belief Distribution Maps der Cluster $cluster(s_i(t - \Delta t_1)), cluster(s_i(t - \Delta t_2)), \dots, cluster(s_i(t - \Delta t_T))$, welchen die vergangenen Zustände des Tracklet h_i zugeordnet wurden, zu adaptieren. Für jedes zuvor relevante Cluster $cluster(s_i(t - \Delta t_\tau))$ zum Zeitpunkt $t - \Delta t_\tau$ wird die aktuelle Personenposition $pos(h_i, t)$ relativ zu der damaligen Roboterposition $pos(r, t - \Delta t_\tau)$ transformiert. Die Belief Distri-

¹Die normalisierte euklidische Distanz ist ein Spezialfall der Mahalanobisdistanz, bei welcher die Kovarianzmatrix eine Diagonalmatrix darstellt

bution des Clusters wird adaptiert, indem der Belief der Gridzelle der Karte $M_{\tau}^{c(s_i(t-\Delta t_{\tau}))}$, welche der relativen Personenposition (x, y) entspricht, erhöht wird. Die aktuelle Personenposition wird somit zur Prädiktion für einen vergangenen Zustand genutzt.

Verwendung der Aufenthaltswahrscheinlichkeiten Grundsätzlich können diese Aufenthaltswahrscheinlichkeiten während der Bewegungsplanung des Roboters zur Realisierung verschiedenster Zielstellungen verwendet werden. In dieser Arbeit werden sie im Kontext der respektvollen Navigation zum reaktiven Ausweichen vor Personen (Abschnitt 3.3.1) eingesetzt, weil dabei die Beeinflussung der Personentrajektorie durch den Roboter besonders hoch ist. Wie bereits in Abschnitt 3.3.1 beschrieben, werden die prädizierten Aufenthaltswahrscheinlichkeiten der Personen verwendet, um zu bestimmen, wie sehr der Personal Space der Personen zukünftig durch die Anwesenheit des Roboters an einer bestimmten Position verletzt würde. Für diese Verletzung werden Kosten berechnet, welche ausgehend von der zukünftigen Personenposition mit euklidischem Abstand entsprechend einer 2D Gauß-Kurve abnehmen. In Kapitel 7 erfolgt die Evaluation der respektvollen Navigation, ohne die Prädiktion direkt zu untersuchen. Deshalb erfolgt eine genauere Untersuchung der Prädiktionsgüte bereits im nächsten Abschnitt.

6.3. Experimente

Zur Bewertung des vorgestellten Verfahrens zur Bewegungsprädiktion wurde zuerst ein Experiment durchgeführt, bei welchem nur die Personenbewegung gelernt wurde, ohne dass dadurch das Roboterverhalten angepasst wurde. In weiteren Experimenten wurde untersucht, wie sich mittels des on-line Lernens der Personenbewegung ein Mensch und ein Roboter beim gegenseitige Ausweichen aufeinander abstimmen können. Die Experimente wurden in den Fluren des Universitätsgebäudes (Abb. 6.3), welches das Einsatzgebiet des Anwendungsszenarios Besucherinformationssystem (Abschnitt 2.1) darstellt, durchgeführt.

Evaluierung der Prädiktionsgüte Das erste Experiment dient der Untersuchung, ob Belief Distribution Maps besser zur Prädiktion menschlicher Bewegung in der Nähe eines Roboters geeignet sind als eine lineare Prädiktion der Personenbewegung. Die Prädiktionen beider Verfahren werden dazu für verschiedene Prädiktionshorizonte mit den tatsächlich beobachteten Positionen verglichen. Als Vergleichsmaß dient die Kreuzkorrelation zwischen der jeweiligen Prädiktion und der tatsächlichen Beobachtung. Damit der Korrelationswert mit wachsendem Fehler der Prädiktion kontinuierlich sinkt, werden

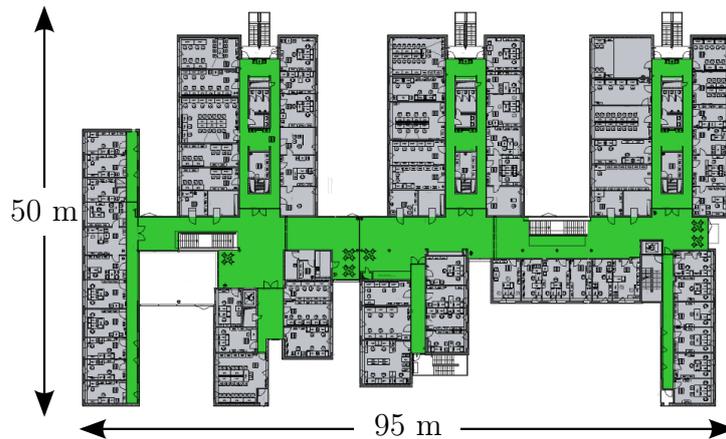


Abbildung 6.3. Versuchsumgebung: Die Experimente wurden in einer Etage des Bürogebäudes, welches die Einsatzumgebung des Besucherinformationssystems darstellt, durchgeführt. Der Betriebsbereich des Roboters ist grün hervorgehoben. (Übernommen aus [Weinrich, 2013a])

sowohl die prädierten Positionen als auch die beobachteten Positionen mit einem Glättungskern gefaltet. Die Prädiktion wird im Rahmen dieser Arbeit vorrangig zur Wahrung des Personal Space (Abschnitt 3.3.1) eingesetzt. Deshalb wird hier ein Gauß-Kern zur Glättung eingesetzt, wie er auch bei der Kostenfunktion des Personal Space verwendet wird (Abschnitt 3.3.1). Somit bewertet die hier berechnete Korrelation die Ähnlichkeit zwischen der auf tatsächlichen Beobachtungen basierten Kostenfunktion und der prädierten Kostenfunktion.

Genauere Erläuterungen zur Berechnung der Korrelation cc^{bel} des lernenden Verfahrens bzw. der Korrelation cc^{lin} der linear prädierten Kostenfunktion sind in Abschnitt F.2 zu finden. Da alle drei Kostenfunktionen (lineare Prädiktion: Gleichung C.3, lernende Prädiktion: F.1, Ground-Truth: F.5) abhängig von der Standardabweichung σ des Gauß-Kerns sind, wurden zwei verschiedene Standardabweichungen untersucht.



Abbildung 6.4. Roboter- und Personentrajektorien: Visualisierung der Trajektorie des Roboters (blau) und der durch den Roboter getrackten Trajektorien der Personen (rot) innerhalb des Hauptflures der Einsatzumgebung. Die getrackten Personenpositionen werden zum on-line Training und zur Evaluation des lernenden Prädiktionsverfahrens genutzt.

Während des Experiments ist der Roboter auf einem Hauptgang patrouilliert und hat dabei die Menschen \mathbf{h}_i in seiner Umgebung getrackt. Die entsprechenden Zustände $\mathbf{s}_i(t)$ wurden genutzt, um die Kostenfunktion mit dem lernenden Verfahren $\text{cost}_{\text{PS}}^{\text{bel,d}}(\mathbf{r}', t', \mathbf{h}_i)$ und linear $\text{cost}_{\text{PS}}^{\text{lin,d}}(\mathbf{r}', t', \mathbf{h}_i)$ zu präzisieren. Außerdem wurden die später beobachteten Personenhypothesen $\mathbf{h}_i(t + \Delta t)$ verwendet, um die Ground-Truth Kostenfunktion $\text{cost}_{\text{PS}}^{\text{gt,d}}(\mathbf{r}', t', \mathbf{h}_i)$ zu berechnen. Es sei angemerkt, dass während der Experimente die unveränderliche Kostenfunktion $\text{cost}_{\text{PS}}^{\text{lin,d}}(\mathbf{r}', t', \mathbf{h}_i)$ zur respektvollen Navigation verwendet wurde, damit sich das Verhalten des Roboters nicht über die Zeit verändert. Eine Veränderung des Roboterverhaltens könnte schließlich Auswirkungen auf das Verhalten der Personen haben. Die Trajektorien des Roboters (blau) und der während des Experiments getrackten Personen (rot) sind in Abbildung 6.4 dargestellt.

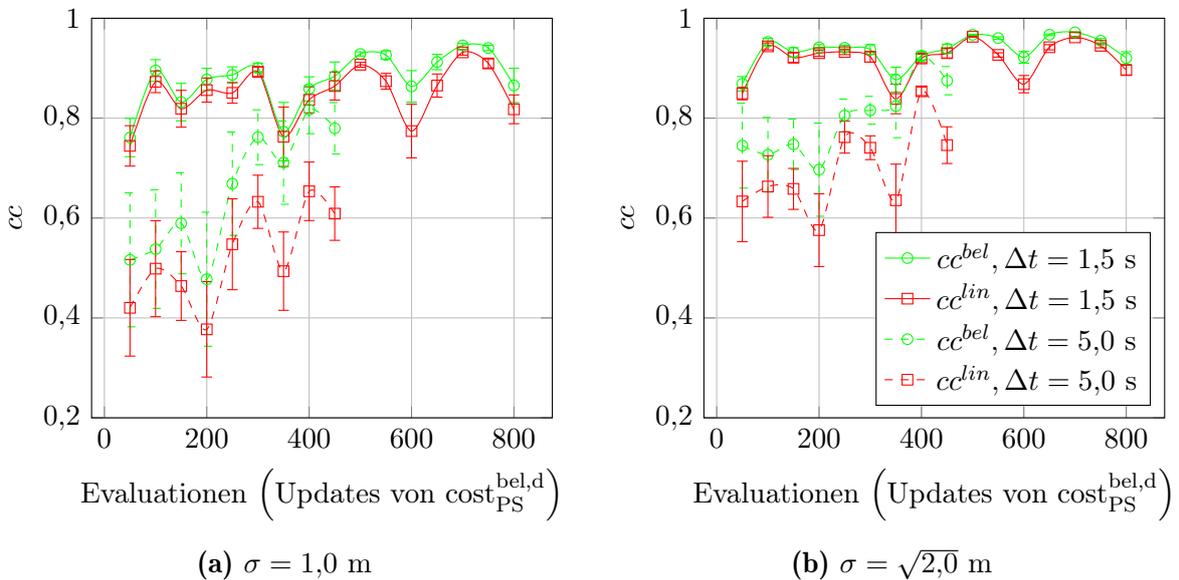


Abbildung 6.5. Evaluation der präzisierten Kostenfunktionen I: Zur Evaluation wird die Korrelation cc^{bel} zwischen der gelernten Kostenfunktion $\text{cost}_{\text{PS}}^{\text{bel,d}}$ und der Ground-Truth Kostenfunktion $\text{cost}_{\text{PS}}^{\text{gt,d}}$ dargestellt. Als Vergleichswert dient die Korrelation cc^{lin} zwischen der linear präzisierten Kostenfunktion $\text{cost}_{\text{PS}}^{\text{lin,d}}$ und der Ground-Truth Kostenfunktion $\text{cost}_{\text{PS}}^{\text{gt,d}}$. Für die Gauß-Funktion, welche zur Modellierung der Kostenfunktionen aus den Prädiktionen bzw. Beobachtungen genutzt wird, wurde eine Standardabweichung (a) $\sigma = 1,0\text{ m}$ und (b) $\sigma = \sqrt{2,0}\text{ m}$ verwendet. Um zu zeigen, wie sich die Adaption der gelernten Prädiktion zur Laufzeit auswirkt, ist die Korrelation über die Anzahl an Adaptionen dargestellt. Die dargestellten Funktionsgraphen sind Tiefpass gefiltert, indem über 50 aufeinander folgende Korrelationswerte gemittelt wurde. Für den 5 s Prädiktionshorizont konnten während des Experiments nicht so viele Evaluationen durchgeführt werden wie für den 1,5 s Horizont, da der on-board Tracker die Personen nicht immer über 5 s tracken konnte.

Abbildung 6.5 stellt die Korrelation cc^{bel} (grün) der Kostenfunktion $\text{cost}_{\text{PS}}^{\text{bel,d}}$, welche auf der vorgestellten Prädiktion basiert, und die Korrelation cc^{lin} (rot) der linear präzisierten

Kostenfunktion $\text{cost}_{\text{PS}}^{\text{lin,d}}$ dar. Die Kreuzkorrelation bei einer gleichverteilten Belief Distribution über die gesamte Grid Map wäre ca. 0,055 und aufgrund einer Normierung ist die maximal erreichbare Korrelation 1,0. Das Diagramm zeigt für die Prädiktionshorizonte $\Delta t = 1,5$ s und $\Delta t = 5$ s jeweils die Kreuzkorrelation cc , nachdem diese durch die Ground-Truth Beobachtungen validiert wurden. Es wird deutlich, dass die Korrelation $\overline{cc}^{\text{bel}}$ des vorgestellten Verfahrens mit den tatsächlichen Beobachtungen durchweg höher ist als die Korrelation $\overline{cc}^{\text{lin}}$ der linearen Methode. Entscheidend ist, dass der Unterschied der Prädiktionsgüte mit der Anzahl der Beobachtungen steigt, weil $\text{cost}_{\text{PS}}^{\text{bel,d}}$ initial eine lineare Gehbewegung der Personen annimmt und die Prädiktionsgüte durch die Beobachtungen verbessert wird, wohingegen $\text{cost}_{\text{PS}}^{\text{lin,d}}$ nicht adaptiert wird. Die wachsende Differenz zwischen den Prädiktionsgüten wird insbesondere bei dem längeren Prädiktionshorizont von 5 Sekunden deutlich.

Die Auswertung für verschiedene Standardabweichungen $\sigma = 1,0$ m bzw. $\sigma = \sqrt{2,0}$ m verdeutlicht, dass eine größere Standardabweichung generell zu einer höheren Kreuzkorrelation führt, jedoch dadurch keine qualitativen Unterschiede hervorgerufen werden.

Abbildung 6.6 zeigt eine Auswahl der Belief Grid Maps eines während des Experiments trainierten Clusters. Insgesamt wurden 600 Cluster verwendet und für jedes wurden 20 Belief Distribution Maps gelernt, wobei jede Karte den Belief über ein Zeitintervall von 250 ms speichert. Daraus ergibt sich ein Prädiktionshorizont von 5 Sekunden. Jede Gridkarte umfasst $15 \text{ m} \times 15 \text{ m} = 225 \text{ m}^2$ der Umgebung des Roboters unter Nutzung von $40 \times 40 = 1600$ Gridzellen. Da der Belief im 4 Byte Gleitkommaformat gespeichert wird, sind insgesamt ca. 74 MB zur Speicherung aller Karten aller Cluster notwendig.

Es sei angemerkt, dass der Personentracker, welcher die Ground-Truth Hypothesen liefert, auf einem Kalman-Filter mit linearem Bewegungsmodell (siehe Kapitel 4.3) basiert. Dies ist zum Vorteil der linearen Prädiktion, wenn Personen nicht ausreichend oft detektiert werden. Außerdem haben sich während des Experiments viele Personen in großem Abstand zum Roboter bewegt, so dass der Roboter kaum Einfluss auf die Trajektorie der Personen hatte. Die Personen haben sich somit meist linear fortbewegt. Die lineare Prädiktion ist insbesondere relevant, wenn sich eine Person und der Roboter sehr nah sind, da in diesen Situationen beide Interaktionspartner zum Ausweichen von ihrem geradlinigen Pfad abweichen müssen. Dies wird im nächsten Experiment untersucht.

Steigerung der sozialen Verträglichkeit In diesem Experiment wird der Einfluss der Bewegungsprädiktion auf das Roboterverhalten untersucht. Es wird davon ausgegangen, dass Menschen eine ökonomische Bewegung bevorzugen, und so basiert die Bewertung des Roboterhaltens auf dem Umweg, welchen eine Person zum Ausweichen des Roboters

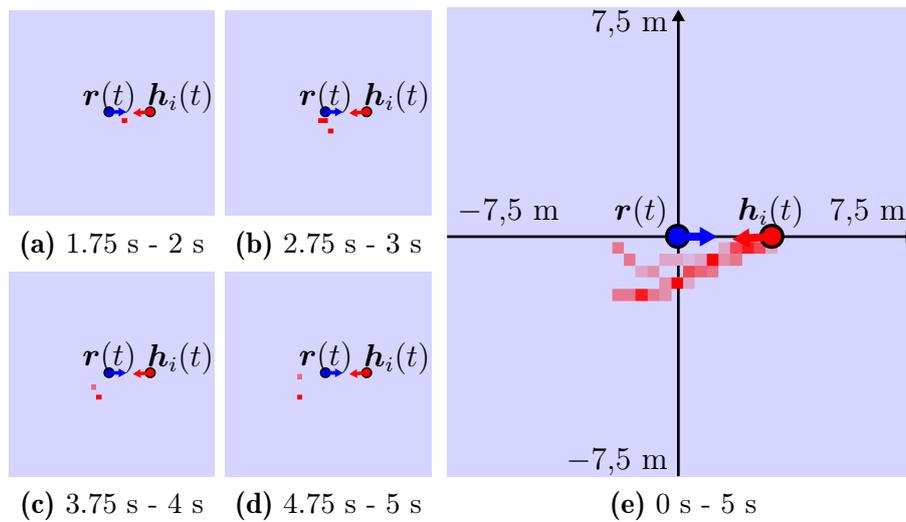


Abbildung 6.6. Belief Distribution Maps: Die Belief Distribution Maps eines Clusters im Zustandsraum speichern für die entsprechenden Ausgangszustände die zukünftigen Aufenthaltswahrscheinlichkeiten der Person über die Zeit (vgl. schematische Darstellung in Abb. 6.2). (a)-(d) zeigen 4 der 20 Belief Distribution Maps eines bestimmten Clusters nach der Durchführung des ersten Experiments. Die initiale Position und Bewegungsrichtung der Person ist als roter Pfeil und die initiale Roboterpose ist als blauer Pfeil dargestellt. Jede Belief Distribution Map deckt ein Zeitintervall von 250 ms und einen Raum von $15\text{ m} \times 15\text{ m}$ mit 40×40 Gridzellen ab. Die Stärke des Belief ist durch die Intensität der roten Farbe dargestellt. Die 4 Maps zeigen, dass die Person dem Roboter in der entsprechenden Situation nach links ausweicht. In (b)-(d) ist zu erkennen, dass die Person dem Roboter mit hoher Wahrscheinlichkeit entweder in einem weiteren oder in einem engeren Bogen ausweichen wird. Insgesamt werden pro Cluster 20 Belief Distribution Maps gespeichert, um die zukünftige Aufenthaltswahrscheinlichkeit über 5 Sekunden präzisieren zu können. (e) zeigt den Durchschnitt aller 20 Belief Distribution Maps dieses Clusters. Der engere und weitere Bogen der Person ist deutlich zu erkennen.

gehen muss. Dazu wurde ein Versuchsaufbau gewählt, bei welchem eine Person und der Roboter wiederholt ihre Positionen tauschen sollten (Abb. 6.7), wodurch sie gezwungen sind, sich gegenseitig auszuweichen. Die beiden Positionen A und B sind so gewählt, dass zum Ausweichen in der einen Richtung Freiraum zur Verfügung steht und der Platz zum Ausweichen in der anderen Richtung durch eine Wand begrenzt ist.

Während vier Telexperimenten sollten durch einen Probanden zwei verschiedene Ausweichstrategien verfolgt werden, und für beide Strategien wurde von dem Roboter einmal die linear prädizierte Kostenfunktion $\text{cost}_{\text{PS}}^{\text{lin,d}}$ und einmal die lernende Kostenfunktion $\text{cost}_{\text{PS}}^{\text{bel,d}}$ zum Ausweichen der Person genutzt. Bei der einen Ausweichstrategie sollte die Person dem Roboter auf der Seite ausweichen, auf welcher der Abstand zum nächsten Hindernis größer ist (Abb. 6.7a und 6.7b). Dieses *freiraumgerichtete Ausweichen* entspricht einem vorsichtigen Verhalten, welches geringe Fähigkeiten des Roboters erwartet.

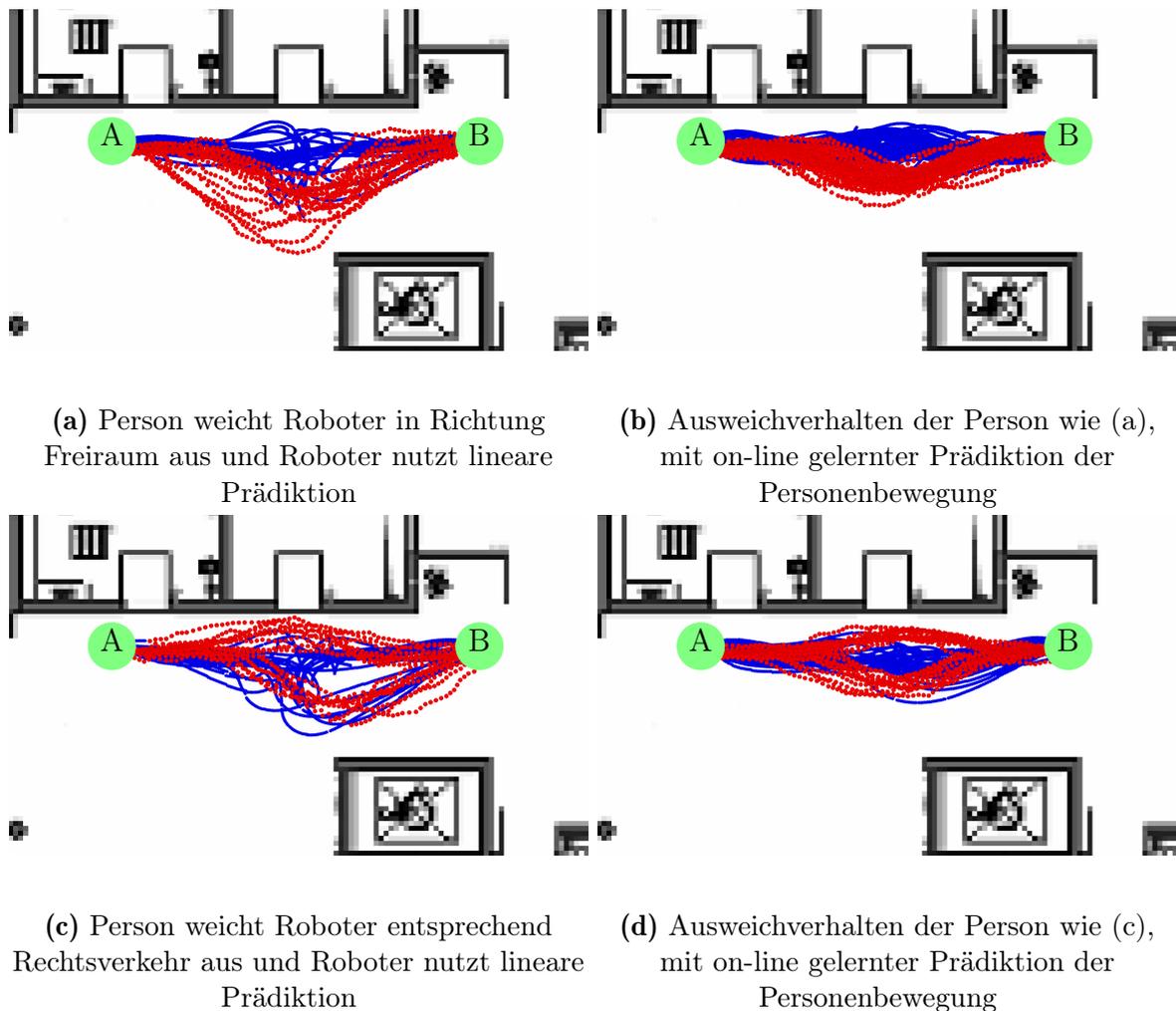


Abbildung 6.7. Gegenseitiges Ausweichen durch Person und Roboter: Pfad des Roboters (blau) und einer Person (rot), während beide wiederholt die Positionen A und B tauschen und sich dabei gegenseitig ausweichen müssen. Weicht die Person dem Roboter in Richtung des Freiraumes aus, führt das Verhalten des Roboters bei linearer Prädiktion der Personenbewegung (a) zu einem weitaus größeren Umweg für die Person, als wenn die gelernte Prädiktion angewendet wird (b). Das gleiche trifft zu, wenn die Person dem Roboter entsprechend dem Rechtsverkehr ausweicht (c),(d). Bei linearer Prädiktion (a),(c) weicht der Roboter der Person häufig in die Richtung aus, in welche auch die Person ausweichen soll. Dann verstärkte die Person meist die Ausweichbewegung und die Ausweichrichtung wird für den Roboter auch bei linearer Prädiktion eindeutig. Um nun der Person in der angemessenen Richtung auszuweichen, fährt der Roboter häufig einen engen Bogen. Bei gelernter Prädiktion (b),(d) weicht der Roboter direkt in die richtige Richtung aus und für die Person ergibt sich ein geringer Umweg.

Bei der anderen Ausweichstrategie (Abb. 6.7c und 6.7d) sollte die Person dem Roboter dem *Rechtsverkehr* entsprechend ausweichen. Das bedeutet, dass die Person dem Roboter auf dem Weg von Startposition B zu Zielposition A Richtung Wand ausweicht.

Während der beiden Telexperimente mit lernender Prädiktion sind die Belief Distribution Maps initial untrainiert, und der Roboter geht von einer linearen Bewegung aus. Während des Experiments wird die menschliche Ausweichstrategie gelernt, und da der Roboter die Prädiktion zum reaktiven Ausweichen der Person nutzt (Abschnitt 3.3.1), verändert sich auch das Verhalten des Roboters. Dies wiederum hat Einfluss auf die Personenbewegung, und so wird der Umweg der Person zum Ausweichen des Roboters als Bewertungskriterium für die Prädiktionsverfahren und das damit verbundene Roboterverhalten genutzt.

Damit die Erfassung des Umwegs nicht durch Fehler des on-board Personentrackers oder durch fehlerhafte Selbstlokalisierung des Roboters beeinflusst wird, erfolgt die Erfassung des Umweges durch ein externes Tracking-System, aufbauend auf [Schenk, 2011]. Während sich die Person und der Roboter ausweichen, wird der Umweg der Person relativ zur direkten Verbindung zwischen Start- und Zielgebiet berechnet. Die zurückgelegten Wege von Person und Roboter sind in Abbildung 6.7 dargestellt. Es ist ersichtlich, dass bei beiden Ausweichstrategien der Umweg, welchen die Person zurücklegen muss, geringer ist, wenn der Roboter die lernende Prädiktion statt der linearen Prädiktion anwendet.

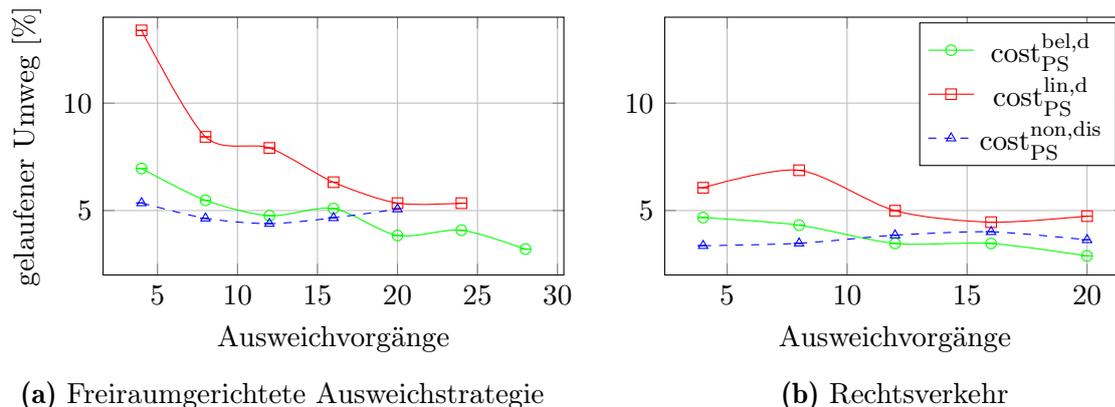


Abbildung 6.8. Umweg des Probanden beim Ausweichen des Roboters: Darstellung des Umweges, welchen die Testperson während der Experimente beim Ausweichen des Roboters entsprechend der (a) freiraumgerichteten Ausweichstrategie bzw. des (b) Rechtsverkehrs zurückgelegt hat. Für beide Ausweichstrategien ist der Umweg der Person bei Verwendung der gelernten Prädiktion $\text{cost}_{\text{PS}}^{\text{bel,d}}$, der linearen Prädiktion $\text{cost}_{\text{PS}}^{\text{lin,d}}$ und ohne Prädiktion $\text{cost}_{\text{PS}}^{\text{non,dis}}$ über die Ausweichvorgänge dargestellt. Die Kurve jedes Telexperiments ist tiefpassgefiltert, indem der Umweg von vier nacheinander folgenden Ausweichvorgängen des Probanden gemittelt wurde.

In Abbildung 6.8 ist die Veränderung des Umwegs der Person über die Zeit gezeigt. Bei beiden Ausweichstrategien sinkt der Umweg der Person während die Prädiktionsfunktion $\text{cost}_{\text{PS}}^{\text{bel,d}}$ adaptiert wird. Und so ergibt sich bei gelernter Prädiktion $\text{cost}_{\text{PS}}^{\text{bel,d}}$ letztlich ein geringerer Umweg für die Person als bei linearer Prädiktion $\text{cost}_{\text{PS}}^{\text{lin,d}}$.

Der gewählte Versuchsaufbau stellt eine besondere Herausforderung an die Prädiktion dar, weil die Person direkt auf den Roboter zu läuft und so nur schwer zu erkennen ist auf welcher Seite die Person dem Roboter ausweichen wird. In dieser nicht eindeutigen Situation tendiert der Roboter bei linearer Bewegungsprädiktion dazu, der Person auf der von der Wand abgewandten Seite auszuweichen. Der Grund dafür ist das *Distance Objective* (Kapitel 3), welches Bewegungen schlechter bewertet, die zu einer Annäherung an Hindernisse, wie die Wand, führen. Dies führt dazu, dass der Roboter beim *freiraumgerichteten Ausweichen* in die gleiche Richtung fährt wie die Person, und auch beim *Rechtsverkehr* weicht der Roboter in dieselbe Richtung aus wie die Person, wenn sich diese von A nach B bewegt. Dieser Umstand legt die Vermutung nah, dass es bei den durchgeführten Experimenten besser ist, die Personenbewegung nicht zu prädizieren, als sie falsch zu prädizieren. Deshalb wurden zusätzlich Experimente zu beiden Ausweichstrategien ohne Prädiktion der Personenbewegung $\text{cost}_{\text{PS}}^{\text{non,dis}}$ durchgeführt. Während der Navigation wurde entsprechend zu jedem Zeitschritt davon ausgegangen, dass die Person da bleibt, wo sie gerade getrackt wird. Abb. 6.8 zeigt, dass das daraus resultierende Verhalten zu Beginn der Experimente tatsächlich zu dem geringsten Umweg führt. Während den Experimenten reduziert sich auch bei der unveränderlichen linearen Prädiktion der Umweg der Person, weil die Person gelernt hat, dem Roboter frühzeitig auszuweichen, damit dieser nicht in die gleiche Richtung wie die Person ausweicht. Dennoch bleibt bei linearer Prädiktion der Umweg der Person höher, als wenn die Personenbewegung nicht

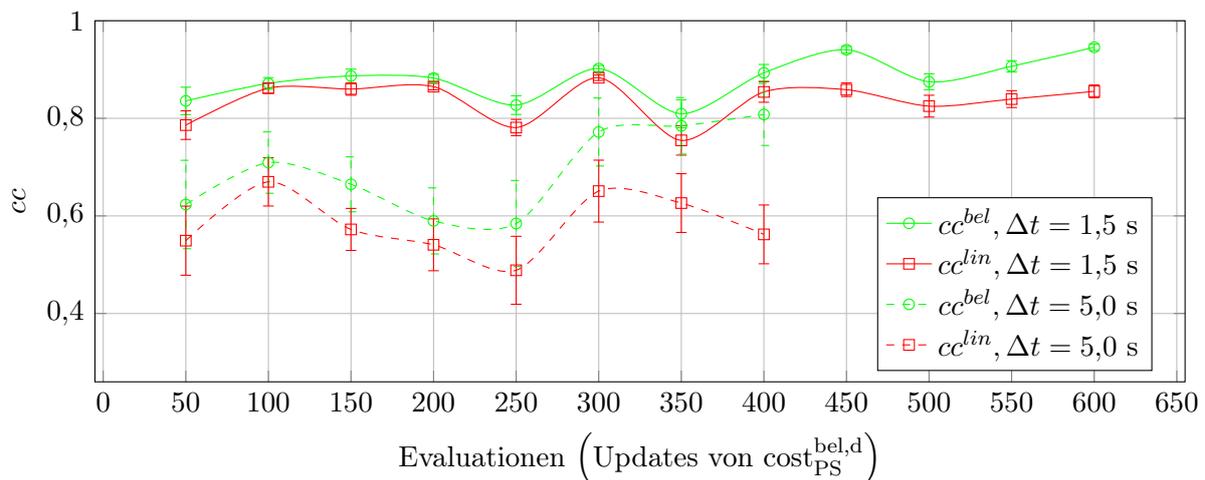


Abbildung 6.9. Evaluation der prädizierten Kostenfunktionen II: Korrelation cc^{lin} der linearen Kostenfunktion und Korrelation cc^{bel} der gelernten Kostenfunktion mit der Kostenfunktion, welche auf den tatsächlichen Beobachtungen basiert. Insbesondere bei dem fünf Sekunden Prädiktionshorizont wird die Verbesserung der gelernten Prädiktion über die Zeit deutlich. Die dargestellten Funktionsgraphen sind Tiefpass gefiltert, indem über 50 aufeinander folgende Korrelationswerte gemittelt wurde.

prädiziert wird. Im Gegensatz dazu schafft es das in dieser Arbeit vorgestellte lernende Verfahren nach ca. 20 Ausweichvorgängen die Ausweichrichtung der Person so gut vorherzusagen, dass der niedrigste Umweg für die Person erreicht wird. Der Umweg ist für die Person beim *freiraumgerichteten Ausweichen* höher als beim *Rechtsverkehr*, da der Roboter der Person Richtung Wand weniger stark ausweichen kann und deshalb die Person den größeren Umweg zurücklegen muss.

Nachdem der Umweg des Probanden ausgewertet wurde, ist in Abb. 6.9 die Prädiktionsgüte während des Telexperiments zum *freiraumgerichteten Ausweichen* mit lernender Prädiktion dargestellt. Während des Experiments wurde zwar die lernende Prädiktion zur Navigation verwendet, es wird jedoch die Prädiktionsgüte beider Prädiktionverfahren bewertet. Es wird deutlich, dass die Prädiktionsgüte von $\text{cost}_{\text{PS}}^{\text{bel,d}}$ durch die on-line Adaption im Vergleich zu $\text{cost}_{\text{PS}}^{\text{lin,d}}$ während des Experiments kontinuierlich besser wird.

6.4. Zusammenfassung und Ausblick

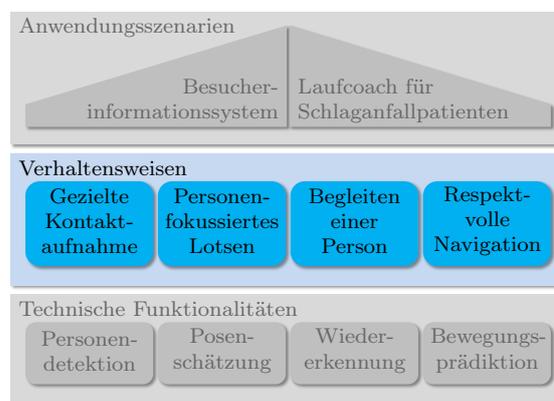
Wenn sich eine Person und ein Roboter so nahe sind, dass sie sich gegenseitig wahrnehmen können, ist häufig auch die Bewegung der Person durch den Roboter beeinflusst und weicht von einer geradlinigen Bewegung ab. So konnte in einem ersten Experiment gezeigt werden, dass eine gelernte Prädiktion der Personenbewegung besser mit den tatsächlichen Beobachtungen korreliert als eine lineare Prädiktion. Wird die on-line gelernte Prädiktion statt der linearen Prädiktion zum Ausweichen vor Personen verwendet, so reduziert sich für die Person der Umweg zum Ausweichen des Roboters. In den dazu durchgeführten Experimenten zeigte sich, dass implizit gelernt wurde, unter Berücksichtigung der Umgebung zu entscheiden, auf welcher Seite der Person der Roboter ausweichen sollte. Um dies zu zeigen, wurden in diesem Kapitel spezielle Experimente mit nur einem Probanden durchgeführt. Im folgenden Kapitel 7 werden umfangreichere Experimente mit mehreren Testpersonen zur respektvollen Navigation mit lernender Prädiktion beschrieben.

Befinden sich Personengruppen in der Nähe des Roboters, kann die Prädiktion der Bewegung für jede einzelne Person angewendet und auch bei der respektvollen Navigation berücksichtigt werden. Allerdings wird dabei die Interaktion der Personen untereinander nicht explizit berücksichtigt.

Bezüglich der Clusterung der Ausgangszustände, auf welcher die Prädiktion basiert, besteht noch Verbesserungspotential. So wäre es sicher vorteilhaft, wenn die Ausdehnung jedes einzelnen Cluster in Abhängigkeit von den Belief Distribution Maps angepasst würde.

7

Evaluation der Navigationsverhalten



In Kapitel 3 wurde die Umsetzung verschiedener Navigationsverhalten des mobilen Roboters beschrieben. In den Kapiteln 4 - 6 erfolgte die Beschreibung technischer Funktionalitäten, welche die Grundlage für diese Verhaltensweisen darstellen und somit maßgeblich Einfluss auf das tatsächlich realisierte Verhalten haben. In diesem Kapitel werden nun verschiedene Aspekte des realisierten Gesamtverhaltens evaluiert. Besonderer Fokus liegt dabei dem Thema entsprechend auf der respektvollen Navigation.

Die experimentellen Untersuchungen wurden im Fakultätsgebäude, dem Einsatzgebiet des Gebäudelotsen, und in der Rehaklinik, dem Einsatzgebiet des Rehabilitationsroboters, durchgeführt. Im Fakultätsgebäude steht ein externes Trackingsystem zur Verfügung. Deshalb können dort bestimmte Interaktionssituationen wiederholt durchgeführt werden, wobei die Bewegung des Roboters und der Probanden erfasst und anhand der später erläuterten Evaluationsmetriken genauer bewertet werden. Die Rehaklinik zeichnet sich für die Evaluation dadurch aus, dass dort auf den verhältnismäßig schmalen Gängen ein hohes Verkehrsaufkommen durch Patienten, Angehörige und Angestellte herrscht, welche nicht mit dem Roboter vertraut sind. Dadurch kommt es zu sehr vielen natürlichen Interaktionen, ohne dass diese durch gezielte Instruktion der Personen herbei geführt werden müssen. Außerdem ist bei vielen Patienten wegen ihrer eingeschränkten Mobilität und Verletzlichkeit ein sozialverträgliches Verhalten des Roboters besonders wichtig, und die Interaktionen dauern wegen des eingeschränkten Freiraumes verhältnismäßig lange. Dadurch wird es besonders deutlich, wenn sich der Roboter während einer Interaktion nicht ausreichend sozial verhält.

7.1. Evaluationskriterien

7.1.1. Bewertungsmetriken für die Mensch-Roboter-Interaktion

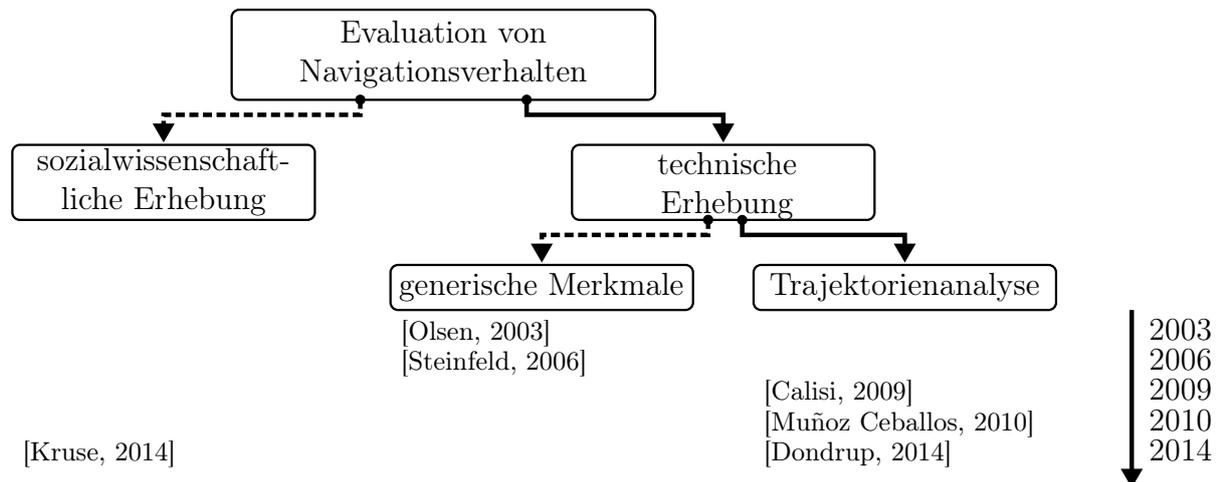


Abbildung 7.1. Arbeiten zur Evaluation von Navigationsverhalten: Die Evaluationsverfahren sind nach Art der Datenerhebung und Jahren eingeordnet.

Die Eignung von Metriken zur Bewertung der Navigationsverhalten dieser Arbeit hängt insbesondere von der Art der Datenerhebung ab. Dementsprechend sind in Abb. 7.1 die Publikationen, in welchen die nachfolgend betrachteten Metriken vorgestellt werden, bzgl. der Datenerhebung eingeordnet.

In [Kruse, 2014] werden zur vergleichenden Bewertung zweier Verfahren zur respektvollen Navigation Bewertungsbögen eingesetzt. 17 Probanden wurden wiederholt in eine vergleichbare Interaktionssituation mit dem Roboter versetzt. Bei jedem Durchlauf wählte der Roboter zufällig eines der beiden Navigationsverfahren, ohne dass den Probanden mitgeteilt wurde, welches Verfahren der Roboter gerade anwendet. Nach jedem Durchlauf bewerteten die Probanden auf einer Skala von 1 bis 5 die Vorhersagbarkeit des Roboterhaltens und die Annehmlichkeit des Interaktionsprozesses.

Der Vorteil solcher sozialwissenschaftlichen Evaluationen ist, dass direkt die Empfindungen der Personen erfasst werden, welche auch das wesentliche Optimierungskriterium darstellen. Nachteilig ist, dass die Bewertung für den Interaktionsprozess als Ganzes gilt und so eine genauere Analyse, was an dem jeweiligen Verhalten verbessert werden sollte, erschwert ist. Auch aufgrund fehlender Expertise im Bereich der sozialwissenschaftlichen Evaluation sollen deshalb während der Experimente dieser Arbeit technisch messbare

Merkmale erhoben werden. Es sollen jedoch Metriken verwendet werden, deren Relevanz für die Sozialverträglichkeit mobiler Roboter bereits in sozialwissenschaftlichen Untersuchungen gezeigt wurde.

Das Thema Benchmarking ist seit über einem Jahrzehnt Gegenstand von Workshops und Konferenztracks internationaler Robotikkonferenzen, wobei z.B. auch technische Bewertungsmetriken für die Mensch-Roboter-Interaktion vorgestellt werden. So werden z. B. in [Olsen, 2003] auf Zeitmessung basierende Metriken zur Bewertung der Kooperation von Menschen und Personen am Beispiel einer gemeinsamen Navigation beschrieben. In [Steinfeld, 2006] werden allgemeine Leistungsmetriken zur Evaluation der Mensch-Roboter-Interaktion beschrieben. Diese umfassen auch aufgabenspezifische Metriken bzgl. Navigation und Sozialverträglichkeit. Die Beschreibung der Metriken ist jedoch sehr allgemein gehalten und unabhängig von den tatsächlichen Aufgaben. Deshalb können auch keine konkreten Metriken aus dieser Veröffentlichung verwendet werden.

Im Gegensatz dazu, werden in [Calisi, 2009] konkrete Metriken zur Bewertung von Navigationsaufgaben beschrieben (Pfadlänge, Kollisionsgefahr, Kurvenwechsel). Ähnliche Metriken werden auch in [Muñoz Ceballos, 2010] vorgeschlagen. So werden Metriken für die Sicherheit (Distanz zu Hindernissen), Dimension (Länge der Trajektorie) und Gleichmäßigkeit (Krümmungen der Trajektorie) vorgestellt. Diese Metriken basieren jedoch ausschließlich auf den Trajektorien des Roboters ohne dass die Interaktion mit einem Menschen bewertet wird. Wie bei der Evaluation der Bewegungsprädiktion (Abschnitt 6.3, [Weinrich, 2013a]) sollen die vorgeschlagenen Metriken deshalb vorrangig auf den Trajektorien der menschlichen Interaktionspartner berechnet werden, um so indirekt das Roboterverhalten zu bewerten.

Dieses Prinzip zur Bewertung der räumlichen Interaktion von Mensch und Roboter wird auch in [Dondrup, 2014] angewendet. Basierend auf [Moon, 2011] wird gezeigt, dass abruptes Abbremsen einer laufenden Person ein Indiz von Unentschlossenheit sein kann und auf ein schlecht vorhersagbares Verhalten des Roboters hindeutet. In [Dondrup, 2014] wurde diese durch einen Roboter hervorgerufen, welcher sich in einer frontalen Ausweichsituation einer Person zu sehr näherte und in deren persönliche Zone eindrang.

Der Vorteil der Metriken in [Calisi, 2009, Muñoz Ceballos, 2010, Dondrup, 2014] ist, dass sie mit technischen Verfahren, wie einem externem Trackingsystem, basierend auf den Trajektorien des Roboters und der Probanden erhoben werden und so über den gesamten Interaktionsprozess analysiert werden können. Nachfolgend werden die zur Evaluation verwendeten Metriken noch einmal aufgeführt, bevor die tatsächlichen Experimente zur Evaluation der einzelnen Verhaltensweisen beschrieben werden. Zum Tracking des Probanden und des Roboters wird ein auf [Schenk, 2011] basierendes verteiltes System aus

Laserscannern verwendet. Die Erweiterungen dieses Systems sind in einer im Rahmen dieser Arbeit betreuten Masterarbeit [Seidl, 2015] beschrieben.

7.1.2. Verwendete Metriken

Wegstrecke der Person und des Roboters Besteht die Aufgabe darin, sich von einer Ausgangsposition zu einer Zielposition zu bewegen, ist die Länge des zurückgelegten Pfades ein naheliegendes Kriterium zur Bewertung der Effizienz der Bewegung. Da Menschen im Allgemeinen bestrebt sind, sich effizient fortzubewegen, kann die soziale Verträglichkeit eines Roboters daran bewertet werden, wie groß der Umweg des Probanden ist, welcher in Abhängigkeit von dem Verhalten des Roboters gelaufen wird. Verhält sich ein Roboter respektvoll gegenüber einer Person, sollte die Effizienz der Person steigen, wobei sich die Effizienz des Roboters möglicherweise reduziert. Deshalb wird bei der Bewertung der Navigationsverhalten sowohl die Wegstrecke der Person als auch des Roboters ausgewertet.

Distanz zwischen Person und Roboter Im Kontext der sozialverträglichen Navigation ist die Proxemik [Hall, 1963] von wesentlicher Bedeutung. Bei der Bewertung der respektvollen Navigationsverhalten wird deshalb insbesondere die minimale Distanz während der Interaktion ausgewertet. Bei den personenfokussierten Navigationsverhalten zum *Lotsen* und *Begleiten von Personen* ist es hingegen wichtig, dass die Interaktion zwischen Nutzer und Roboter nicht abreißt. Deshalb sind im Kontext dieser Verhaltensweisen vor allem die maximalen Distanzwerte relevant.

Krümmung Die Gleichmäßigkeit einer Trajektorie zeigt laut [Muñoz Ceballos, 2010] die Fähigkeit zu antizipieren und rechtzeitig auf Ereignisse zu reagieren. Um zu bewerten, wie gut sich das Roboterverhalten vorhersagen lässt, wird deshalb in dieser Arbeit die Gleichmäßigkeit der Trajektorie des Probanden genutzt. Als Maß für die Gleichmäßigkeit werden die Krümmungen entlang der Trajektorie abgetastet und gemittelt. Die Berechnung einer einzelnen Krümmung κ erfolgt entsprechend der Menger Krümmung basierend auf drei Punkten der Trajektorie. Sie ist reziprok zum Radius des Kreises, welcher durch diese drei Punkte führt. Die Krümmung κ ergibt sich aus der Fläche A , welche durch die drei Punkte aufgespannt wird und dem Produkt der drei Abstände a, b, c zwischen den Punkten.

$$\kappa = \frac{4A}{abc} = \frac{4\sqrt{s(s-a)(s-b)(s-c)}}{abc} \quad \text{mit } s = \frac{a+b+c}{2} \quad (7.1)$$

In Abb. 7.2 ist beispielhaft für verschiedene Trajektorien die durchschnittliche Krümmung gezeigt.

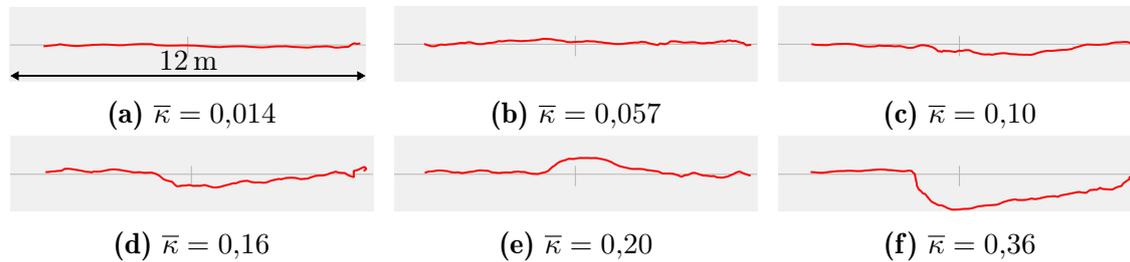


Abbildung 7.2. Krümmung von Beispieltrajektorien: Gezeigt ist die durchschnittliche Krümmung von sechs Trajektorien. Jede Trajektorie wurde im Abstand von 20 cm abgetastet und für diese Punkte wurde die Krümmung bestimmt und über alle Punkte gemittelt. Da die Berechnung eines Krümmungswertes auf drei Punkten basiert, wurden zwei weitere Punkte der Trajektorie im Abstand von jeweils 80 cm verwendet. (Übernommen aus einer im Rahmen dieser Arbeit betreuten Masterarbeit [Seidl, 2015] und angepasst)

Geschwindigkeitsspannweite Ähnlich der Krümmung widersprechen auch Geschwindigkeitsänderungen einer gleichmäßigen Bewegung und stellen ein Anzeichen von Unentschlossenheit dar [Dondrup, 2014]. Um zu bewerten, ob der Roboter Unentschlossenheit bei den Probanden hervorruft, wird das Geschwindigkeitsprofil der Probanden während der Interaktion mit dem Roboter ausgewertet. Bewertungskriterium ist die Differenz Δv zwischen maximaler und minimaler Geschwindigkeit der Person.

Hautleitwert Der Hautleitwert eines Menschen wird durch die Aktivität der Schweißdrüsen (elektrodermale Aktivität) beeinflusst. Gemäß [Bösel, 1987, S. 99 f.] führt Schreck, hohe Aufmerksamkeit, freudige oder ängstlicher Anspannung und Muskeltätigkeit nach 1,3 bis 2,5 Sekunden zu einer Erhöhung des Hautleitwerts. Weil der Hautleitwert nicht bewusst¹ beeinflusst werden kann, wird er bei psychophysiologischen Experimenten zur objektiven Bewertung eingesetzt. Unter der Annahme, dass der Roboter während der Interaktion mit Personen je nach Verhalten unterschiedlich viel Anspannung bei den Personen hervorruft, wurde im Rahmen dieser Arbeit eine Masterarbeit [Seidl, 2015] betreut, welche den Zusammenhang zwischen Roboterverhalten und Hautleitwert untersucht. Während der nachfolgend beschriebenen Experimente zur respektvollen Navigation wurde deshalb auch der Hautleitwert zwischen dem vorletzten Glied des Zeige- und des

¹Nur mittels spezieller, erlernter Techniken soll eine indirekte Beeinflussung des Hautleitwertes möglich sein.

Mittelfingers aufgezeichnet. Ein Zusammenhang zwischen dem Hautleitwert und den Verhaltensweisen konnte jedoch nicht nachgewiesen werden. Eine Erklärung dafür ist, dass der Hautleitwert neben der Anspannung der Probanden auch durch die übrigen, oben aufgeführten Ursachen beeinflusst wird. Insbesondere kann während der Experimente die Muskeltätigkeit der Probanden nicht verhindert werden, und diese ist laut [Boucsein, 1992, S. 124] eine wesentliche Ursache für Artefakte. Deshalb ist hier nur die Grundidee und ein erstes Fazit beschrieben, ohne dass der Hautleitwert in dieser Arbeit zur Bewertung des Roboterverhaltens genutzt wird. Für weitere Details zu dieser Problematik sei auf [Seidl, 2015] verwiesen.

7.2. Respektvolle Navigation

7.2.1. Ausweichen vor Personen

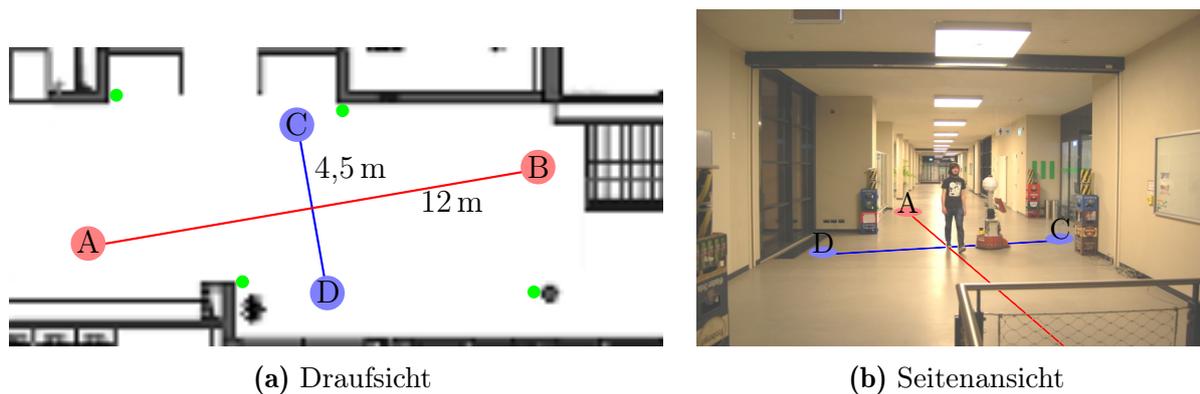


Abbildung 7.3. Versuchsaufbau zur respektvollen Navigation: Der jeweilige Proband läuft bei jedem Durchlauf des Experiments von der Position A nach B bzw. umgekehrt. Sobald er sich 0,8 m von der jeweiligen Startposition entfernt hat, wird dem Roboter die Aufgabe gegeben, sich zu der Position C bzw. D zu bewegen, so dass sich deren Wege kreuzen. Der Roboter verfolgt bei jedem Durchlauf eines von vier evaluierten Navigationsverhalten und ein externes Trackingsystem (grün) zeichnet die Trajektorien des Probanden und des Roboters auf.

Zur Evaluation des Roboterverhaltens beim Ausweichen vor Personen wurde der in Abbildung 7.3 gezeigte Versuchsaufbau in der Einsatzumgebung des Gebäudelotsen gewählt. Der Aufbau erlaubt die Untersuchung des respektvollen Navigationsverhaltens, während sich die Wege des Roboters und eines Menschen kreuzen. Die Probanden wurden gebeten, sich möglichst zielgerichtet zwischen den Positionen A und B, welche auf dem Fußboden markiert waren, hin und her zu bewegen. Sobald der jeweilige Proband einen Abstand

von 0,8 m zu dessen Startposition überschritten hatte, wurde durch ein externes Evaluationssystem² der aktuelle Task (Abschnitt 3.1.3) des Roboters gesetzt, wodurch wiederum das *Path Objective* (siehe Kapitel 3.1.2) mit der entsprechenden Zielposition C bzw. D aktiviert wurde. Wenn Person und Roboter ihre Zielposition erreicht haben, wird durch einen Ton signalisiert, dass der Durchlauf abgeschlossen ist, und die Person kann mit einem weiteren Durchgang beginnen. Für jeden Durchlauf wählt das Evaluationssystem zufällig einen von vier verschiedenen Tasks³ für den Roboter, wodurch eine unterschiedlich starke Ausprägung des respektvollen Navigationsverhaltens erreicht wird:

Geradlinig: Aktivierung des *Path Objective* mit Zielposition, so dass sich die Wege von Person und Roboter kreuzen, jedoch keine Aktivierung des *Personal Space Objective*

Respektvoll: Aktivierung des *Path Objective* mit Zielposition, so dass sich die Wege von Person und Roboter kreuzen, und Aktivierung des *Personal Space Objective* mit geringem Personal Space ($\sigma = 0,1$ in Gleichung C.4)

Vorsichtig: Aktivierung des *Path Objective* mit Zielposition, so dass sich die Wege von Person und Roboter kreuzen, und Aktivierung des *Personal Space Objective* mit großem Personal Space ($\sigma = 0,3$ in Gleichung C.4)

Reglos: Aktivierung des *Path Objective* mit Zielposition, welche der aktuellen Startposition entspricht, so dass der Roboter nicht den Weg der Person kreuzt

Die Probanden hatten keine Kenntnis, welche Verhaltensausrprägung der Roboter während des jeweiligen Durchlaufs anwendete. Das Evaluationssystem hingegen extrahierte die zuvor beschriebenen Bewertungsmetriken aus den Trajektorien der Probanden und des Roboters und ordnete diese der jeweiligen Ausprägung zu. Für dieses Experiment wurden insgesamt 540 Durchläufe mit 11 Probanden (2 weibliche Studierende, 2 männliche Studierende und 7 männliche Mitarbeiter des Fachgebiets NIKR) durchgeführt. Die Durchschnittsgeschwindigkeit über alle Probanden beträgt $1,28 \text{ m/s} \approx 4,6 \text{ km/h}$. Obwohl sich die Probanden und der Roboter während des Experiments nicht weit voneinander entfernen, ist der Versuchsaufbau so gewählt, dass der Roboter den Probanden nicht immer wahrnehmen kann. Wenn sich der Roboter auf Position D befindet und sich der Proband von A nach B bewegt, wird die Person durch eine Wand verdeckt, und der Roboter kann die Person erst tracken, wenn sich die Person und der Roboter bereits bewegen. In Abb. 7.4 sind für die vier Verhaltensausrprägungen beispielhaft jeweils drei Trajektorien der Person und des Roboters gezeigt.

²siehe Masterarbeit [Seidl, 2015], welche im Rahmen dieser Arbeit betreut wurde.

³Die Gewichte der Objectives sind in Tabelle 3.1 aufgeführt.

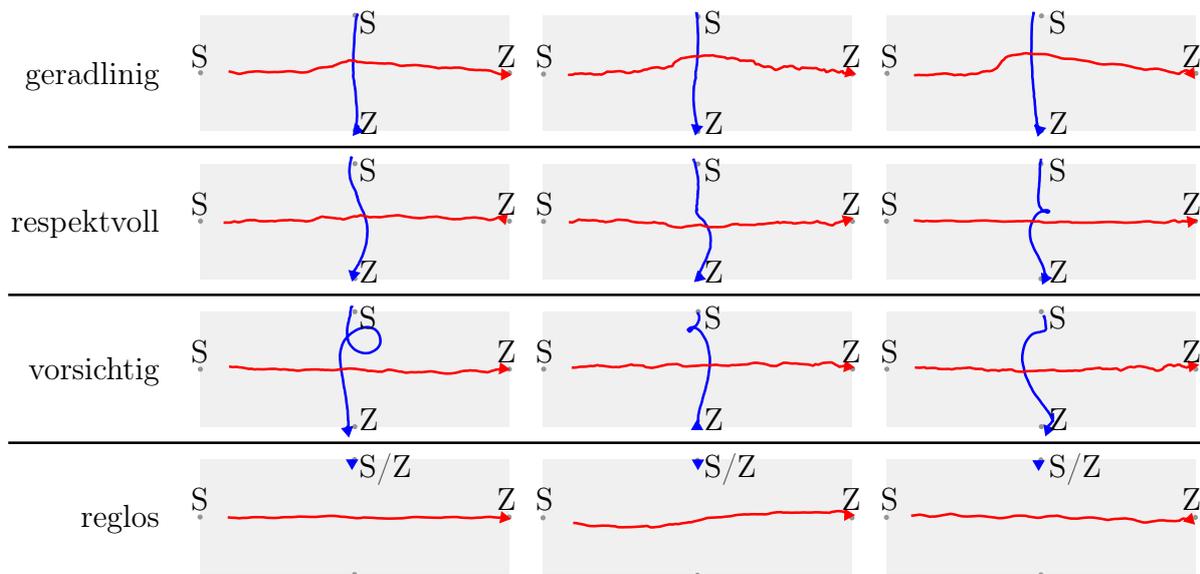


Abbildung 7.4. Beispieltrajektorien zur respektvollen Navigation: Für jede der vier Verhaltensausprägungen sind beispielhaft jeweils drei Trajektorien der Person (rot) und des Roboters (blau) gezeigt. Der Proband läuft in jeder Grafik von der linksseitigen Startposition S zu der rechtsseitigen Zielposition Z . Entsprechend fährt der Roboter jeweils von oben nach unten oder bleibt oben stehen. (Übernommen aus [Seidl, 2015] und angepasst)

Wegstrecken Wie Tabelle 7.1 zeigt, hat die gewählte Verhaltensausprägung einen signifikanten Einfluss auf die Wegstrecke des Roboters. Diese Strecke übersteigt auch bei dem geradlinigen Verhalten den Abstand zwischen der Start- und Zielposition von 4,5 m (Abb. 7.3), weil der Roboter zum Erreichen der Zielpose einen Bogen fährt, um am Ziel mit einer Orientierung in Richtung des zukünftigen Ziels zum Stillstand zu kommen. Erwartungsgemäß erhöht sich durch die respektvolle bzw. die vorsichtige Verhaltensausprägung die Wegstrecke des Roboters zusätzlich. Entscheidend ist jedoch, dass sich dies zugunsten einer effizienteren Bewegung der Personen auswirkt. Die Wegstrecke der unbehinderten Person bei reglosem Roboterverhalten beträgt durchschnittlich 10,94 m. Dieser Wert ist niedriger als der Abstand zwischen Start- und Zielposition von 12,0 m (Abb. 7.3), weil die ersten 0,8 m und die letzten 0,3 m der Trajektorie des Probanden nicht bewertet werden. Diese Bereiche fließen nicht in die Bewertung ein, weil die Beschleunigung der Personen am Anfang und das Positionieren auf dem Ziel, insbesondere bei den später behandelten Bewertungskriterien, nicht einfließen soll. Die Reduzierung der Wegstrecke der Person durch das respektvolle Roboterverhalten ist gegenüber der geradlinigen Verhaltensausprägung nach dem T-Test mit Signifikanzniveau $\alpha = 0,05$ signifikant. Die Wegstrecken der Person reduzieren sich jedoch durch das vorsichtige Verhalten bzw. das reglose Verhalten des Roboters nicht mehr wesentlich.

Die nachfolgenden Bewertungsmetriken werden zeigen, dass sich der Unterschied zwischen der respektvollen und der vorsichtigen Verhaltensausprägung dennoch auf die Bewegung der Person auswirkt. Dies deutet darauf hin, dass die Probanden ihre Trajektorie eher bzgl. der übrigen Kriterien an das Roboterverhalten anpassen und es möglichst vermeiden, einen Umweg zu gehen.

Tabelle 7.1. Wegstrecken und Minimalabstand von Person und Roboter: Für die vier Ausprägungen des respektvollen Roboterhaltens sind die Wahrscheinlichkeitsverteilungen der Versuchsdurchführungen über die Bewertungskriterien aufgeführt. Die Wahrscheinlichkeitsverteilungen, welche sich nach dem T-Test mit dem Signifikanzniveau $\alpha = 0,05$ signifikant unterscheiden, sind durch eine horizontale Linie getrennt.

Verhaltensausprägung	# Durchläufe	Wegstrecke [m]		Minimale Distanz von Person & Roboter [m]
		Person	Roboter	
geradlinig	137	$\varnothing 11,1 \pm 0,25$	$\varnothing 5,0 \pm 0,4$	$\varnothing 0,67 \pm 0,17$
respektvoll	136	$\varnothing 11,0 \pm 0,22$	$\varnothing 5,4 \pm 1,1$	$\varnothing 0,95 \pm 0,32$
vorsichtig	135	$\varnothing 10,9 \pm 0,19$	$\varnothing 5,9 \pm 1,4$	$\varnothing 1,23 \pm 0,31$
reglos	132	$\varnothing 10,9 \pm 0,28$	$\varnothing 0,1 \pm 0,5^1$	$\varnothing 2,23 \pm 0,21$

¹ Obwohl sich der Roboter beim reglosen Verhalten nicht bewegt, wurde durch das Tracking-System eine Wegstrecke von 0,1 m gemessen. Dieser Wert entspricht der Summe des statistischen Fehlers bei der Positionsbestimmung des stehenden Roboters mit 10 Hz über einen gesamten Durchlauf (ca. 85 Messungen).

Minimale Distanz von Person und Roboter Die Distanz zwischen Person und Roboter ist das wesentliche Optimierungskriterium des *Personal Space Objective*. Entsprechend stark unterscheidet sich die minimale Distanz von Person und Roboter bei den unterschiedlichen Verhaltensausprägungen. Beim geradlinigen Verhalten versucht der Roboter, eine Kollision zu vermeiden. Dennoch kam es in verschiedenen Durchläufen vor, dass die Person den Roboter am Stoßleistsensor berührt hat, weil sie selbst gegen den Roboter gelaufen ist. Die in Tabelle 7.1 aufgeführten Distanzen geben den minimalen Abstand zwischen dem Mittelpunkt der Person und des Roboters an. Subtrahiert man den Radius des Roboters von ca. 0,3 m und die Ausdehnung der Person, wird die Steigerung des Mindestabstandes um ca. 0,28 m durch das respektvolle Verhalten deutlich. Das vorsichtige Verhalten des Roboters führt zu einer weiteren Steigerung des Mindestabstandes um ca. 0,28 m. Die minimale Distanz zwischen dem reglosen Roboter und der Person von 2,23 m entspricht dem minimalen Abstand des direkten Weges der Person und den Roboterpositionen (Abb. 7.3).

Krümmung Die mittlere Krümmung der Robotertrajektorie steht im Zusammenhang mit dessen Wegstrecke, da Kurvenfahrten des Roboters einen Umweg darstellen. Entspre-

chend unterscheiden sich die durchschnittlichen Krümmungen der Robotertrajektorien zwischen den vier Verhaltensausrägungen ebenso signifikant (Tab. 7.2). Die Krümmung der menschlichen Trajektorie liefert im Vergleich zur Wegstrecke jedoch deutlichere Anhaltspunkte, ob die Person die Roboterbewegung antizipieren und den Umweg vorausschauend planen konnte, oder ob die Person spontan auf den Roboter reagieren musste. Tabelle 7.2 zeigt, dass sich die mittlere Krümmung der Personentrajektorie durch das respektvolle bzw. vorsichtige Verhalten gegenüber dem geradlinigen Verhalten signifikant reduziert. Der reglose Roboter ist für die Probanden ideal prädzierbar und so ist der Unterschied dieser Verhaltensausrägung gegenüber den anderen nach dem T-Test signifikant. Dennoch ist die Differenz der mittleren Krümmung zwischen geradliniger und respektvoller Verhaltensausrägung weitaus größer als zwischen respektvollem und reglosem Verhalten.

Tabelle 7.2. Mittlere Krümmung der Trajektorien und Geschwindigkeitsänderung: Für die vier Ausprägungen des respektvollen Navigationsverhaltens sind die Wahrscheinlichkeitsverteilungen der Versuchsdurchführungen über die Bewertungskriterien aufgeführt. Die Wahrscheinlichkeitsverteilungen, welche sich nach dem T-Test mit dem Signifikanzniveau $\alpha = 0,05$ signifikant unterscheiden, sind durch eine horizontale Linie getrennt.

Verhaltensausrägung	# Durchläufe	Mittlere Krümmung $\kappa \left[\frac{0,1}{m} \right]$		Geschwindigkeitsspannweite der Person $\Delta v \left[\frac{m}{s} \right]$
		Person	Roboter	
geradlinig	137	$\emptyset 1,21 \pm 0,59$	$\emptyset 1,14 \pm 1,47$	$\emptyset 0,89 \pm 0,36$
respektvoll	136	$\emptyset 0,68 \pm 0,35$	$\emptyset 3,49 \pm 3,13$	$\emptyset 0,51 \pm 0,19$
vorsichtig	135	$\emptyset 0,68 \pm 0,25$	$\emptyset 4,89 \pm 4,18$	$\emptyset 0,45 \pm 0,15$
reglos	132	$\emptyset 0,59 \pm 0,23$	$\emptyset 0,00 \pm 0,10$	$\emptyset 0,38 \pm 0,12$

Geschwindigkeitsspannweite Anstatt dem Roboter auszuweichen, kann der Proband auch abbremsen, um dem Roboter den Vortritt zu lassen. Dieses spontane Abbremsen wird durch die Geschwindigkeitsspannweite erfasst. Tabelle 7.2 zeigt, dass sich die Spannweite bei allen vier Verhaltensausrägungen signifikant unterscheidet. Dabei ist der Unterschied zwischen dem geradlinigen und dem respektvollen Verhalten besonders groß. Diese Ergebnisse können im Vergleich mit der mittleren Krümmung und der Wegstrecke so gedeutet werden, dass Menschen zur Vermeidung einer Kollision mit dem Roboter vor allem ihre Geschwindigkeit reduzieren. Ist das Verhalten des Roboters nicht gut vorhersagbar, weichen sie dem Roboter zusätzlich aus und müssen deshalb einen längeren Weg zum Ziel zurücklegen.

7.2.2. Abstand zu Personen in der Rehaklinik

Die zuvor beschriebenen Experimente im Fakultätsgebäude zeigen, dass sich das Roboterverhalten im Sinne einer respektvollen Navigation beeinflussen lässt. Die vorsichtige Verhaltensausprägung ($\sigma = 0,3$ in Gleichung C.4) zeichnet sich durch eine hohe soziale Verträglichkeit aus, führt jedoch auch zu einer wesentlich geringeren Effizienz des Roboters. Die respektvolle Verhaltensausprägung ($\sigma = 0,1$) führt zu einer wesentlich besseren Effizienz des Roboters bei geringfügig schlechterer Sozialverträglichkeit und ist somit besser für den praktischen Einsatz geeignet. Deshalb soll die respektvolle Ausprägung im Vergleich zur geradlinigen Ausprägung in der Rehaklinik untersucht werden. Dort werden keine festen Interaktionssituationen vorgegeben. Stattdessen navigiert der Roboter⁴ mehrere Kilometer zwischen zwei Zielpositionen in der Klinik (Abb. 7.5) und muss dabei den zufällig anwesenden Personen auf den Gängen ausweichen. Während der Durchläufe mit geradliniger bzw. respektvoller Verhaltensausprägung wurde der Roboter in größerem Abstand von einer Person begleitet, welche über eine Funkschaltung protokollierte, wann sich Personen in der Nähe des Roboters befanden und ob der Roboter deren Personal Space verletzte. Diese subjektive Bewertung ist zwar abhängig von der protokollierenden Person, hat jedoch den Vorteil, dass die Person die aktuelle Gesamtsituation einbeziehen kann, wenn sie nach menschlichem Empfinden entscheidet, ob der persönliche Raum einer Person verletzt wurde. Dies ist auch vorteilhaft im Vergleich zu nachträglichen Evaluationen mittels Fragebögen. Tabelle 7.3 zeigt, dass durch das respektvolle Verhalten gegenüber dem geradlinigen Verhalten mehr als 93% der Verletzungen des Personal Space verhindert werden konnten. Es wird vermutet, dass sich Verletzungen des Personal Space aufgrund der vielen Personen und der schmalen Gänge selbst bei idealem Roboterverhalten nicht vollständig ausschließen lassen.

Tabelle 7.3. Verletzung des Personal Space: Während der Roboter entsprechend der Verhaltensweisen in der Rehaklinik zu den Navigationspositionen fuhr, wurde er in größerem Abstand durch einen Beobachter begleitet. Dieser protokollierte mittels eines Funkschalt-systems, wann sich Personen im 2 m Radius des Roboters befanden, und entschied, ob der Personal Space von Personen durch den Roboter verletzt wurde. (siehe [Scheidig, 2015b])

Eigenschaften Verhalten	Gefahrene Strecke	Relevante Personen(gruppen)	Verletzungen des Personal Space
geradlinig	1.500 m	24 (16,0 pro km)	11 (7,3 mal pro km)
respektvoll	8.000 m	133 (16,6 pro km)	4 (0,5 mal pro km)

⁴Im Rahmen des Projektes zum Rehabilitationsroboter wurde durch den Projektpartner MetraLabs eine neue Roboterplattform ROREAS (siehe [Scheidig, 2015b]) entwickelt. Bei diesem Experiment zur respektvollen Navigation wurde bereits die neue Plattform eingesetzt, ohne dass eine Anpassung der Algorithmen vorgenommen wurde.

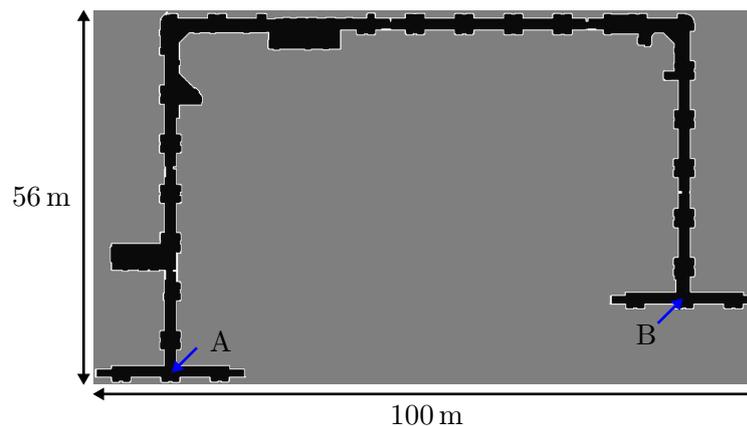


Abbildung 7.5. Einsatzumgebung Rehaklinik: Die Experimente zur respektvollen Navigation und zum Rechtsfahren wurden im 2. Obergeschoss der Rehaklinik durchgeführt. Während der Experimente ist der Roboter mit geradlinigem Verhalten, mit respektvollem Verhalten sowie mit zusätzlicher Aktivierung des Rechtsfahrverhaltens zwischen den Positionen A und B hin und her gefahren.

7.2.3. Rechtsfahren

Bei den Experimenten zum Rechtsfahren in der Rehaklinik wurde der gleiche Versuchsaufbau wie bereits bei der respektvollen Navigation verwendet, bei welchem der Roboter mit und ohne Rechtsfahrverhalten zwischen zwei Positionen hin und her fährt (Abb. 7.5).

Simulation für Rehaklinik Um die Auswirkungen des Rechtsfahrverhaltens ohne den Einfluss von Personen oder dynamischen Hindernissen zu zeigen, sind in Abb. 7.6 die im Simulator aufgezeichneten Trajektorien des Roboters mit und ohne Rechtsfahrverhalten dargestellt. Ohne *Two-Way Traffic Objective* (Abb. 7.6a und 7.6b) ist das Verhalten vorrangig durch das *Path Objective* geprägt, dessen Optimierungskriterium die möglichst schnelle Annäherung an das Ziel ist. Außerdem ist das *Distance Objective* aktiv, welches verhindert, dass sich der Roboter Hindernissen zu sehr annähert. Dies hat zur Folge, dass der Roboter in längeren Gangpassagen mittig des Ganges fährt, weil dadurch der Abstand zu den Wänden maximal ist und der Roboter mit maximaler Geschwindigkeit fahren kann. In diesen Passagen können entgegenkommende Personen frühzeitig wahrgenommen werden, so dass rechtzeitig durch das *Personal Space Objective* mit dem Ausweichen begonnen werden könnte. Kritischer ist das Umfahren von Ecken, da bei diesen der Gegenverkehr erst kurzfristig wahrgenommen werden kann. Abb. 7.6a und 7.6b zeigen deutlich, dass bei Abbiegungen ähnlich der Ideallinie im Rennsport abgekürzt wird und sich die entgegenkommenden Trajektorien kreuzen.

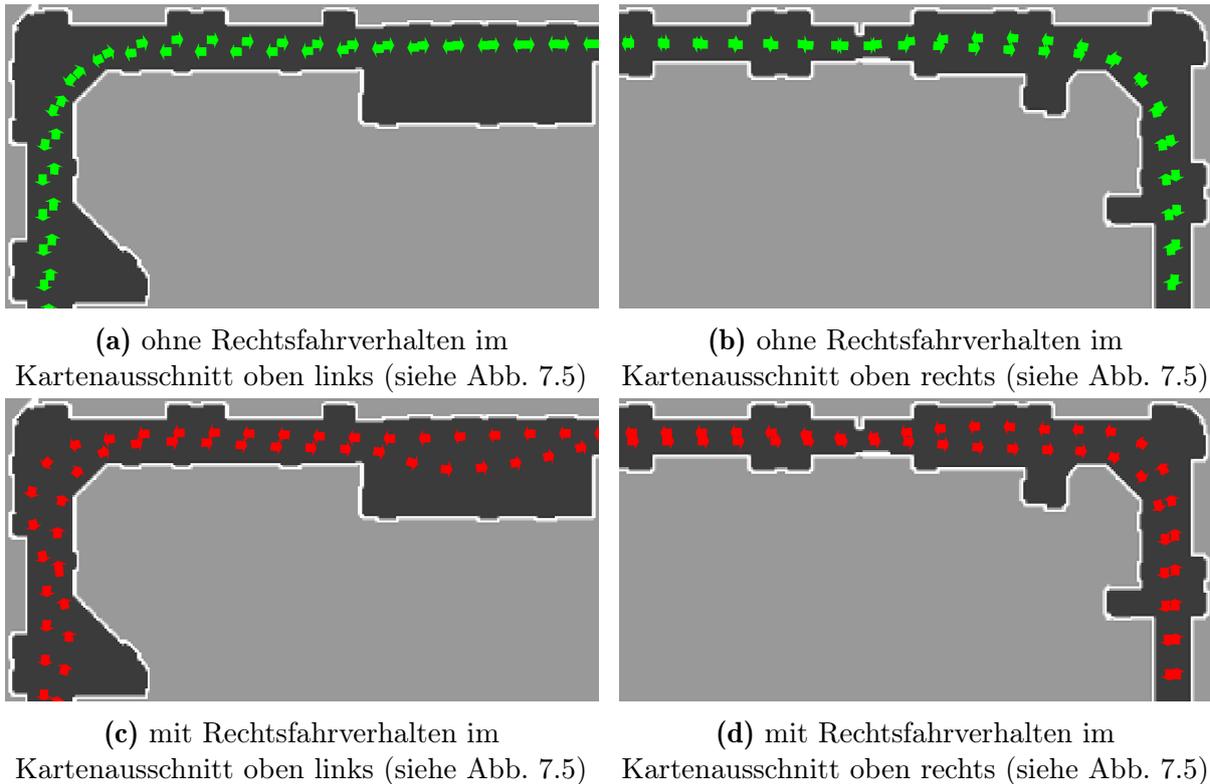
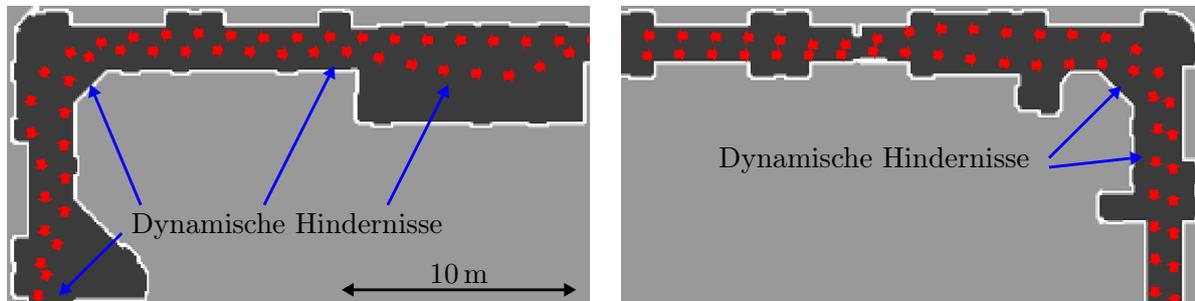


Abbildung 7.6. Simuliertes Rechtsfahrverhalten: Dargestellt sind die Trajektorien des Roboters Cora, welche im Simulator ohne (grüne Pfeile in (a) und (b)) und mit (rote Pfeile in (c) und (d)) Rechtsfahrverhalten basierend auf der Karte des Rehabilitationszentrums (Abb. 7.5) aufgezeichnet wurden.

Dies wird durch zusätzliche Aktivierung des *Two-Way Traffic Objective*⁵ verhindert. Abb. 7.6c und 7.6d zeigen, dass die Kurven dadurch so weit ausgefahren werden, dass die entgegenkommenden Trajektorien selbst in den Kurven aneinander vorbei führen. Auch in den schmalen geraden Gangabschnitten führen die entgegenkommenden Trajektorien deutlich aneinander vorbei. Dadurch können Personen, welche auf den Roboter zulaufen, frühzeitig erkennen, auf welcher Seite sie dem Roboter ausweichen können. In breiteren Gangabschnitten ist es dem Roboter tatsächlich möglich, den gewünschten linksseitigen Abstand $d_{l,des} = 1,0\text{ m}$ zu Hindernissen einzunehmen. So ist insbesondere in Abb. 7.6c zu erkennen, dass der Roboter die lange Ausbuchtung in einem gleichmäßigen Bogen durchfährt. Ohne Rechtsfahrverhalten (Abb. 7.6a und 7.6b) beträgt der durchschnittliche Abstand zwischen den entgegenkommenden Trajektorien $-0,01\text{ m}$. Der Roboter ist folglich ohne Rechtsfahrverhalten im Durchschnitt etwas links der entgegenkommenden Trajektorie gefahren. Durch das Rechtsfahrverhalten konnte der linksseitige Abstand zwischen den

⁵Parametrierung: betrachtete Pfadlänge $d_{p,max} = 2,5\text{ m}$, beidseitiger Mindestabstand $d_{min} = 0,0\text{ m}$, gewünschter linksseitiger Abstand $d_{l,des} = 1,0\text{ m}$

entgegenkommenden Trajektorien auf 0,50 m gesteigert werden. Dadurch ist jedoch auch die Wegstrecke um ca. 1% gestiegen. Entscheidender ist jedoch, dass der Roboter aufgrund der Nähe zu den rechtsseitigen Hindernissen langsamer fährt und so die Fahrzeit um ca. 4,7% gestiegen ist.



(a) Ausschnitt aus Abb. 7.5 oben links

(b) Ausschnitt aus Abb. 7.5 oben rechts

Abbildung 7.7. Reales Rechtsfahrverhalten: Im Gegensatz zur Simulation (Abb. 7.6 (c) und (d)) ist hier das reale Roboterverhalten des Prototypen ROREAS eingesetzt und das Gewicht des Two-Way Traffics von 1 auf 80 erhöht. Darüber hinaus werden die Trajektorien durch dynamische Hindernisse, wie Stühle und Stationswagen (blaue Pfeile), welche nicht in der Karte sichtbar sind, beeinflusst.

Reale Experimente in der Rehaklinik Neben der Simulation des Rechtsfahrverhaltens wurde das Verhalten auch mit einem realen Roboter in der Rehaklinik getestet. Wie bereits bei dem Experiment zum Personal Space (Abschnitt 7.2.2) wurde auch hier der Prototyp ROREAS [Scheidig, 2015b] eingesetzt, welcher im Rahmen des Projektes zum Rehabilitationsassistenten entwickelt wurde. In Bezug auf dieses Experiment unterscheidet sich ROREAS bzgl. der Breite mit 0,45 m von der Experimentierplattform Cora (0,6 m) deutlich. Darüber hinaus war die Maximalgeschwindigkeit von ROREAS zum Zeitpunkt der Experimente auf 0,8 m/s reduziert, wohingegen Coras Maximalgeschwindigkeit 1,0 m/s beträgt. Entscheidend ist auch, dass das Gewicht des *Two-Way Traffic Objective* von 1 auf 80 erhöht wurde (vgl. Tabelle 3.1). Abb. 7.7a und 7.7b zeigen im Vergleich zu Abb. 7.6c und 7.6d, dass ROREAS dem Gegenverkehr etwas mehr Freiraum lässt. So wurde der durchschnittliche linksseitige Abstand zwischen den Trajektorien noch einmal auf 0,83 m gesteigert. Die Erhöhung des Gewichtes auf $\alpha_{TWT} = 80$ hat jedoch zur Folge, dass sich das *Two-Way Traffic Objective* stärker gegenüber dem *Path Objective* und dem *Distance Objective* durchsetzt, was dazu führt, dass der Roboter teilweise signifikant langsamer fährt⁶ und sich die Fahrzeit von ROREAS gegenüber der Fahrzeit ohne Rechtsfahrverhalten sogar um fast 10% erhöht hat. Es sollte somit darauf geachtet werden, dass das *Two-Way Traffic Objective* nicht zu stark gewichtet wird.

⁶Im Simulator kommt Cora bei dieser Parametrierung sogar zum Stillstand.

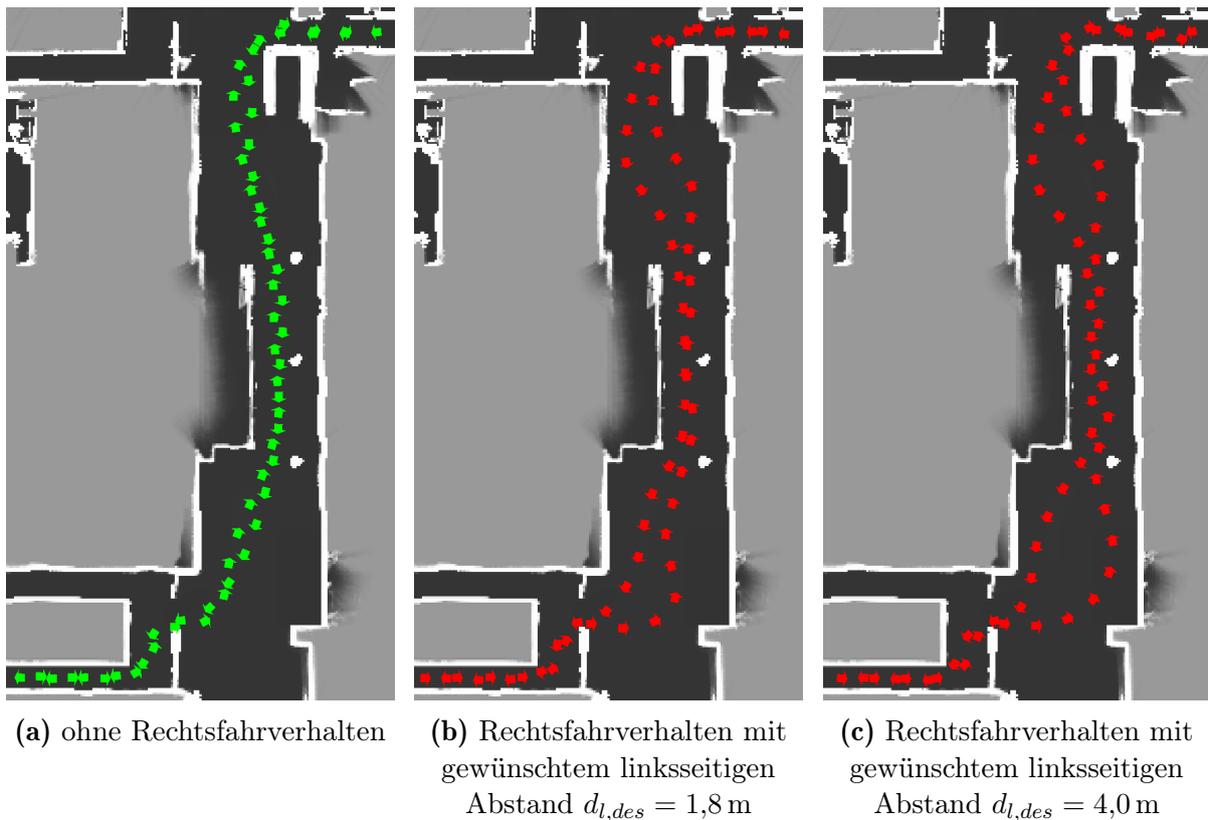


Abbildung 7.8. Rechtsfahrverhalten in der Einsatzumgebung des Lotsenroboters: Dargestellt sind die Trajektorien des Roboters Cora, welche im Einsatzgebiet des Besucherinformationssystems (Abb. 6.3) aufgezeichnet wurden. Bei allen drei Durchläufen ist der Roboter zwischen den selben zwei Positionen hin und zurück gefahren. Dabei war (a) das Rechtsfahrverhalten deaktiviert, (b) der gewünschte linksseitige Freiraum des aktiven Rechtsfahrverhaltens $d_{l,des} = 1,8$ m eingestellt und (c) das Rechtsfahrverhalten mit $d_{l,des} = 4,0$ m aktiviert.

Experimente mit Lotsenroboter Das gleiche Experiment wurde auch mit der Experimentierplattform Cora im Einsatzgebiet des Lotsenroboters (Abb. 6.3) durchgeführt. Ein Unterschied zur Rehaklinik ist, dass hier die Hauptflure sehr breit und die Nebengänge besonders schmal sind. Um den Einfluss des Parameters zum gewünschten linksseitigen Abstand $d_{l,des}$ deutlich zu machen, sind in Abb. 7.8 neben den Trajektorien ohne Rechtsfahrverhalten die Trajektorien mit Rechtsfahrverhalten für zwei verschiedene Parametrierungen von $d_{l,des}$ dargestellt. Durch das Rechtsfahrverhalten steigt der linksseitige Abstand zwischen den entgegenkommenden Trajektorien von $-0,21$ m auf $0,77$ m bei $d_{l,des} = 1,8$ m bzw. $0,83$ m bei $d_{l,des} = 4,0$ m. Die Fahrzeit des Roboters steigt entsprechend um $2,6\%$ bei $d_{l,des} = 1,8$ m bzw. $4,3\%$ bei $d_{l,des} = 4,0$ m. Der geringe Anstieg der Fahrzeit ist mit dem breiten Hauptflur und mit dem ursprünglichen Gewicht $\alpha_{TWT} = 1$ des *Two-Way Traffic Objective* zu erklären.

7.2.4. Fazit zur respektvollen Navigation

Im Kontext der respektvollen Roboternavigation wurden die Wahrung des Personal Space und die Reduktion der Fahrgeschwindigkeit beim Ausweichen vor Personen untersucht. Anhand verschiedener Merkmale konnte gezeigt werden, dass eine respektvolle Navigation einen deutlich positiven Einfluss auf die menschliche Trajektorie hat. So können die Wegstrecke der Personen beim Ausweichen des Roboters und die Krümmung des zurückgelegten Weges signifikant reduziert werden. Noch deutlicher ist die Reduktion der Geschwindigkeitsspannweite der Personen beim Ausweichen des Roboters. Diese konnte durch das respektvolle Ausweichen von durchschnittlich $\Delta v = 0,89$ m/s auf $\Delta v = 0,51$ m/s reduziert werden. Dies ist laut [Dondrup, 2014] ein Indiz für eine bessere Vorhersagbarkeit des Roboterhaltens. In der Rehabilitationsklinik konnte dadurch die Anzahl der Verletzungen des Personal Space von nicht instruierten Personen um 93% reduziert werden. Über die Experimente dieser Arbeit hinausgehend konnte dort beobachtet werden, dass der Roboter auch sich ihm von hinten nähernden Personen bei ausreichend Freiräumen Platz macht und sich in schmalen Gängen von entgegenkommenden Personen zurückdrängen lässt.

Durch das Rechtsfahrverhalten kann verhindert werden, dass sich die Trajektorien zweier sich entgegenkommender Roboter beim Umfahren einer Ecke kreuzen. Dadurch kann möglichen Kollisionen mit Menschen oder Robotern, welche erst bei geringer Distanz durch die Sensorik des Roboters erfasst werden, vorgebeugt werden. Auf den geraden Gangabschnitten können sich Personen und Roboter bereits über größere Distanz wahrnehmen, und auch ohne Rechtsfahrverhalten sollte ein sicheres Ausweichen möglich sein. Durch ein eindeutiges Rechtsfahrverhalten können die Personen jedoch frühzeitig erkennen, auf welcher Seite sie dem Roboter ausweichen können. Da durch das Rechtsfahrverhalten auch die Fahrzeit des Roboters steigt, könnte zukünftig das Rechtsfahrverhalten bei einsehbaren Streckenabschnitten nur aktiviert werden, wenn auch tatsächlich eine Person wahrgenommen wird.

7.3. Lotsen

Beim Lotsen ist es die Aufgabe des Roboters, den Nutzer zu einem bestimmten Ziel zu führen, indem er selbst zu diesem Ziel fährt. Wichtig ist dabei, dass die Person selbst ihre Geschwindigkeit wählen kann, ohne dass der Roboter zu weit vorausfährt oder die Person aufgehalten wird.

Um das Lotsenverhalten des Roboters zu evaluieren, wurde der in Abb. 7.9 dargestellte Versuchsaufbau gewählt. Die blau markierten Positionen (A1, A2, B1, B2) stellen die Ausgangspositionen des Roboters dar. Startet der Proband den Lotsenvorgang durch Berührung des Touchscreens, lotst der Roboter den Probanden zu einem der entfernten Zielpositionen, ohne dass dem Proband bekannt ist zu welchem Ziel. Von der Position A1 würde er beispielsweise entweder zu B1 oder zu B2 gelotst, jedoch nicht zu A2. Sobald der Roboter die Zielposition erreicht hat, signalisiert er dies durch eine akustische Ausgabe, und die Probanden bestätigen ihrerseits das Erreichen des Roboters durch eine weitere Berührung des Touchscreens, wodurch ein Durchlauf des Experiments abgeschlossen ist und mit einem Weiteren begonnen werden kann.

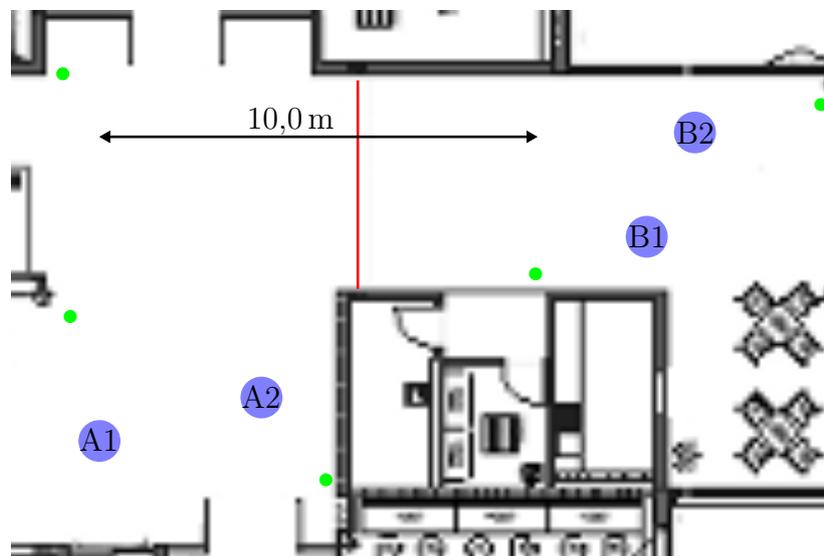


Abbildung 7.9. Versuchsaufbau zum Lotsen: Durch Berührung des Touchscreen können die Probanden einen Lotsenvorgang von den Positionen A1 bzw. A2 zu einer durch das Evaluationssystem gewählten Position B1 bzw. B2 und umgekehrt starten. Es wurden sowohl Experimente durchgeführt, bei welchen die Probanden dem Roboter in einer ihnen angenehmen Geschwindigkeit folgten als auch Experimente, bei welchen die Probanden beim Überqueren einer Markierungslinie (rot) anhalten sollten, bis auch der Roboter zum Stillstand kam. Die Trajektorien der Probanden und des Roboters wurden durch ein externes System (grün) getrackt.

Während der Experimente wurde der Vorsprung des Roboters gegenüber der Person, ab welchem der Roboter abbremst auf $d_G = 1,8\text{ m}$ (siehe Gleichung 3.2) gesetzt. Um dieses Abbremsen untersuchen zu können, wurden die Probanden (2 männliche Studenten und 4 Mitarbeiter) in einem Experiment gebeten, dem Roboter natürlich zu folgen, und in einem weiteren Experiment sollten sie immer beim Überqueren der in Abb. 7.9 rot dargestellten Linie stehen bleiben, bis auch der Roboter zum Stillstand kommt. Die Durchschnittsgeschwindigkeiten der Probanden mit und ohne Anhalten sind in Tab. 7.4 gezeigt.

Tabelle 7.4. Datensatz zum Lotsen: Für alle Probanden ist die Anzahl und die Durchschnittsgeschwindigkeit der Durchläufe mit und ohne Anhalten aufgeführt. Die letzte Zeile zeigt die Summen der Durchläufe bzw. die Durchschnittsgeschwindigkeit aller Probanden.

Proband	mit Anhalten		ohne Anhalten	
	Durchläufe	Geschwindigkeit [m/s]	Durchläufe	Geschwindigkeit [m/s]
1	10	$\varnothing 0,60 \pm 0,026$	10	$\varnothing 0,62 \pm 0,065$
2	10	$\varnothing 0,51 \pm 0,046$	11	$\varnothing 0,57 \pm 0,049$
3	12	$\varnothing 0,62 \pm 0,046$	0	
4	10	$\varnothing 0,57 \pm 0,039$	0	
5	10	$\varnothing 0,52 \pm 0,018$	0	
6	4	$\varnothing 0,57 \pm 0,028$	16	$\varnothing 0,64 \pm 0,046$
	56	$\varnothing 0,57 \pm 0,056$	37	$\varnothing 0,62 \pm 0,059$

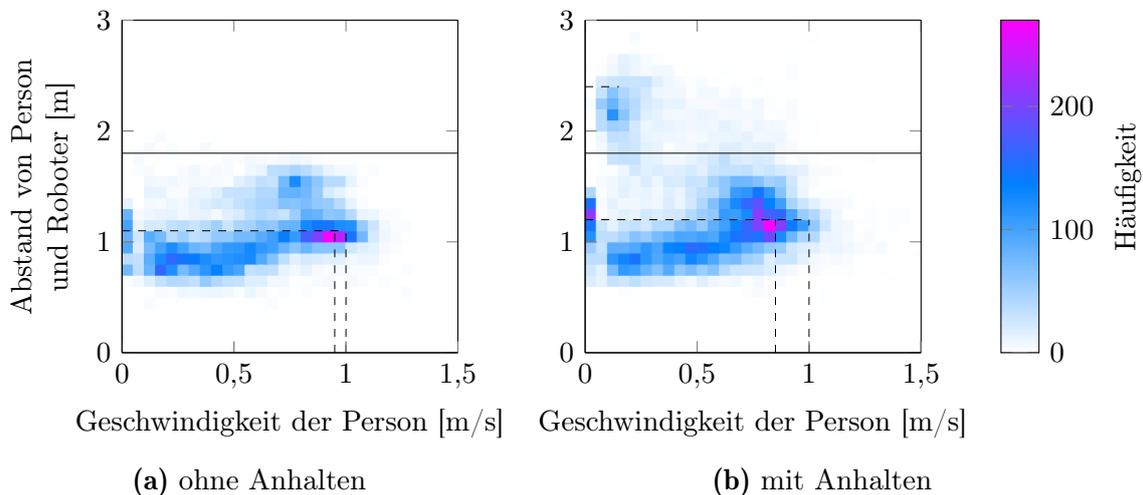


Abbildung 7.10. Personengeschwindigkeit und Abstand zum Roboter: Die Häufigkeitsverteilungen über die Geschwindigkeit der Person und den Vorsprung des Roboters zeigt, dass der Roboter unabhängig von der Geschwindigkeit meist einen Vorsprung von ca. 1,1 m hält. Am häufigsten bewegen sich die Personen mit ca. 0,9 m/s. Der Vergleich der Häufigkeiten (a) ohne und (b) mit Anhalten zeigt, dass der Abstand durch das plötzliche Anhalten der Person auf ca. 2,2 m steigt, bis auch der Roboter zum Stillstand kommt. (Übernommen aus [Seidl, 2015] und angepasst)

Personengeschwindigkeit und Vorsprung des Roboters Während sich die Person und der Roboter gemeinsam bewegen, nehmen sie beide Einfluss auf den Vorsprung des Roboters. Die Häufigkeitsverteilungen in Abb. 7.10 zeigen, dass der Vorsprung des Roboters gegenüber der Person unabhängig von der Geschwindigkeit der Person meist ca. 1,1 m bzw. 1,2 m beträgt. Dies lässt sich damit erklären, dass die Probanden während der Experimente eine Laufgeschwindigkeit bevorzugt hätten, welche über der Maximalgeschwindigkeit des Roboters von 1,0 m/s liegt. Der Vorsprung des Roboters wurde somit

im Wesentlichen durch die Personen und nicht den Roboter bestimmt. Die häufigste Geschwindigkeit mit der sich die Probanden fortbewegt haben, beträgt entsprechend ca. 0,85m/s bzw. 0,95m/s und liegt somit nur knapp unterhalb der Maximalgeschwindigkeit des Roboters. Während der Experimente wird die Geschwindigkeit der Probanden folglich durch die physikalischen Grenzen des Roboters und nicht durch das *Guide Objective* beschränkt.

Neben der Aufgabe, die Nutzer nicht aufzuhalten, besteht die eigentliche Aufgabe des *Guide Objective* darin, den Roboter zu bremsen, wenn dessen Vorsprung zu groß wird. Entscheidend ist deshalb, wie sich der Roboter verhält, wenn die Probanden plötzlich anhalten. Beim Vergleich der Häufigkeitsverteilungen mit und ohne Anhalten (Abb. 7.10a und 7.10b) wird deutlich, dass sich durch das Anhalten häufiger Abstände von ca. 2,2 m ergeben und Abstände von 2,4 m nur sehr selten überschritten werden.

Beim Anwendungsszenario des Rehabilitationsroboters bevorzugen die Nutzer im Gegensatz zu diesem Experiment eine Geschwindigkeit unterhalb der Maximalgeschwindigkeit des Roboters. Deshalb konnte dort ein gleichmäßiges Lotsenverhalten mit größerem Vorsprung, welcher maßgeblich durch den Roboter beeinflusst wurde, beobachtet werden. Mangels eines externen Trackingsystems in dieser Einsatzumgebung sind diese Daten jedoch nicht technisch erfasst wurden.

Vorsprung, Relativgeschwindigkeit und Beschleunigung des Roboters Um das Abbremsen, welches die wesentliche Aufgabe des *Guide Objective* ist, genauer zu evaluieren, ist in Abb. 7.11 der Zusammenhang zwischen dem Vorsprung und der Relativgeschwindigkeit des Roboters gegenüber der Person, sowie der Beschleunigung des Roboters gezeigt. Es ist deutlich zu erkennen, dass der Roboter bei einem Vorsprung über 1,8 m und positiver Relativgeschwindigkeit seine Fahrt verzögert um den Vorsprung nicht weiter zu vergrößern. Darüber hinaus ist zu erkennen, dass der Roboter bei einem Vorsprung unter 1,8 m und negativer Relativgeschwindigkeit tendenziell eher beschleunigt. Damit entspricht das gemessene Verhalten des Gesamtsystems im realen Einsatz der Kostenfunktion des *Guide Objective* (Gleichung 3.2).

Fazit zum Lotsen Die mit dem externen Trackingsystem aufgezeichneten Trajektorien zeigen, dass das gemessene Verhalten dem gewünschten Verhalten entspricht und der Roboter entsprechend abbremst, um der Person selbst bei plötzlichem Anhalten nicht weiter als 2,4 m voraus zu fahren. Darüber hinaus wurde gezeigt, dass nicht das *Guide Objective* sondern die physikalischen Grenzen des Roboters dazu führen, dass vitale Personen nicht ihre gewünschte Lotsengeschwindigkeit erreichen können.

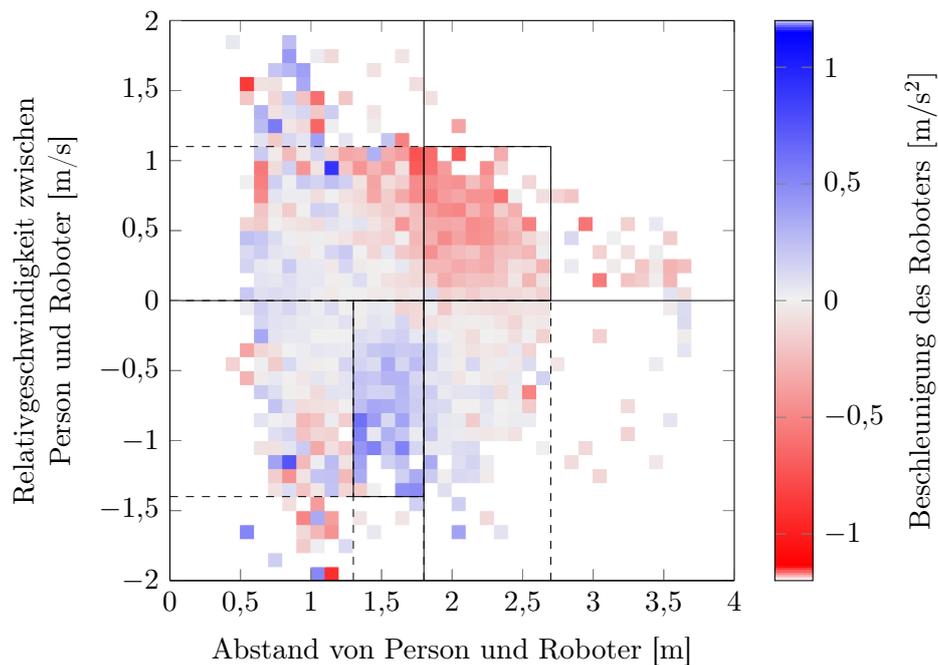


Abbildung 7.11. Vorsprung, Relativgeschwindigkeit und Beschleunigung beim Lotsen: Der Zusammenhang zwischen Abstand, Geschwindigkeit und Beschleunigung zeigt, dass der Roboter bei einem Vorsprung über 1,8 m und einer positiven Relativgeschwindigkeit verzögert. (Übernommen aus einer im Rahmen dieser Arbeit betreuten Masterarbeit [Seidl, 2015] und angepasst)

7.4. Begleiten

Ziel des Begleitverhaltens ist, dass der Roboter in der Nähe seines Nutzers bleibt, während sich dieser z. B. im Rahmen des Eigentrainings fortbewegt. Der Nutzer sollte dabei möglichst wenig Rücksicht auf den Roboter nehmen müssen und sich ungehindert fortbewegen können.

Der Versuchsaufbau zum *Begleiten von Personen* ist in Abbildung 7.12 gezeigt. Die Probanden bewegten sich während jedes Durchlaufs von der Position A nach B bzw. umgekehrt. Die Ausgangsposition des Roboters ist C bzw. D in 1,75 m Entfernung zur Person. Sobald sich die Person 0,8 m von der Startposition entfernt hat, bekommt der Roboter die Aufgabe der Person zu folgen. Die Zielpositionen der Person sind auf dem Fußboden markiert, jedoch dem Roboter nicht bekannt. Eine Herausforderung des Versuchsaufbaus ist, dass sich die Person bereits in einem gewissen Abstand zum Roboter bewegt, wenn dieser mit der Pfadplanung beginnt. Darüber hinaus führt der Weg der Person um eine Ecke herum, so dass der Roboter die Person nicht mehr wahrnehmen kann, wenn er zu spät losfährt oder der Person zu langsam folgt. Die Probanden wurden angewiesen sich in

einer ihnen angenehmen Geschwindigkeit zu bewegen. Falls sie bemerken, dass der Roboter ihnen nicht mehr folgt, sollten sie umkehren, damit der Roboter den Begleitvorgang wieder aufnehmen kann.

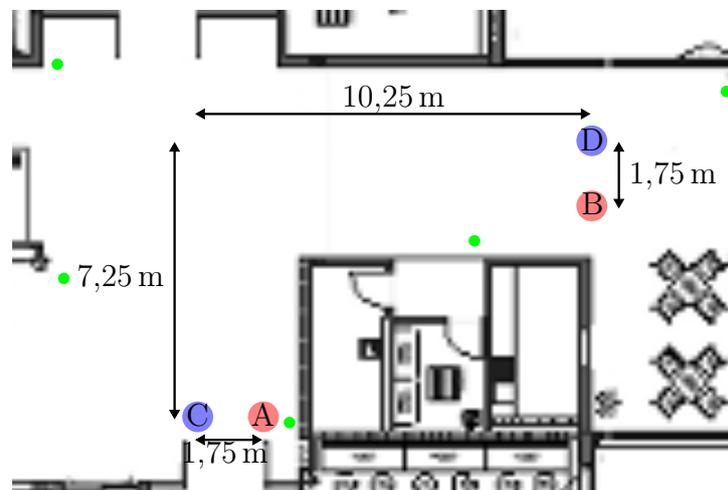


Abbildung 7.12. Versuchsaufbau zum Begleiten: Die Probanden bewegten sich zwischen den Positionen A und B hin und her. Die Ausgangsposition des Roboters sind entsprechend C und D. Sobald sich der Roboter der Zielposition der Person näher als 1,3 m angenähert hat, gilt der Durchlauf als abgeschlossen und der Roboter fährt zur Ausgangsposition für den nächsten Durchlauf. Die Trajektorien der Probanden und des Roboters wurden durch ein externes System (grün) getrackt.

Eine Übersicht über die erhobenen Daten ist in Tab. 7.5 gegeben. Die Durchschnittsgeschwindigkeit aller Probanden liegt mit 0,94 m/s nahe der Maximalgeschwindigkeit des Roboters von 1,0 m/s. Außerdem ist erkennbar, dass sowohl der Roboter als auch der Proband durchschnittlich bei jedem Durchlauf eine Wegstrecke von 12,8 m zurücklegen. Da der Roboter erst die Aufgabe bekommt, der Person zu folgen, wenn diese bereits losgelaufen ist, und aufgrund der begrenzten Maximalgeschwindigkeit ergibt sich selbst bei idealem Begleitverhalten ein Rückstand des Roboters. In der Tabelle sind die tatsächlichen Rückstände des Roboters zu dem Zeitpunkt, wenn die Person das Ziel erreicht, aufgeführt.

Durchschnittsgeschwindigkeit des Probanden und des Roboters Wie in Tabelle 7.5 ersichtlich, steigt der Rückstand des Roboters, wenn die Person das Ziel erreicht hat, linear mit der Durchschnittsgeschwindigkeit des Probanden (empirischer Korrelationskoeffizient 0.98). Dies könnte darauf hindeuten, dass der Roboter seine Geschwindigkeit nicht an den Probanden anpasst. Deshalb ist in Abb. 7.13 der Zusammenhang zwischen der durchschnittlichen Geschwindigkeit des Probanden und des Roboters während der

Tabelle 7.5. Datensatz zum Folgen: Für alle Probanden ist die Anzahl und die Durchschnittsgeschwindigkeit der Durchläufe aufgeführt. Außerdem ist die durchschnittliche Wegstrecke des Probanden selbst und des Roboters, sowie der durchschnittliche Rückstand des Roboters, wenn die Person das Ziel erreicht hat, aufgelistet. Die Daten sind nach der Durchschnittsgeschwindigkeit der Probanden sortiert. Die letzte Zeile zeigt die Summe aller Durchläufe und die Durchschnittswerte über alle Durchläufe.

Proband	Durchläufe	Geschwindigkeit Proband [m/s]	Wegstrecke [m]		Rückstand am Ziel [m]
			Proband	Roboter	
6	10	$\bar{0},78 \pm 0,08$	$\bar{14},5 \pm 5,16$	$\bar{14},0 \pm 4,02$	$\bar{3},2 \pm 0,47$
5	9	$\bar{0},87 \pm 0,14$	$\bar{12},8 \pm 0,66$	$\bar{12},2 \pm 0,30$	$\bar{3},6 \pm 0,87$
1	27	$\bar{0},91 \pm 0,05$	$\bar{12},9 \pm 0,29$	$\bar{12},4 \pm 0,30$	$\bar{3},8 \pm 0,25$
3	13	$\bar{0},95 \pm 0,86$	$\bar{13},0 \pm 2,16$	$\bar{13},5 \pm 3,72$	$\bar{4},3 \pm 0,47$
4	8	$\bar{1},04 \pm 0,07$	$\bar{11},8 \pm 0,44$	$\bar{12},4 \pm 0,47$	$\bar{4},7 \pm 0,65$
2	13	$\bar{1},07 \pm 0,11$	$\bar{12},0 \pm 0,59$	$\bar{12},5 \pm 0,97$	$\bar{5},3 \pm 1,01$
	80	$\bar{0},94 \pm 0,12$	$\bar{12},8 \pm 2,03$	$\bar{12},8 \pm 2,06$	$\bar{4},1 \pm 0,88$

einzelnen Durchläufe gezeigt. Dabei wird deutlich, dass es einen Zusammenhang zwischen den Geschwindigkeiten gibt (empirischer Korrelationskoeffizient 0,85), die Durchschnittsgeschwindigkeit des Roboters jedoch weniger stark steigt als die durchschnittliche Personengeschwindigkeit. Dies lässt sich damit erklären, dass die Probanden zu Beginn der Durchläufe unbewusst langsam laufen, damit der Roboter ihnen folgen kann. Während jedes Durchlaufs steigern sie jedoch die Geschwindigkeit, wenn sie merken, dass der Roboter robust folgt. Während der Anfangsphase der einzelnen Durchläufe kann der Roboter der Person dicht folgen und dessen Geschwindigkeit hängt von der Geschwindigkeit der Person ab. In der Endphase laufen die Probanden jedoch so schnell, dass der Roboter selbst mit maximaler Geschwindigkeit von 1,0 m/s hinter den Probanden zurückfällt.

Geschwindigkeit und Vorsprung der Person Um die Ursachen für den Rückstand des Roboters genauer zu untersuchen, ist in Abb. 7.14 der Zusammenhang zwischen der Momentangeschwindigkeit der Personen und dem zeitgleich gemessenen Rückstand des Roboters gezeigt. Die Häufigkeitsverteilung zeigt zwei deutliche Modi. Während des eigentlichen Begleitvorgangs läuft die Person meist mit einer Geschwindigkeit von ca. 1 m/s, was der Maximalgeschwindigkeit des Roboters entspricht, in einem Abstand von 4 m vor dem Roboter. Dies deutet darauf hin, dass die Person ihre Laufgeschwindigkeit an den Roboter anpasst, um sich nicht weiter als 4 m von dem Roboter zu entfernen. Der zweite Modus entsteht, wenn die Person das Ziel erreicht hat, stehen bleibt und sich der Roboter der Person auf ca. 1 m annähert.

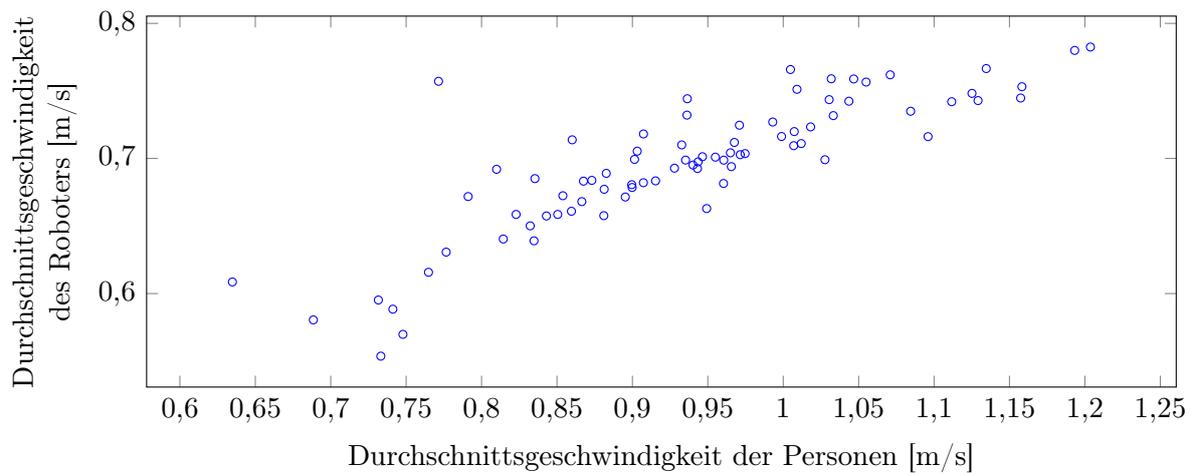


Abbildung 7.13. Durchschnittsgeschwindigkeit des Probanden und des Roboters: Die Durchschnittsgeschwindigkeiten des Probanden und des Roboters bei den einzelnen Durchläufen unterscheiden sich deutlich, zeigen jedoch eine starke Korrelation (Empirischer Korrelationskoeffizient 0,85). (Übernommen aus einer im Rahmen dieser Arbeit betreuten Masterarbeit [Seidl, 2015] und angepasst)

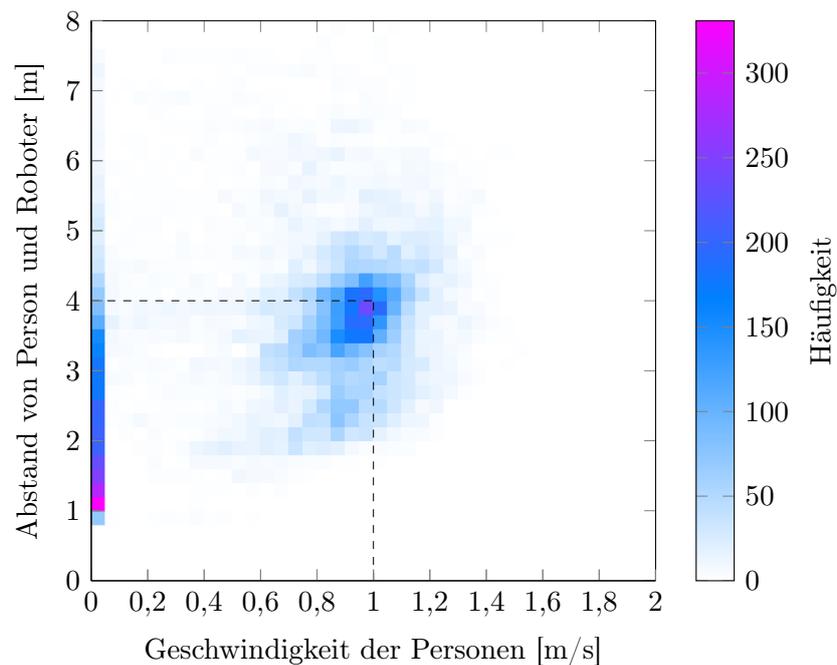


Abbildung 7.14. Geschwindigkeit und Vorsprung der Probanden: Die Häufigkeitsverteilung über der Geschwindigkeit der Person und dem Abstand zwischen Person und Roboter zeigt zwei Modi, welche dem Stillstand der Person am Ziel und dem eigentlichen Begleitvorgang entsprechen. Während des Begleitvorgangs bewegen sich die Probanden meist mit der Maximalgeschwindigkeit des Roboters (1 m/s) und dieser folgt den Probanden in durchschnittlich 4 m Abstand. (Übernommen aus einer im Rahmen dieser Arbeit betreuten Masterarbeit [Seidl, 2015] und angepasst)

Fazit zum Begleiten Die Versuche haben gezeigt, dass der Roboter mit dem vorgestellten Begleitverhalten in der Lage ist, einer Person, welche sich mit 1 m/s bewegt, in 4 m Abstand zu folgen. Der Task, die Person zu begleiten, wurde dem Roboter dabei erst gegeben, wenn sich die Probanden bereits in ca. 1,9 m Entfernung zum Roboter Richtung Ziel bewegt hatten. Der Rückstand und die Geschwindigkeit werden jedoch maßgeblich durch die Probanden bestimmt, da diese merken, dass der Roboter nicht schneller als 1 m/s fahren kann und ihre Geschwindigkeit auf dessen Maximalgeschwindigkeit reduzieren, damit der Roboter nicht wesentlich weiter als 4 m zurückfällt. Verlassen die Personen den Erfassungsbereich des Roboters, weil sie um eine Ecke abbiegen, wird der Aufenthaltsort der Person durch den Tracker für wenige Sekunden prädiziert und der Roboter navigiert zu dieser Position. Auf diese Weise konnte der Roboter während des Experiments auch Personen begleiten, welche in größerem Abstand um eine Ecke abbogen. In 4 der 80 Experimente (5%) kam es jedoch vor, dass der Roboter die Person nicht korrekt verfolgte und der Proband umkehren musste, um den Roboter abzuholen. Der Grund dafür war, dass die Person den Erfassungsbereich des Roboters verlassen hatte und der Roboter danach eine andere, zufällig anwesende Person begleitete. Diese Fehler sollten sich durch die Integration eines Verfahrens zur Wiedererkennung [Eisenbach, 2015] beheben lassen.

7.5. Zusammenfassung und Ausblick

Zur Evaluation der Sozialverträglichkeit des Roboters wurden vorrangig die Auswirkungen des Roboterhaltens auf die Fortbewegung eines menschlichen Interaktionspartners bewertet. Durch die Anwendung des *respektvollen Navigationsverhaltens* während des gegenseitigen Ausweichens, konnte die Beeinflussung der Person signifikant reduziert werden. Dies zeigte sich besonders an der Reduzierung der Krümmung der menschlichen Trajektorie und daran, dass die Personen beim Ausweichen des Roboters weniger stark abbremsen mussten. So konnten die Krümmungen der menschlichen Trajektorien im Vergleich zum „rücksichtslosen“ Roboterhalten ohne Wahrung des Personal Space von durchschnittlich $\bar{\kappa} = 0,12 \frac{1}{\text{m}}$ auf $\bar{\kappa} = 0,07 \frac{1}{\text{m}}$ reduziert werden. Außerdem reduzierten die Personen ihre Geschwindigkeit in der Nähe des Roboters nicht mehr um $\Delta v = 0,89 \text{ m/s}$, sondern nur noch um $\Delta v = 0,51 \text{ m/s}$. Im Rahmen der *respektvollen Navigation* wurde auch untersucht, wie sich das Rechtsfahrverhalten auf den Freiraum auswirkt, welcher entgegenkommenden Personen zur Verfügung steht, bevor der Roboter mit dem Ausweichverhalten beginnt.

Durch das Rechtsfahrverhalten konnte der linksseitige Abstand der sich entgegenkommenden Trajektorien des Roboters im Einsatzgebiet Rehaklinik von durchschnittlich $-0,01 \text{ m}$

auf 0,50 m gesteigert werden. Dadurch erhöhte sich jedoch auch die Fahrzeit des Roboters um ca. 4.7%, weshalb die Ausprägung des Rechtsfahrverhaltens je nach Einsatzumgebung und Zielanwendung parametrisiert werden sollte.

Beim *Lotsenverhalten* geht es darum, dass der Roboter dem Nutzer einerseits nicht zu weit vorausfährt und andererseits die Laufgeschwindigkeit des Nutzers möglichst wenig einschränkt. Die Experimente haben gezeigt, dass sich Personen, welche zügig laufen wollen, mit 0,9 m/s fortbewegen können, was nur geringfügig unterhalb der Maximalgeschwindigkeit des Roboters von 1,0 m/s liegt. Bremsst die Person unvermittelt aus dieser Geschwindigkeit ab, kommt der Roboter ca. 2,2 m bis 2,4 m vor der Person zum Stehen.

Beim *Begleiten von Personen* bewegten sich die Probanden meist mit 1 m/s, während ihnen der Roboter in ca. 4 m Abstand folgte. Dies zeigt offensichtlich, dass der Roboter in der Lage ist einer Person zu folgen, selbst wenn sich die Person mit der Maximalgeschwindigkeit des Roboters bewegt. Der Abstand wird jedoch maßgeblich durch die Probanden beeinflusst. Diese merken, dass der Roboter nicht schneller als 1 m/s fahren kann und passen dementsprechend ihre Geschwindigkeit an, um dem Roboter nicht weiter als 4 m voraus zu laufen. Daraus lässt sich nicht ableiten bis zu welchem Abstand der Roboter einer Person folgen könnte. Entscheidend ist jedoch, dass das Begleitverhalten durch die Probanden so robust wahrgenommen wird, dass diese 4 m vorweg laufen, ohne zu befürchten, dass der Roboter den Kontakt verliert.

Bei 5% der Begleitvorgänge konnte der Roboter den Probanden nicht durchgängig tracken und begleitete danach eine andere zufällig anwesende Person, so dass der eigentliche Proband umkehren musste, damit der Begleitvorgang wieder aufgenommen werden konnte. Um dieser Problematik zu begegnen ist eine Wiedererkennung des Nutzers erforderlich. Bei der zukünftigen Arbeit im Anwendungsszenario Rehabilitationsroboter ist deshalb die Integration der Wiedererkennung [Eisenbach, 2015] von wesentlicher Bedeutung. Außerdem steht noch eine detaillierte Evaluation des Roboterverhaltens zur Kontaktaufnahme mit sitzenden Patienten in natürlichen Situationen aus. Bisher wurden dazu nur Simulationen durchgeführt bzw. Situationen durch Mitarbeiter nachgestellt.

8

Zusammenfassung und Ausblick

Es folgt eine Zusammenfassung der vorangegangenen Kapitel dieser Arbeit, wobei deren wesentlichen Beiträge und Ergebnisse hervorgehoben werden. Zuletzt werden noch Ideen zur Fortsetzung dieser Arbeit erläutert.

8.1. Zusammenfassung

Diese Arbeit behandelte den sozialen Aspekt der Navigation mobiler Serviceroboter bei der Interaktion mit Personen in öffentlichen Einsatzumgebungen.

Als Anwendungsszenarien dieser Arbeit dienten ein Lotsenroboter, welcher als Besucherinformationssystem in einem Universitätsgebäude fungiert, und ein Rehabilitationsassistent zur Unterstützung von Schlaganfallpatienten beim Eigentraining. Ausgehend von diesen Szenarien wurden vier Verhaltensweisen (*Gezielte Kontaktaufnahme*, *Personenfokussiertes Lotsen*, *Begleiten einer Person*, *Respektvolle Navigation*) bestimmt, durch deren Realisierung die soziale Akzeptanz des Roboters gesteigert werden sollte.

Für die Realisierung der Verhaltensweisen galt insbesondere die Bedingung, dass keine Veränderungen an den Einsatzumgebungen oder der Experimentierplattform notwendig sein durften. Daraus leitete sich auch der Forschungsbedarf dieser Arbeit ab, denn die Aufarbeitung des Forschungsstandes bzgl. der Verhaltensweisen (Kapitel 2) zeigte, dass die sozialen Aspekte der Navigation bislang vorrangig in Laborstudien erforscht wurden. Bei Robotersystemen in realen Anwendungen und ohne Zuhilfenahme stationärer externer Sensorik waren diese Verhaltensweisen bisher meist von untergeordneter Bedeutung.

Ziel dieser Arbeit war daher die Realisierung der Verhaltensweisen in einem on-board Gesamtsystem für reale Einsatzumgebungen. Die Systemarchitektur des Gesamtsystems wurde im 2. Kapitel vorgestellt.

Die Beschreibung der Navigationsalgorithmen, welche auf reaktiver und taktischer Navigationsebene die Verhaltensweisen realisieren, erfolgte im 3. Kapitel. Besonders wichtig war dabei, dass die Unsicherheiten des Umweltmodells toleriert werden. Insbesondere die Informationsgewinnung über die Personen in der Nähe des Roboters sind für die sozialverträgliche Navigation relevant und waren deshalb wesentlicher Bestandteil dieser Arbeit.

So wurden im 4. Kapitel Verfahren zur laser- und kamerabasierten Personendetektion vorgestellt. Für den laserbasierten Ansatz wurden neuartige Merkmale entwickelt, welche nicht nur segmentierungsunabhängig und entfernungsinvariant sind, sondern zusätzlich unspezifisch bzgl. der zu detektierenden Objekte. Dies war insbesondere für das Anwendungsszenario des Rehabilitationsroboters relevant, da dadurch auch Personen, welche Gehhilfen mit sich führen, detektiert werden können und sogar eine Unterscheidung der Gehhilfen ermöglicht wurde. Zusätzlich zeichnet sich der Detektor mit den neuartigen Merkmalen gegenüber dem Referenzverfahren durch einen wesentlich geringeren Rechenaufwand bei deutlich verbesserter Detektionsgüte aus.

Bei dem kamerabasierten Detektor lag der Fokus auf der Klassifikation. Durch die datengetriebene Kombination von mehreren linearen Support Vector Machines in einem Entscheidungsbaum (SVM Tree) wurde zum einen die Klassifikationsleistung gegenüber einer einzelnen linearen SVM erhöht. Zum anderen wurde eine effiziente Mehrklassenunterscheidung ermöglicht, so dass neben der reinen Personendetektion auch die Unterscheidung von acht Orientierungen des Oberkörpers möglich ist. Die Leistungsfähigkeit und Effizienz des SVM Tree konnte insbesondere für hochdimensionale Merkmalsräume, wie sie bei der gradientenbasierten Objektdetektion gebräuchlich sind, im Vergleich mit anderen Klassifikatoren nachgewiesen werden.

Im 5. Kapitel wurde ein kamerabasiertes Verfahren zur erscheinungsbasierten Schätzung der menschlichen Oberkörperpose mit den Gelenkstellungen der Arme und der Kopfes untersucht. Dadurch konnte die Genauigkeit der Personenhypothesen, welche auf Basis der zuvor beschriebenen Detektoren getrackt wurden, gesteigert werden. Wichtiger war, dass dadurch auch die Wiedererkennung von Personen ermöglicht wurde. Allerdings wird dafür eine Grafikkarte benötigt, welche frei programmierbare Shader unterstützt. Während der experimentellen Untersuchung dieses Verfahrens mussten die Kamerabilder dazu auf einen externen PC übertragen werden.

Für die Bewegungsplanung des Roboters ist auch eine Prädiktion der Personenbewegung notwendig. Insbesondere, wenn sich eine Person und der Roboter gegenseitig ausweichen, ist die Bewegung der Personen stark durch die Präsenz des Roboters beeinflusst. Deshalb wurde in Kapitel 6 ein on-line lernendes Verfahren vorgestellt, mittels welchem die tatsächliche Personenbewegung gegenüber einem spezifischen Roboter gelernt und prädiziert werden kann. Der Neuheitswert dieses Verfahrens besteht darin, dass das Training nicht auf Daten der Mensch-Mensch-Interaktion erfolgt, sondern beim on-line Training selbst die Auswirkungen des Roboterverhaltens und die Gewöhnung der Menschen an den Roboter einfließen. Die Experimente haben gezeigt, dass verschiedene umgebungsabhängige Ausweichstrategien der Probanden gelernt werden können. Werden die Bewegungsprädiktionen der Personen beim gegenseitigen Ausweichen durch den Roboter berücksichtigt, reduzierte sich mit dem Training der Bewegungsprädiktion auch der Umweg der Probanden deutlich.

Zur Evaluation der bearbeiteten Verhaltensweisen in Kapitel 7 wurden vor allem die menschlichen Trajektorien während der Mensch-Roboter-Interaktion bewertet. In diese Bewertung flossen sowohl die Funktionalitäten zur Informationsgewinnung als auch die implementierten Navigationsalgorithmen ein. Gemessen wurden jedoch die tatsächlichen Auswirkungen auf die Fortbewegung der Personen, mit welchen der Roboter interagierte. Durch die *respektvolle Navigation* beim Ausweichen vor Personen konnten sich die Probanden zielgerichteter fortbewegen, ohne dem Roboter in engen Bögen ausweichen zu müssen oder die Geschwindigkeit zu reduzieren, damit der Roboter vorbei fahren kann. In dem Rehabilitationszentrum, wo weniger Freiraum zur Verfügung steht und sich die Patienten generell langsamer bewegen, wurde die Verletzung des Personal Space drastisch reduziert und das Roboterverhalten ist durch ein deutliches Rechtsfahrverhalten besser für die Patienten vorhersagbar. Auch beim *Lotsen* und *Begleiten von Personen* konnte gezeigt werden, dass sich Person und Roboter gemeinsam bis annähernd zur Maximalgeschwindigkeit des Roboters bewegen können, ohne dass sich der Roboter so weit von seinem Nutzer entfernt, dass der Kontakt verloren geht.

8.2. Ausblick

In den Kapiteln 4 bis 6 wurden zu den methodisch-technischen Funktionalitäten zur Informationsgewinnung über die anwesenden Personen auch Ansätze zu deren Verbesserung benannt. Ebenso erfolgte in Kapitel 7 eine kurze Beschreibung der nächsten Bearbeitungsschwerpunkte, welche sich im Rahmen des Projektes zum Rehabilitationsroboter

an diese Arbeit anschließen. Diese Ideen sollen hier noch einmal kurz aufgegriffen und komprimiert wiedergegeben werden.

In Abschnitt 4.1 wurden Merkmale zur laserbasierten Personendetektion und Unterscheidung von Gehhilfen vorgestellt, welche bzgl. Detektionsqualität und Effizienz überzeugen konnten. Obwohl das Verfahren auch zur Detektion von Personen innerhalb einer Gruppe eingesetzt wird, steht eine explizite Untersuchung der Detektionsqualität bei Personengruppen noch aus. So wäre einerseits die gezielte Untersuchung der Detektionsqualität bei Verdeckung durch andere Personen interessant. Andererseits ist zu untersuchen, ob bei nah beieinander stehenden Personen zusätzliche Hypothesen zwischen den Personen generiert werden.

Bei der kamerabasierten Personendetektion, welche in Abschnitt 4.2 vorgestellt wurde, ist die Generierung abwechslungsreicher Trainingsdaten mit der Greenscreen-Technik ein wesentlicher Bestandteil des Verfahrens. Während der Evaluation des Verfahrens stellte sich jedoch heraus, dass die Erfassung der Oberkörperorientierung durch das Skeletal Tracking der OpenNI™ [OpenNI, 2011] vereinzelt fehlerhaft war. Von einem geeigneterem Verfahren zur Erfassung der Oberkörperorientierungen und der Erzeugung eines hochwertigeren Trainingsdatensatzes wird auch eine bessere Klassifikation der Oberkörperorientierungen erwartet. Zur Erfassung der menschlichen Oberkörperorientierung könnte beispielsweise das zur Evaluation (Kapitel 7) verwendete System aus verteilten Laserscannern oder ein Motion-Capture-System verwendet werden.

Wie in Kapitel 5 beschrieben, wird zur Umsetzung des vorgestellten Verfahrens zur Schätzung von Gelenkstellungen und zur Wiedererkennung von Personen eine Grafikkarte mit programmierbaren Shadern benötigt. Da auf dem Roboter keine solche Grafikkarte zur Verfügung steht und voraussichtlich aufgrund des Energieverbrauchs auch nicht stehen wird, wurden die Berechnungen auf einem stationären PC durchgeführt und die Daten wurden per WLAN übertragen. Diese Lösung war jedoch nur bei den Laborexperimenten praktikabel und lässt sich insbesondere in der Rehaklinik nicht anwenden. Ein Verfahren zur Wiedererkennung ist jedoch unverzichtbar, denn spätestens wenn der Nutzer den Erfassungsbereich des Roboters verlässt, kann die Person nicht durchgängig getrackt werden. So haben die Experimente zum Begleitverhalten gezeigt, dass bei 5% der Begleitvorgänge eine andere, zufällig anwesende Person begleitet wurde, nachdem das Tracking des eigentlichen Nutzers zwischenzeitlich unterbrochen wurde, weil er um eine Ecke gebogen ist. Ein ähnliches Problem wurde bei Funktionstests in der Rehaklinik beobachtet, wenn der Nutzer zwischenzeitlich durch entgegenkommende Personen verdeckt wurde. Im weiteren Verlauf der Bearbeitung des Projektes zum Rehabilitationsroboter liegt deshalb ein Schwerpunkt auf der Entwicklung eines neuartigen Verfahrens zur ansichtsbasierten

Wiedererkennung des aktuellen Nutzers, mit dessen Integration bereits gegen Ende dieser Arbeit begonnen wurde [Eisenbach, 2015].

Das Verfahren zum gegenseitigen Ausweichen durch Personen und Roboter (Kapitel 6) nutzt die Bewegungsprädiktion jeder Personenhypothese in der Nähe des Roboters, damit der Roboter all diesen Personen möglichst gut ausweicht. Bei der Bewegungsprädiktion jeder einzelnen Person, wird jedoch vernachlässigt, dass sich die Personen auch untereinander in ihrer Bewegung beeinflussen. Es ist zu erwarten, dass sich die Güte der Bewegungsprädiktion mehrerer Personen, durch die Berücksichtigung deren räumlicher Relation und Geschwindigkeit zueinander, verbessern lässt. Dies erfordert eine geeignete Repräsentation der Relationen. Die Anzahl der Dimensionen des Zustandsraumes würde dieser Repräsentation entsprechend steigen. Darüber hinaus könnte das Clustering des Zustandsraumes weiterentwickelt werden. So sollte die Ausdehnung jedes einzelnen Clusters von den zugehörigen Aufenthaltswahrscheinlichkeiten, welche in den Belief Distribution Maps gespeichert sind, abhängen. Dadurch könnte vermieden werden, dass mehrere benachbarte Cluster auf ähnliche Aufenthaltswahrscheinlichkeiten verweisen, indem diese Cluster zusammengefasst würden. Umgekehrt könnten Cluster aufgeteilt werden, wenn sich abzeichnet, dass die entsprechenden Aufenthaltswahrscheinlichkeiten zu große Ähnlichkeit mit einer Gleichverteilung entwickeln und somit keine spezifische Prädiktion mehr möglich wäre.

In Kapitel 7 wurde anhand einzelner Funktionstests gezeigt, dass das Gesamtsystem die gewünschten Verhaltensweisen erfolgreich umsetzt. Abgesehen von der Evaluation des respektvollen Verhaltens in der Rehaklinik wurden die Interaktionen zwischen Mensch und Roboter dafür gezielt herbeigeführt. Als nächstes sollten deshalb Nutzertests folgen, welche beispielsweise in der Rehaklinik mit tatsächlichen Patienten durchgeführt werden. In diesem Kontext ist insbesondere die Evaluation der gezielten Kontaktaufnahme relevant. Dazu wurden bisher zwar erste Experimente durch Mitarbeiter des Fachgebiets durchgeführt, es konnte jedoch nicht abgeschätzt werden, ob die Interaktionsposition des Roboters auch für einen tatsächlichen Patienten angenehm gewesen wäre und ob die konstruierten Hindernissituationen mit einem Rollator und künstlichen Hindernissen realistisch waren.

Endgültige Aussagen über die Sozialverträglichkeit des Robotersystems werden jedoch erst durch echte Nutzertests unter Einsatz sozialwissenschaftlicher Evaluationsprinzipien möglich sein. So werden sich auch für die in dieser Arbeit behandelten Verhaltensweisen im Kontext der Gesamtapplikation andere Situationen ergeben bzw. werden bestimmte Situationen nicht auftreten. So wurde beispielsweise zur Evaluation der respektvollen Navigation im Rehabilitationszentrum ein bestimmtes Navigationsverhalten aktiviert, und

dieses Verhalten wurde von dem Roboter befolgt, bis er sein Ziel am anderen Ende der Etage erreicht hatte. In der Endanwendung soll jedoch eine Situationserkennung frühzeitig Verklemmungssituationen durch entgegenkommende Personen in schmalen Gangabschnitten detektieren [Gross, 2014]. Wird eine Verklemmungssituation erkannt, so soll das Verhalten des Roboters angepasst werden. Der Roboter soll vor dem entsprechenden Gangabschnitt warten und die Person durchlassen, bevor die ursprüngliche Aufgabe des Roboters fortgesetzt wird. Idealerweise sollte dadurch in der Endanwendung eine respektvolle Navigation gegenüber entgegenkommenden Personen in engen Gangabschnitten nicht nötig sein. Ebenso sind durch die Applikationstests auch neue Situationen und damit verbunden interessante Erkenntnisse zu den hier vorgestellten Verhaltensweisen zu erwarten.

Diese Arbeit behandelt lediglich ausgewählte Aspekte der sozialverträglichen Roboternavigation, und der Ausblick stellt nur die Aufgaben dar, welche sich unmittelbar an diese Arbeit anschließen. Dies verdeutlicht, dass in diesem spannenden Forschungsgebiet noch viele interessante Herausforderungen zu lösen sind. Die Fortschritte auf diesem Gebiet können mit Spannung erwartet werden, insbesondere da sie zukünftig in immer mehr Bereichen des alltäglichen Lebens zu beobachten sein werden.

A

Soziale Assistenz- roboter in öffentlichen Einsatzumgebungen

Die nachfolgende Tabelle gibt einen Überblick über Robotersysteme, welche einen Bezug zu den Anwendungsszenarien dieser Arbeit haben. Die Systeme werden bezüglich der in dieser Arbeit bearbeiteten Verhaltensweisen eingeordnet. Darüber hinaus werden Informationen zu den methodisch-technischen Funktionalitäten der einzelnen Systeme geliefert. Damit stellt diese Tabelle die Grundlage für die Motivation dieser Arbeit und deren Einordnung in den Forschungsstand (Abschnitt 2.5) dar.

Tabelle A.1. Prominente Robotersysteme mit Bezug zu dieser Arbeit: Für jedes System sind deren Einsatzfelder und technische Funktionalitäten im Kontext einer sozialverträglichen Navigation, sowie eine Bewertung bezüglich implementierter (■) sozial relevanter Verhaltensweisen gezeigt. (Basierend auf [Weinrich, 2011])

Verhaltensweise System	Gezielte Kontakt- aufnahme	Personen- fokussiertes Lotsen	Begleiten einer Person	Respektvolle Navigation
Robotersysteme in natürlichen Einsatzumgebungen				
RHINO & MINERVA: Museum Tour-Guide - 1997-2000 - [Burgard, 1998, Thrun, 1999b, Thrun, 1999a]				
Einsatz	6- bzw 13-tägiger Einsatz als Tour-Guide im Deutschen Museum in Bonn und im Smithonian in Washington; ohne technische Ausstattung der Einsatzumgebung; Interaktion mit mehr als 2.000 bzw. 10.000 Besuchern; 18,6 km bzw. 44,0 km zurückgelegter Weg			
Technische Funktionalität	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Kartenbasierte Kollisionsvermeidung mit statischen, bekannten Hindernissen; Detektion von dynamischen Hindernissen, keine explizite Detektion von Besuchern und daher keine respektvolle Navigation; Bewegung von Hindernissen nicht explizit berücksichtigt; Lotsen nur als Anfahren festgelegter Positionen ohne Berücksichtigung der gelotsten Person			
Nursebot Pearl: Assistenten für ältere Personen - 2001-2003 - [Pineau, 2003, Montemerlo, 2002]				
Einsatz	5-tägiger Einsatz von einem System in einem Altenheim bei Pittsburgh (USA); ohne technische Ausstattung der Einsatzumgebung; mehrere erfolgreiche Lotsenfahrten an den letzten 2 Tagen			
Technische Funktionalität	<input type="checkbox"/>	■	<input type="checkbox"/>	<input type="checkbox"/>
	Personenlokalisierung mittels Kartenvergleich, aber keine explizit respektvolle Navigation; Schätzung des Interesses des Nutzers erst während des Dialogs; Anpassung der Fahrgeschwindigkeit an Gehgeschwindigkeit der zu stützenden älteren Personen			

Verhaltensweise System	Gezielte Kontakt- aufnahme	Personen- fokussiertes Lotsen	Begleiten einer Person	Respektvolle Navigation
TOOMAS: Einkaufsassistent - 2001-2010 - [Gross, 2009, Müller, 2008, Müller, 2007, Pöschl, 2008, Pöschl, 2009, Pöschl, 2011]				
Einsatz	15-monatiger Einsatz mit 9 Robotern in drei Baumärkten in Deutschland; ohne technische Ausstattung der Einsatzumgebung; Interaktion mit mehr als 13.000 Baumarktkunden, 8.600 erfolgreiche Lotsenfahrten und 2.200 km zurückgelegter Weg; erster alltagstauglicher Shoppingroboter weltweit			
Technische Funktionalität	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Laser- und visonbasierte Navigation und Selbstlokalisierung ohne robotergerechte Umgestaltung des Einsatzfeldes; on-board Interessenschätzung auf Basis einer einfachen Gesichtsdetektion sowie der Bewegungsspuren potenzieller Nutzer in der lokalen Umgebung; Personentracking während der Lotsenfahrt zwecks Anhalten bei Kontaktverlust			
RoboX: Expo.02 Tour Guide - 2002 - [Siegwart, 2003, Jensen, 2005]				
Einsatz	Über 5-monatiger Einsatz von 11 Robotern in einem Pavillon während der Expo 2002 in der Schweiz; ohne technische Ausstattung der Einsatzumgebung; Interaktion mit 686.000 Besuchern, 3315 km zurückgelegter Weg			
Technische Funktionalität	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Personendetektion basierend auf Laserdaten und visueller Hautfarben- sowie Gesichtsdetektion; Hindernisvermeidung mittels Laser ohne besondere Behandlung von Personen; bei Hindernisvermeidung keine Berücksichtigung des „Personal Space“ oder der Bewegungsrichtung der Besucher; Kontaktaufnahme mit allen Besuchern in Roboternähe, ohne deren Interesse zu schätzen; Lotsen nur als Anfahren festgelegter Positionen			

Verhaltensweise System	Gezielte Kontakt- aufnahme	Personen- fokussiertes Lotsen	Begleiten einer Person	Respektvolle Navigation
Robovie: Einkaufsassistent - 2002-2009 - [Shiomi, 2009, Mutlu, 2009, Satake, 2009, Morales, 2014]				
Einsatz	Untersuchungen mit einem Roboter in einer Shoppingmall in Japan; technische Ausstattung der Einsatzumgebung (Einsatz von 6 stationären Laserscannern); Unterstützung der Roboter mittels Fernsteuerung und Telepräsenz durch menschlichen Operator mit unterschiedlichem Aufgabenspektrum; Interaktionen mit 53 Shoppingmallbesuchern [Satake, 2009]			
Technische Funktionalität	■	□	■	□
	Persondetektion in lokalem Einsatzfeld (20m x 5m) mittels verteilter, stationärer Laserscanner; Interessenschätzung anhand erkannter Bewegungsmuster (Anhalten, Gehen und Rennen); bei vermutetem Interesse autonome Anfahrt des Roboters an einen zur Kontaktaufnahme geeigneten „Abfangpunkt“ auf der antizipierten Bewegungstrajektorie der Person			
RoboCart: Einkaufsassistent für Sehbehinderte - 2005-2008 - [Gharpure, 2008]				
Einsatz	Untersuchungen in einem Supermarkt in Logan (USA); technische Ausstattung der Einsatzumgebung (RFID-Teppich) zwecks Vereinfachung der Navigation; konstante, niedrige Geschwindigkeit beim Lotsen			
Technische Funktionalität	□	□	□	□
	Navigation und Selbstlokalisierung mittels RFID und Laser; Hindernisvermeidung ohne besondere Behandlung von Personen			

Verhaltensweise System	Gezielte Kontakt- aufnahme	Personen- fokussiertes Lotsen	Begleiten einer Person	Respektvolle Navigation
InBOT(2): Einkaufsassistent - 2008-2011 - [Germa, 2009, Göller, 2010, Göller, 2009]				
Einsatz	Laborstudien im Rahmen des CommRob-Projektes mit Nachbildung eines Supermarktes; technische Ausstattung der Einsatzumgebung (RFID-Tagging)			
Technische Funktionalität	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	Personendetektion und -tracking mittels Gesichtsdetektion und RFID-Tagging; Verfolgung von Personen unter Berücksichtigung ihres „Personal Space“ mittels „Visual Servoing“; Berücksichtigung dynamischer Hindernisse und einfache vorausschauende Navigation; bislang nur Demonstration von Teilfunktionalitäten, keine vollständige Integration im Zielsystem			
Laborstudien				
Nomadic Scout II - [Butler, 2001]				
Einsatz	Laborstudie als häuslicher Assistent			
Technische Funktionalität	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Roboter und Testperson befinden sich initial auf vordefinierten Positionen; Roboter führt verschiedene feste Bewegungsmuster aus, welche zeitlich, durch Odometrie- und Sonardaten getriggert werden			
BIRON - 2004-2006 - [Hüttenrauch, 2006]				
Einsatz	Laboruntersuchung als häuslicher Assistent im Rahmen des FP6 Projektes COGNIRON			
technische Funktionalität	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Roboter wird von einer Person gesteuert (Wizard of Oz)			

Verhaltensweise System	Gezielte Kontakt- aufnahme	Personen- fokussiertes Lotsen	Begleiten einer Person	Respektvolle Navigation
Roboterplattform PeopleBot - [Pacchierotti, 2005, Walters, 2008, Syrdal, 2007]				
Einsatz	Laboruntersuchungen als häuslicher Assistent im Rahmen des FP6 Projektes COGNIRON			
Technische Funktionalität	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	In [Pacchierotti, 2005]: Laser und Sonar zur Personendetektion, Hindernisvermeidung und zum Ausweichen vor Personen unter Berücksichtigung des „Personal Space“; in den übrigen Veröffentlichungen navigiert der Roboter nicht oder wird von einer Person ferngesteuert (Wizard of Oz)			
Grace - [Gockley, 2007]				
Einsatz	Laborstudie zum Folgen einer Person			
Technische Funktionalität	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Personentracking basierend auf Laserdaten ohne Posenschätzung; Personenverfolgung mit bestimmtem Abstand zur Person mittels „Path Following“ bzw. „Direction Following“			

B

Kontrollparadigmen für Roboter

Zur Einordnung der verwendeten Steuerungsarchitektur (Abschnitt 2.6) bzgl. der populären Kontrollparadigmen werden diese kurz mit ihren anwendungsspezifischen Vor- und Nachteilen vorgestellt. Danach wird in Abschnitt B.2 die Zuordnung der verwendeten Architektur zum Paradigma Behavior-Based Control begründet.

B.1. Populäre Kontrollparadigmen für Roboter

Ein Kontrollparadigma ist definiert durch das Verhältnis zwischen den folgenden drei Primitiven: Wahrnehmung, Planung, und Aktion (Sense, Plan, Act). Darüber hinaus ist ein Kontrollparadigma bestimmt durch die Art, wie Sensordaten verarbeitet und im System verteilt werden [Murphy, 2000, S. 5]. Es werden vier populäre Paradigmen unterschieden, welche nachfolgend anhand dieser Kriterien beschrieben werden.

Hierarchisches Paradigma Bei dem rein *hierarchischen Paradigma*, welches auch als *deliberatives Paradigma* bezeichnet wird, werden Wahrnehmung, Planung und Aktion sequentiell abgearbeitet (Abbildung B.1a). Ein wesentlicher Nachteil davon ist, dass während der Wahrnehmungsphase das gesamte Umweltmodell aktualisiert wird und in der Planungsphase keine weiteren Sensorwahrnehmungen berücksichtigt werden. Erst wenn die Planungsphase abgeschlossen ist und die entsprechenden Aktionen durchgeführt wurden, beginnt wieder eine Wahrnehmungsphase. Dadurch findet die Reaktion auf neue Sensoreindrücke, unabhängig von der Komplexität der Reaktion, erst verzögert statt.

Reaktives Paradigma Im Gegensatz dazu sind die Aktionen bei dem *reaktiven Paradigma* direkt an die aktuellen Sensorwahrnehmungen gekoppelt, ohne dass zwischendurch eine Planung erfolgt (Abbildung B.1b). Diese unmittelbare Verbindung wird als Verhaltensweise bezeichnet. Die entsprechenden Systeme sind aus mehreren parallelen Verhaltensweisen aufgebaut, welche zusammen das Gesamtverhalten des Roboters bilden. Da die Wahrnehmungskomponenten verhaltensspezifisch sind und keine Planungskomponenten vorhanden sind, sind auch unmittelbare Reaktionen möglich. Eine Herausforderung dieses Paradigmas besteht in der Verrechnung der Aktionsvorschläge der einzelnen Verhaltensweisen. In diesem Zusammenhang werden beispielsweise die folgenden zwei Architekturen dem reaktiven Paradigma zugeordnet:

Subsumtion-Architektur: Die Verhaltensweisen sind hierarchisch in Kompetenzschichten angeordnet. Entsprechend setzen sich die komplexeren Verhaltensweisen der oberen Schichten bezüglich der Aktionsauswahl gegenüber den Basisverhalten der niedrigeren Schichten durch. Jede Verhaltensweise kann dabei auch auf die Aktionsvorschläge der niedrigeren Schichten zugreifen, so dass die ermittelte Aktion die Aktionen der niedrigen Schichten subsumiert. In Abhängigkeit von der aktuellen Aufgabe des Roboters können bestimmte Verhaltensweisen deaktiviert werden.

Potentialfeldmethoden: Die Verhaltensweisen liefern keine direkten Aktionen, sondern Vektorfelder, bei welchen jeder Vektor für den entsprechenden Punkt im Raum eine Kraft auf den Roboter repräsentiert. Diese Vektorfelder der einzelnen Verhaltensweisen werden in der Summe überlagert und ergeben so ein Gesamtvektorfeld. Die resultierende Aktion des Roboters ergibt sich aus dem Kraftvektor des Gesamtvektorfeldes an der aktuellen Roboterposition.

Der Nachteil des rein reaktiven Paradigmas ist, dass die Planung oder andere Funktionen, welche auf einer Repräsentation eines globalen Zustandes des Roboters aufbauen, nicht durchgeführt werden können. Das betrifft beispielsweise die Pfadplanung, Kartierung, Auswahl von Verhaltensweisen zur Erfüllung einer Aufgabe uvm.

Hybrides Deliberativ-Reaktives Paradigma Das hybride Paradigma soll die Nachteile des reaktiven Paradigmas gegenüber dem hierarchischen Paradigma kompensieren und gleichzeitig dessen Vorteile erhalten. Reaktive Steuerung und kognitive Funktion werden in diesem vereint, indem den reaktiven Verhaltensweisen eine deliberative Komponente hinzugefügt wird (Abbildung B.1c). Diese Komponente dient als virtueller Sensor für die reaktiven Verhaltensweisen. Außerdem können durch diese Komponente einzelne reaktive Verhaltensweisen aktiviert und deaktiviert werden. Dank multi-threading können die reaktiven Verhaltensweisen weiterhin schnell arbeiten, während die deliberativen/planenden

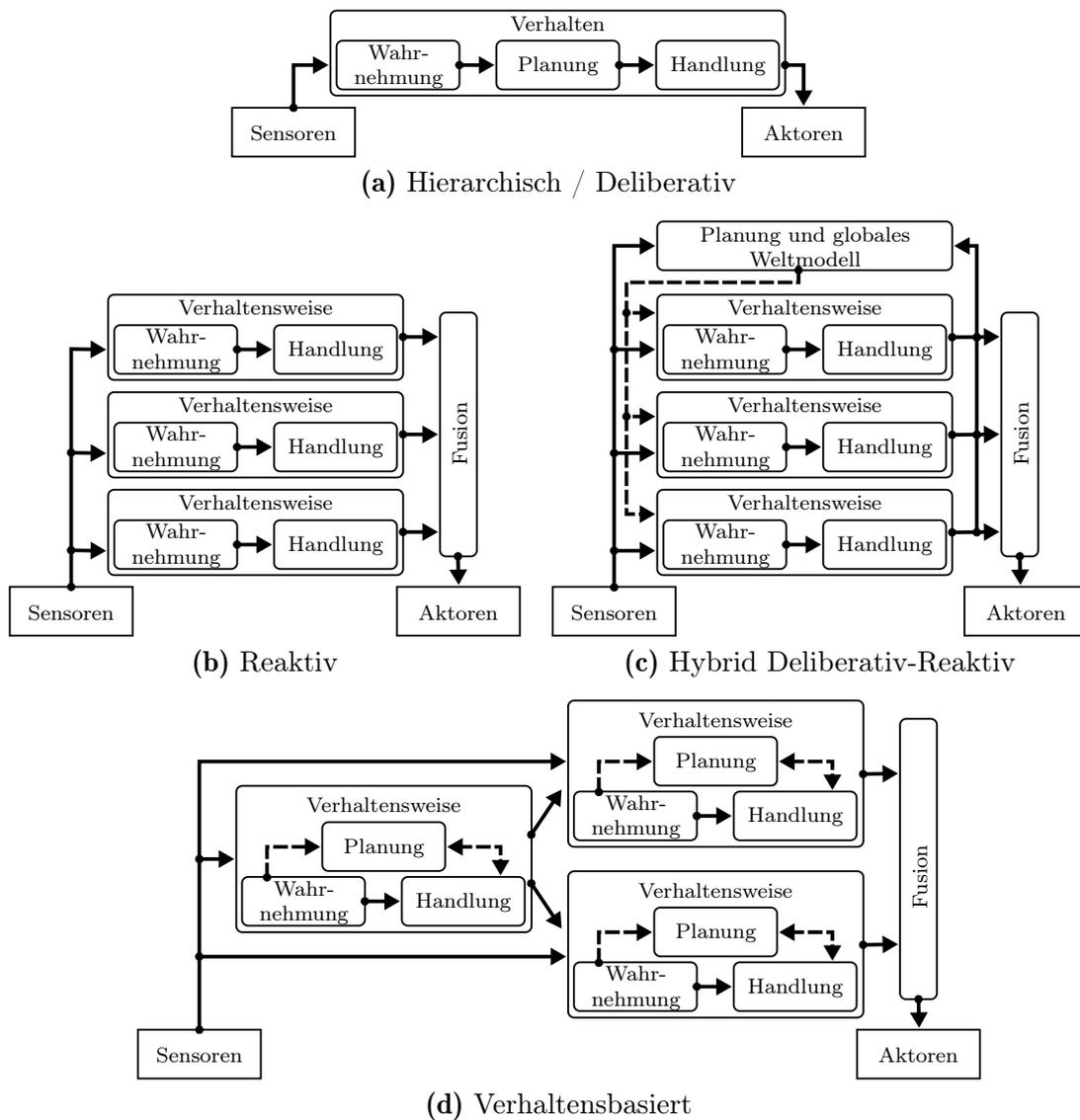


Abbildung B.1. Kontrollparadigmen: Schematische Darstellung der wesentlichen Komponenten und Datenflüsse von vier gebräuchlichen Kontrollparadigmen.

Module über einen längeren Zeitraum globale Ziele ermitteln. Die Planung erfolgt entkoppelt von der echtzeitfähigen Ausführung der reaktiven Verhaltensweisen.

In [Murphy, 2000, S. 264 ff.] werden drei verschiedene Arten unterschieden, wie der deliberative Teil der hybriden Architektur aufgebaut sein kann. Beim *Managementstil* (managerial style) werden die deliberativen Komponenten in hierarchische Schichten eingeteilt. Die Komponenten der oberen Schichten haben auf abstraktem Level einen großen Steuerungsumfang und dirigieren die deliberativen Komponenten in den niedrigeren Schichten, welche konkretere Kontrolle über die Teilbereiche haben. Weithin gibt es hybride Architekturen deren deliberative Komponenten in eine *Zustandshierarchie* (state hierarchy)

eingeordnet sind. Im Gegensatz zu den reaktiven Verhaltensweisen, welche keinen Zustand halten, werden deliberative Komponenten, welche Wissen über die Vergangenheit oder Wissen über die Zukunft speichern, unterschieden. Der dritte Stil hybrider Architekturen ist der *modellorientierte Stil* (model-oriented style), bei welchem die Verhaltensweisen danach eingeteilt werden, auf welche Teile des Weltmodells sie zugreifen.

Verhaltensbasierte Steuerung (Behavior-Based Control) Bei hybriden Systemen sind demnach die deliberativen und die reaktiven Komponenten voneinander getrennt. Diese strikte Unterteilung der einzelnen Komponenten gibt es bei der verhaltensbasierten Steuerung nicht. Stattdessen interagieren mehrere Module (Verhaltensweisen) in einem strukturierten Netzwerk und bilden gemeinsam das Verhalten des Roboters (Abbildung B.1d). Jede Verhaltensweise erhält Sensordaten oder Daten von anderen Verhaltensweisen und liefert Ausgaben an die Aktoren des Roboters bzw. an andere Verhaltensweisen. Im Gegensatz zum hybriden Paradigma gibt es kein zentrales Weltmodell. Stattdessen können die einzelnen Verhaltensweisen Zustandsinformationen speichern und so implizit ein verteiltes Modell der Umwelt bilden. Andererseits können die Verhaltensweisen auch rein reaktiv arbeiten und auf die Planungskomponente verzichten. Die Berechnung der einzelnen Verhaltensweisen erfolgt parallel, wodurch die Ausgabefrequenz von Verhaltensweisen mit geringen Rechenanforderungen nicht auf die Ausgabefrequenz der rechenaufwendigen Verhaltensweisen reduziert wird. Laut [Matarić, 2008, S. 896] unterscheiden sich das hybride und das verhaltensbasierte Paradigma nicht im Leistungsvermögen, sondern vielmehr in der Modularisierungsstrategie.

Die strukturelle Anordnung der einzelnen Verhaltensweisen kann sowohl homogen als auch hierarchisch sein. Bei der hierarchischen Struktur sind die Verhaltensweisen des verhaltensbasierten Paradigmas ähnlich den Komponenten des hybriden Paradigmas in Schichten angeordnet. Es werden auch Verhaltensweisen als zentrale Arbitrer (Lenker) zur Koordinierung der übrigen Verhaltensweisen eingesetzt. Der Arbitrer kann beispielsweise auf einem Zustandsautomat basieren.

B.2. Einordnung des Paradigmas dieser Arbeit

Die in dieser Arbeit verwendete Architektur (Abschnitt 2.6) ist dem Paradigma Behavior-Based Control zuzuordnen. Das wesentliche Kriterium für diese Zuordnung ist die im vorigen Absatz beschriebene Modularisierungsstrategie. Im Gegensatz zu einer hybriden Architektur wird in der verwendeten Architektur kein zentrales, sondern ein verteiltes Weltmodell verwendet. Alle Skills verwenden eine mehr oder weniger stark ausgeprägte

Zustands- und Planungskomponente. Die hierarchische Struktur der einzelnen Skills und die Verwendung einer zentralen Zustandsmaschine (Abb. 2.2), welche die übrigen Skills koordiniert, mag auch für die Zuordnung zum hybriden Paradigma sprechen. Allerdings kann beides auch Bestandteil der Behavior-Based Control sein, wohingegen ein verteiltes Weltmodell nicht Bestandteil des hybriden Paradigmas ist. In [Göller, 2014, S. 28 f.] wird die Einordnung bzgl. des hybriden und des verhaltensbasierten Paradigmas zusätzlich davon abhängig gemacht, ob der zentrale Arbiter seine Entscheidungen unabhängig von den übrigen Behaviors trifft. In dieser Arbeit hängen die Entscheidungen der Zustandsmaschine von den Behaviors der Behavior Layer ab, welche wiederum durch die Skills beeinflusst werden. Damit werden die Entscheidungen der Zustandsmaschine in Abhängigkeit von den übrigen Verhaltensweisen getroffen, und so kann die Zustandsmaschine als zentraler Arbiter der Behavior-Based-Control angesehen werden.

C

Umsetzung der Verhaltensweisen

C.1. Gezielte Kontaktaufnahme

Wie in Abschnitt 3.2 beschrieben, wird mittels Particle Swarm Optimization [Kennedy, 1995] eine geeignete Interaktionspose des Roboters gegenüber dem Interaktionspartner gesucht. Die Optimierungskriterien bei der Suche einer Interaktionspose sind (a) die Erreichbarkeit durch den Roboter, (b) der Abstand und (c) die Orientierung des Roboters relativ zur Person und (d) die Distanz zu Hindernissen. Darüber hinaus wird (e) eine Zielposition bevorzugt deren Abstand zur aktuellen Roboterposition möglichst gering ist. Um die Erfüllung des notwendigen Kriteriums, der (a) Erreichbarkeit der Pose durch den Roboter zu beurteilen, wird ein Pfadplanungsalgorithmus verwendet.

Bestimmung der Erreichbarkeit mittels Pfadplanungsalgorithmus Wie bereits in Abschnitt 3.1.2 beschrieben, bewertet die mittels E*-Algorithmus berechnete Navigationsfunktion M_{Nav} jede Gridzelle einer metrischen Karte der Einsatzumgebung bzgl. der Kosten, welche für die Navigation von der Zelle zu einem Ziel entstehen. Berechnungsgrundlage des Algorithmus' ist eine metrische Kostenkarte $M_{\text{travel_costs}}$, deren Zellen die Kosten zum Durchfahren des entsprechenden Gebiets in der Einsatzumgebung spezifizieren. Die berechneten Fahrtkosten sind unabhängig von der Fahrtrichtung, und so gibt die Navigationsfunktion auch die Fahrtkosten für die umgekehrte Fahrtrichtung von der Zielposition zu der jeweiligen Zelle an. Dies wird im Kontext der gezielten Kontaktaufnahme genutzt, um den Navigationsaufwand des Roboters zu beliebigen Zellen bestimmen zu können. Dazu wird die Navigationsfunktion ausgehend von der aktuellen Roboterposition berechnet, anstatt wie üblich eine Zielposition als Ausgangspunkt zu verwenden. Die

Navigationsfunktion liefert somit nun für jede Zelle die Kosten, um zu der entsprechenden Position zu fahren. Die Berechnung der Navigationsfunktion erfolgt ausgehend von der Ausgangsposition ähnlich einer sich ausbreitenden Wellenfront. Die lokale Ausbreitungsgeschwindigkeit der Wellenfront hängt von den Navigationskosten der durchquerten Zellen ab. Der Funktionswert $M_{\text{Nav}}(i, j)$ einer Zelle (i, j) entspricht der Zeit, die benötigt wird bis die Wellenfront die jeweilige Zelle erreicht hat. Die Zellen, welche aufgrund von Hindernissen nicht durch die Wellenfront erreicht werden können, behalten den Maximalwert c_{max} , welcher initial allen Zellen zugeordnet wird. Wenn der Algorithmus wie üblich zur Navigation zu einem bestimmten Ziel eingesetzt wird, kann dieser vorzeitig abgebrochen werden, wenn die Wellenfront den Roboter erreicht hat. Denn dann ist ein vollständiger Gradientenabstieg vom Roboter zu der Zielposition möglich. Im Kontext der Erreichbarkeitsbestimmung wird die Berechnung vorzeitig abgebrochen, sobald die Person erreicht wird, gegenüber welcher die Interaktionsposition eingenommen werden soll. Sollte die Person nicht durch die Navigationsfunktion erreichbar sein, so ist die Berechnung beendet sobald die Wellenfront überall Hindernisse erreicht hat.

Damit kann die von der Roboterposition ausgehende Navigationsfunktion genutzt werden, um für einen Partikel i zu ermitteln, ob die Partikelpose \mathbf{p}_i durch den Roboter erreichbar ist. Dieses Kriterium stellt schließlich die notwendige Bedingung der Kostenfunktion $f(\mathbf{p}_i)$ zur Bewertung des Partikels i dar:

$$f(\mathbf{p}_i) = \begin{cases} g(\mathbf{p}_i), & \text{wenn } M_{\text{Nav}}(\mathbf{p}_i) \neq c_{\text{max}} \\ c_{\text{max}}, & \text{sonst} \end{cases} \quad (\text{C.1})$$

Für die erreichbaren Roboterposen erfolgt die Bewertung der Partikel bzgl. der übrigen Optimierungskriterien (b-e):

$$g(\mathbf{p}) = \underbrace{\alpha_{\text{dist}} \cdot |\text{dist}(\text{pos}(\mathbf{h}_A), \text{pos}(\mathbf{p})) - d_{\text{int}}|}_{\text{(b) Abweichung von der idealen Interaktionsdistanz}} + \underbrace{\alpha_{\text{ori}} \cdot \frac{|\text{diff}(\text{ori}(\mathbf{p}), \text{atan2}(\text{pos}_y(\mathbf{h}_A) - \text{pos}_y(\mathbf{p}), \text{pos}_x(\mathbf{h}_A) - \text{pos}_x(\mathbf{p})))|}{\pi}}_{\text{(c) Abweichung von der idealen Displayorientierung zur Person}} + \underbrace{\alpha_{\text{tc}} \cdot M_{\text{travel_costs}}(\text{pos}(\mathbf{p}))}_{\text{(d) Nähe zu Hindernissen}} + \underbrace{\alpha_{\text{move}} \cdot \tanh(\text{dist}(\text{pos}(\mathbf{r}), \text{pos}(\mathbf{p})) - (\text{dist}(\text{pos}(\mathbf{r}), \text{pos}(\mathbf{h}_A)) - d_{\text{int}}))}_{\text{(e) Mehrweg gegenüber dichtester Interaktionsposition}} \quad (\text{C.2})$$

Der Mehrweg der Partikelposition gegenüber der dichtesten Position im Interaktionsradius der Person (e) fließt vor allem in die Optimierung ein, um die Stabilität der Interaktionspose zu erhöhen. Während sich der Roboter zu der Interaktionspose bewegt, wird die Particle Swarm Optimierung (PSO) wiederholt durchgeführt, um auf Veränderungen der Einsatzumgebungen reagieren zu können. Es soll jedoch verhindert werden, dass die optimierte Interaktionspose aufgrund von Sensorrauschen stark springt. Deshalb werden Interaktionsposen bevorzugt, welche bereits in der Nähe der aktuellen Roboterposition liegen.

C.2. Reaktives Ausweichen im Kontext der respektvollen Navigation

In Abschnitt 3.3.1 ist im Kontext des reaktiven Ausweichens die Formel 3.1 zur Bewertung der Verletzung des Personal Space für zukünftige Zeitpunkte Δt aufgeführt. Diese Formel basiert auf einer abstrakten Funktion $\text{cost}_{\text{PS}}^*(\mathbf{h}_i, t, \Delta t, \mathbf{v})$, welche die Nähe und Geschwindigkeit des Roboters gegenüber der zukünftigen Position einer Personenhypothese bewertet. Für die konkrete Realisierung dieser Funktion können verschiedene Prädiktionsmethoden verwendet werden. In Kapitel 6 wird dazu ein lernendes Prädiktionsverfahren mit der entsprechenden Kostenfunktion $\text{cost}_{\text{PS}}^{\text{bel}}(\mathbf{h}_i, t, \Delta t, \mathbf{v})$ beschrieben. Als Referenzmethode ist nachfolgend die Kostenfunktion $\text{cost}_{\text{PS}}^{\text{lin}}(\mathbf{h}_i, t, \Delta t, \mathbf{v})$ dargestellt, welche auf einer linearen Prädiktion basiert:

$$\text{cost}_{\text{PS}}^{\text{lin}}(\mathbf{h}_i, t, \Delta t, \mathbf{v}) = \overbrace{\text{cost}_{\text{PS}}^{\text{lin,d}}(\mathbf{h}_i, t, \Delta t, \text{pos}(\mathbf{r}, t, \Delta t, \mathbf{v}))}^{\text{distanzbasierte Kosten}} \cdot \left(1,0 + \alpha^{\text{app}} \cdot \overbrace{\max\left(0, \text{cost}_{\text{PS}}^{\text{lin,v}}(\mathbf{h}_i, t, \Delta t, \mathbf{v})\right)}^{\text{geschwindigkeitsbasierte Kosten}} \right) \quad (\text{C.3})$$

$$\text{cost}_{\text{PS}}^{\text{lin,d}}(\mathbf{h}_i, t, \Delta t, \mathbf{r}') = \exp\left(-\frac{1}{2} \left(\frac{|\text{dist}^{\text{lin}}(\mathbf{h}_i, t, \Delta t, \mathbf{r}')|}{\sigma}\right)^2\right) \quad (\text{C.4})$$

$$\text{cost}_{\text{PS}}^{\text{lin,v}}(\mathbf{h}_i, t, \Delta t, \mathbf{v}) = \overbrace{\frac{\text{dist}^{\text{lin}}(\mathbf{h}_i, t, \Delta t, \text{pos}(\mathbf{r}, t, \Delta t, \mathbf{v}))}{|\text{dist}^{\text{lin}}(\mathbf{h}_i, t, \Delta t, \text{pos}(\mathbf{r}, t, \Delta t, \mathbf{v}))|}}^{\text{Geschwindigkeit in Richtung Person}} \cdot v_{\text{trans}}(\mathbf{r}, t, \Delta t, \mathbf{v}) \quad (\text{C.5})$$

$$\text{dist}^{\text{lin}}(\mathbf{h}_i, t, \Delta t, \mathbf{r}') = \text{pos}(\mathbf{h}_i, t, \Delta t) - \mathbf{r}' \quad (\text{C.6})$$

Die Kostenfunktion $\text{cost}_{\text{PS}}^{\text{lin}}(\mathbf{h}_i, t, \Delta t, \mathbf{v})$ bewertet zum einen durch die Teilfunktion $\text{cost}_{\text{PS}}^{\text{lin,d}}$, wie nah sich der Roboter und die prädizierte Personenhypothese \mathbf{h}_i zum Zeitpunkt $t + \Delta t$ sind. Zum anderen fließt in die Bewertung mit $\text{cost}_{\text{PS}}^{\text{lin,v}}$ auch ein, wie schnell sich der Roboter auf die prädizierte Position der Personenhypothese zubewegt.

C.3. Taktisches Ausweichen im Kontext der respektvollen Navigation

In Abschnitt 3.1.2 wird die taktische Navigation zum Ausweichen stehender Personen beschrieben. Um den Personal Space der Personen in der Kostenkarte für den Planungsalgorithmus zu modellieren, wird eine Gauß-Funktion eingesetzt. In [Lu, 2013] wird beschrieben, dass diese Methode häufig sehr effektiv ist. Allerdings sei die Festlegung der Varianz σ der Gauß-Funktion, um ein bestimmtes Ausweichverhalten zu erreichen, schwierig. Da in dieser Arbeit der eigentliche Abstand, welcher gegenüber den Personen gewahrt wird, vorrangig durch ein reaktives Objektive bestimmt wird, ist hier eine genaue Parametrierung nicht erforderlich. Die Varianz sollte nur so groß sein, dass der Planungsalgorithmus keine Pfade zwischen einer Person und einem Hindernis in geringem Abstand hindurch plant, wenn auf der anderen Seite der Person bedeutend mehr Freiraum zum Ausweichen vor der Person zur Verfügung steht (Abb. 3.4c in Abschnitt 3.3.1).

In [Lu, 2013] wurde gezeigt, dass durch die Parametrierung von σ nur bestimmte Maximaldistanzen zur Person gewahrt werden können. Eine weitere Steigerung von σ würde dazu führen, dass direkt an der Person vorbeigefahren würde oder dass die Pfadlänge vernachlässigt würde und der Person mit maximalem Abstand ausgewichen würde.

Um beide Fälle zu vermeiden, ergibt sich ein maximaler Wert für die Varianz σ , welcher von dem Verhältnis $\frac{P}{A}$ der Kosten zum Befahren von Freiraum P und der Amplitude A der Kostenfunktion abhängt. Der in dieser Arbeit verwendete E*-Planungsalgorithmus [Philippsen, 2005] berechnet zum Durchfahren von 1 m Freiraum die Kosten $P = 1,0$ und Bereiche mit den Navigationskosten $\mathbf{M}_{\text{travel_costs}(x,y)} = 1,0$ werden nicht durchfahren. Entsprechend ist für die Amplitude der Funktion zur Beschreibung des persönlichen Raumes $A = 1,0$ gewählt. Die Simulationen in [Lu, 2013] haben ergeben, dass somit eine Varianz von $\sigma_{\text{max}} \approx 200$ gewählt werden kann und damit bei ausreichend Freiraum ein Abstand zur Person von $d_{\text{max}} \approx 20$ m erreicht würde. Würde ein $\sigma > \sigma_{\text{max}}$ gewählt, würde der Personal Space gar nicht mehr beachtet. Ein möglicher Abstand von $d_{\text{max}} \approx 20$ m ist jedoch weitaus mehr als notwendig, denn in dieser Arbeit wird eine Varianz $\sigma = 1,6 \ll \sigma_{\text{max}}$ verwendet.

C.4. Reaktives Einhalten des Rechtsverkehrs

In Abschnitt 3.3.3 wird das Objective zur Wahrung des Rechtsverkehrs beschrieben. Dieses lässt sich in zwei Teilalgorithmen zerlegen. Der eine Algorithmus (Alg. 2) bestimmt den Korrekturfaktor θ_{TWT} , welcher ausgehend vom ursprünglich geplanten Pfad $\mathbf{path}_{orig} = \{\mathbf{p}_0, \mathbf{p}_1, \dots\}$ einen angepassten Pfad \mathbf{path}_{adap} spezifiziert. Der angepasste Pfad verläuft rechtsseitig des ursprünglichen Pfades, wobei der Abstand des angepassten Pfades zum ursprünglichen Pfad entlang der Pfade linear steigt. Der Abstand ergibt sich aus dem Produkt von Korrekturfaktor θ_{TWT} und Abstand zum Roboter d_p (siehe Abb. 3.5). Dieser angepasste Pfad wird danach in einem weiteren Teilalgorithmus (Alg. 3) zur Bewertung der Geschwindigkeitskommandos \mathbf{v} verwendet.

Bestimmung des Korrekturfaktors Wie bereits in Abschnitt 3.3.3 beschrieben, wird zur Ermittlung des Korrekturfaktors der ursprünglich geplante Pfad \mathbf{path}_{orig} bis zu einer Distanz $d_{p,max}$ ausgewertet. Der betrachtete Pfadausschnitt wird gesampelt, und über die Pfadposition \mathbf{p}_i wird der maximal mögliche Korrekturfaktor θ_r bestimmt, welcher nicht dazu führt, dass der angepasste Pfad mit rechtsseitigen Hindernissen kollidiert. Ebenso wird der minimal notwendige Korrekturfaktor θ_l bestimmt, um den gewünschten linksseitigen Abstand zu Hindernissen einzuhalten. Bei ausreichend Freiraum ist der maximal mögliche Korrekturfaktor θ_r größer als der notwendige Korrekturfaktor θ_l und es kann $\theta_{TWT} = \theta_l$ angewendet werden, so dass der adaptierte Pfad den gewünschten linksseitigen Abstand zu Hindernissen wahrt. Bei den Gängen im Anwendungsszenario war jedoch meist nicht genug Freiraum vorhanden, und so wurde der angepasste Pfad durch den maximal möglichen Korrekturfaktor $\theta_{TWT} = \theta_r$ bestimmt.

Bewertung der Geschwindigkeitskommandos Nachdem der Korrekturfaktor bekannt ist, wird dieser zur Bewertung der Bewegungskommandos \mathbf{v} genutzt. Zu diesem Zweck wird die klothoidenförmige Trajektorie \mathbf{traj} entsprechend des Geschwindigkeitskommandos \mathbf{v} und des Prädiktionshorizontes ΔT bestimmt. Die letzten 20% dieser Trajektorie $\mathbf{traj} = \{\mathbf{tr}_0, \mathbf{tr}_1, \dots, \mathbf{tr}_{Tr}\}$ werden gesampelt und für jedes Sample wird der dichteste Punkt \mathbf{p}_i auf dem ursprünglich geplanten Pfad \mathbf{path}_{orig} ermittelt. Der rechtsseitigen Abstand d zwischen prädizierter Roboterposition \mathbf{tr}_j und entsprechender Pfadposition \mathbf{p}_i wird mit dem entsprechenden Abstand des angepassten Pfades zum ursprünglichen Pfad $d_p \cdot \theta_{TWT}$ verglichen. Der so ermittelte Fehler zwischen potentieller Robotertrajektorie und angepasstem Pfad wird über den Trajektorienabschnitt gemittelt und als Kosten $\text{cost}(\mathbf{v})$ des Geschwindigkeitskommandos \mathbf{v} ausgegeben.

```

Input: Pfad  $path_{orig} = \{\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_P\}$ , Navigationskosten  $\mathbf{M}_{travel\_costs}$ , gewünschter
linksseitiger Abstand  $d_{l,des}$ , minimaler beidseitiger Abstand  $d_{min}$ , betrachtete
Pfadlänge  $d_{p,max}$ 
Output: Korrekturfaktor  $\theta_{TWT} = 0,0$ 
1  $\theta_r = 1000,0$ ; // max. Korrektur, basierend auf rechtsseitigem  $d_{min}$ 
2  $\theta_l = -1000,0$ ; // notwendiger Korrekturfaktor entsprechend  $d_{l,des}$ 
3 for  $i \leftarrow 0$  to  $P - 1$  do
    // nur ein begrenzter Pfadabschnitt wird betrachtet
4   if  $dist(\mathbf{r}, \mathbf{p}_i) > d_{p,max}$  then
5     | break;
6   end
7    $\mathbf{o} \leftarrow$  Orthogonale zu  $\overline{\mathbf{p}_i \mathbf{p}_{i+1}}$ ;
8    $d_l \leftarrow$  Abstand zw.  $\mathbf{p}_i$  und nächstem Hindernis in  $\mathbf{M}_{travel\_costs}$  entlang  $\mathbf{o}$ ;
9    $d_r \leftarrow$  Abstand zw.  $\mathbf{p}_i$  und nächstem Hindernis in  $\mathbf{M}_{travel\_costs}$  entlang  $-\mathbf{o}$ ;
10   $d_p \leftarrow dist(\mathbf{p}_i, \mathbf{p}_0)$ ; // Abstand zwischen aktuellem Pfadsample und Roboter
11  if  $(d_r - d_{min}) / d_p < \theta_r$  then
12    |  $\theta_r = (d_r - d_{min}) / d_p$ ;
13  end
14  if  $(d_l - d_{l,des}) / d_p > \theta_l$  then
15    |  $\theta_l = (d_l - d_{l,des}) / d_p$ ;
16  end
17 end
18 if  $\theta_l < \theta_r$  then
    // Auf dem gesamten Abschnitt ist ausreichend Freiraum
    // Anpassung richtet sich nach linksseitigem Korrekturfaktor
19  |  $\theta_{TWT} = max(0, \theta_l)$ 
20 end
21 else
    // Freiraum reicht nicht um gewünschte Abstände einzuhalten
    // Es wird so weit wie möglich rechts gefahren
22  |  $\theta_{TWT} = max(0, \theta_r)$ 
23 end

```

Algorithmus 2 correctionFactor: Ermittlung des Faktors θ_{TWT} zur Pfadkorrektur

```

Input: Geschwindigkeitskommando  $\mathbf{v}$ , Prädiktionshorizont  $\Delta T$ , Pfad
          $\mathbf{path}_{orig} = \{\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_P\}$ , Korrekturfaktor  $\theta_{TWT}$ 
Output: Geschwindigkeitsbewertung  $cost(\mathbf{v})$ 
1  $\mathbf{traj} = \{\mathbf{tr}_0, \mathbf{tr}_1, \dots, \mathbf{tr}_{Tr}\} \leftarrow$  Prädizierte Trajektorie entsprechend  $\mathbf{v}, \Delta T$ ;
2  $error_{sum} \leftarrow 0, 0$ ;
3  $error_{count} \leftarrow 0$ ;
  // die letzten 20% der prädizierten Trajektorie werden bewertet
4 for  $j \leftarrow Tr \cdot 0,8$  to  $Tr - 1$  do
  | // Ermittlung des dichtesten Pfadsamples zum Trajektoriensample
5    $i \leftarrow \arg \min_{k=0 \dots P} dist(\mathbf{tr}_j, p_k)$ 
  | // Abstand zwischen dichtestem Pfadsample  $\mathbf{p}_i$  und Roboter
6    $d_p \leftarrow dist(\mathbf{p}_i, \mathbf{p}_0)$ ;
7    $\mathbf{o} \leftarrow$  Orthogonale zu  $\overline{\mathbf{p}_i \mathbf{p}_{i+1}}$ ;
8    $d \leftarrow$  Länge der Projektion von  $\overline{\mathbf{p}_i \mathbf{tr}_j}$  auf  $\mathbf{o}$ 
9    $error_{sum} += \max(0, d_p \cdot \theta_{TWT} - d)$ ;
10   $error_{count} += 1$ ;
11 end
  // Ermittlung der durchschnittlichen Abweichung
12  $error_{avg} \leftarrow error_{sum} / error_{count}$ ; // Normierung der durchschnittlichen
  | Abweichung mit maximal möglicher Abweichung
13  $cost(\mathbf{v}) \leftarrow error_{avg} / d_{l,des}$ ;

```

Algorithmus 3 voteTWT: Bewertung der Geschwindigkeitskommandos durch das Two-Way-Traffic Objective

D

Laser- und kamerabasierte Personendetektion

D.1. Parametrierung der laserbasierten Personendetektion

Während der Experimente zur laserbasierten Personendetektion (Abschnitt 4.1.7) wurde für alle Verfahren eine Jump Distance Segmentierung der Tiefendaten mit einem Schwellwert $\Delta = 0,1$ m angewendet. Die Basislinie l_j der GDIF Merkmale ist $w = 1,0$ m breit und die Tiefe des Extraktionsbereichs ist auf $d = 3,0$ m beschränkt. Die Basislinie ist in $n = 15$ Liniensegmente von ca. 6,7 cm aufgeteilt. Der Versatz der Basislinie zum ersten Messpunkt der Segmente beträgt $o = 6,7$ cm.

D.2. Mehrklassenunterscheidung mittels SVMs

Da in Abschnitt 4.2.1 die Eigenschaften verschiedener Methoden zur Mehrklassenunterscheidung mittels SVMs nur tabellarisch aufgeführt sind, werden diese in diesem Kapitel etwas ausführlicher erklärt.

Lineare SVMs Zur Klassifikation von HOG-Merkmalen werden in [Dalal, 2005, Wang, 2009] und vielen anderen Detektoren lineare Support Vector Machines (SVMs) eingesetzt. Bei linear separierbaren Problemen zeichnen sich SVMs durch gute Generalisierungseigenschaften aus. Der Berechnungsaufwand zur Klassifikation mittels einer SVM ist relativ gering, da sich die auf den Support Vectors basierende Klassifikation auf ein Skalarprodukt des D -dimensionalen Merkmalsvektors \mathbf{x} mit einem Gewichtsvektor \mathbf{w} und einen Vergleich mit einem Schwellwert b reduzieren lässt:

$$f(\mathbf{x}) = \begin{cases} 1 & \text{if } \sum_{d=1}^D \mathbf{w}_d \cdot \mathbf{x}_d > b \\ -1 & \text{else} \end{cases} \quad (\text{D.1})$$

Ein weiterer Vorteil der linearen SVMs ist die geringe Anzahl an Trainingsparametern. Ein Kostenfaktor reguliert den Kompromiss zwischen Generalisierungsfähigkeit und Präzision des Klassifikators, und ein Toleranzparameter fungiert als Abbruchkriterium. Optional kann auch noch der Kostenfaktor einzelner Klassen oder sogar einzelner Datenpunkte angepasst werden.

Mehrklassenunterscheidung Allerdings kann eine einzelne SVM nur zur binären Klassifikation eingesetzt werden. Es gibt jedoch verschiedene Methoden um mehrere Klassen mittels SVMs zu unterscheiden. Eine One-Versus-Rest (1-v-R) SVM [Vapnik, 1998] verwendet C binäre SVMs um jede der C Klassen von den verbleibenden Klassen zu unterscheiden. Die resultierende Klasse wird mittels Winner-Take-All Strategie ermittelt, wodurch der Klassifikator mit dem höchsten Ausgabewert die resultierende Klasse bestimmt.

Bei der One-Versus-One (1-v-1) Methode [Kreßel, 1999] werden $\frac{C^2-C}{2}$ binäre SVMs verwendet um jede der C Klassen von jeder anderen Klasse zu separieren. Die Max-Wins Strategie, bei welcher gezählt wird, wie oft die binären Klassifikatoren für jede einzelne Klasse stimmen, wird angewendet, um die resultierende Klasse zu bestimmen. Mittels 1-v-1 SVMs lassen sich tendenziell komplexere Klassifikationsprobleme lösen, da die Separation von einer Klasse und allen anderen Klassen zusammen häufig komplexer ist, als die Separation von dieser Klasse und jeder anderen Klasse einzeln.

Allerdings sind 1-v-1 SVMs auch rechenaufwändiger als 1-v-R SVMs. Deshalb bestehen Directed Acyclic Graph SVMs (DAGSVMs) [Platt, 2000] zwar auch aus $\frac{C^2-C}{2}$ binären SVMs, aber es werden zur Klassifikation eines einzelnen Samples nur $C - 1$ binäre Klassifikationen berechnet. Welche binäre SVMs angewendet werden ergibt sich entsprechend eines gerichteten azyklischen Graphens aus den Klassifikationsergebnissen der bereits berechneten binären SVMs.

Nichtlineare SVMs Die drei beschriebenen Methoden (1-v-1, 1-v-R, DAGMSVM) zur Unterscheidung mehrerer Klassen mittels binärer SVMs lassen sich sowohl auf die beschriebenen linearen SVMs als auch auf nichtlineare SVMs anwenden. Nichtlineare SVMs werden dann verwendet, wenn sich die Klassen nicht durch Hyperebenen separieren lassen. Allerdings sind nichtlineare SVMs berechnungsintensiver, denn sie lassen sich wegen der Verwendung von Kernelfunktionen nicht vereinfachen. So müssen alle k Support Vectors verarbeitet werden. Außerdem muss für jeden Support Vector $\mathbf{x}^{(i)}$ die Kernelfunktion $K(\mathbf{x}^{(i)}, \mathbf{x})$ berechnet werden:

$$f(\mathbf{x}) = \begin{cases} 1 & \text{if } \sum_{i=0}^k \alpha_i K(\mathbf{x}^{(i)}, \mathbf{x}) > b \\ -1 & \text{else} \end{cases} \quad (\text{D.2})$$

Die Verwendung einer Kernelfunktion erfordert auch deren Parametrierung. Für den häufig verwendeten Gauß-Kern muss eine Kovarianzmatrix definiert werden. Diese hat einen großen Einfluss auf das Klassifikationsergebnis, wodurch deren Parametrierung nicht trivial ist.

Außerdem wird zum Training nichtlinearer SVMs häufig sehr viel Arbeitsspeicher benötigt. Deshalb wird in [Chang, 2010] eine Decision Tree SVM (DTSVM) verwendet, bei welcher nichtlineare SVMs mit einem Entscheidungsbaum kombiniert werden. Im Wesentlichen dient der Entscheidungsbaum der Zerlegung des Merkmalsraumes bis die Unterräume effizient durch die SVMs in den Blättern des Baumes bewältigt werden können. Bei reinen Entscheidungsbäumen müsste diese Zerlegung so lange erfolgen, bis die Unterräume nur noch Samples einer einzigen Klasse enthalten, was häufig zu Überanpassung führt.

Entscheidungsbäume mit SVMs zur binären Entscheidung Grundsätzlich gibt es eine große Auswahl an Klassifikatoren zur Mehrklassenunterscheidung mit unterschiedlicher Leistungsfähigkeit und Rechenaufwand. Da verschiedene Veröffentlichungen bereits gezeigt haben, dass SVMs zur Klassifikation im HOG-Merkmalsraum geeignet sind, werden in dieser Arbeit Erweiterungen von linearen SVMs zur Mehrklassenunterscheidung untersucht, welche den Berechnungsaufwand möglichst wenig erhöhen. Deshalb werden Entscheidungsbäume untersucht, bei welchen, ähnlich den DAGSVMs, in den Knoten SVMs als binäre Entscheider eingesetzt werden und welche im folgenden als SVMBDT bezeichnet werden.

Eine Herausforderung bei der datengetriebenen Strukturierung von Entscheidungsbäumen zur Mehrklassenentscheidung ist die Abbildung der Klassenlabels auf zwei Grup-

penlabel, welche in einem Knoten voneinander getrennt werden sollen. In [Osman, 2007] wird initial für jede Klasse ein eigenes Gruppenlabel erzeugt. Danach werden iterativ die Gruppen, deren Datenschwerpunkte sich am nächsten sind, zusammengefasst bis nur noch zwei Gruppen übrig sind. Die Samples dieser Gruppen werden dann in dem Knoten voneinander separiert und entsprechend dieser Trennung an dessen zwei Kindknoten übergeben. In den Kindknoten wird dieser Prozess rekursiv wiederholt, bis alle Samples eines Knotens zur selben Klasse gehören.

Der Vorteil der SVMs als binäre Entscheider in den Knoten eines Entscheidungsbaumes gegenüber Entscheidungsbäumen mit einfachen Schwellwertentscheidungen [Quinlan, 1986] ist, dass für die Trennebenen zur Separation des Merkmalsraumes nicht die Einschränkung gilt, dass diese achsenparallel sind, sondern beliebig orientiert sein können und dadurch bessere Generalisierung möglich ist. Wie Abbildung 4.16 zeigt, können mittels SVMBDT konkave Datenverteilungen separiert und Mehrklassenentscheidungen getroffen werden. Die Anzahl der zur Klassifikation notwendigen binären SVMs, welche der Pfadlängen des Baumes entspricht, hängt von der Komplexität des Problems ab. Somit wird für Samples in leicht separierbaren Unterräumen des Merkmalsraumes weniger Rechenzeit benötigt, als für schwerer separierbare Samples.

D.3. Optimierung der SVM eines Knotens des Entscheidungsbaumes

Bei dem in Abschnitt 4.2.4 beschriebenen rekursiven Training eines Entscheidungsbaumes mit SVMs als binäre Entscheider gibt es verschiedene Parameter, welche anzupassen sind, um die binäre Entscheidung der SVM hinsichtlich des Gains (Gleichung 4.2) zu optimieren. Um möglichst gute Parameter zu finden, werden diese, wie in Abbildung D.1 dargestellt, rekursiv angepasst. Es wird immer die SVM, deren binäre Klassifikation zum höchsten Gain geführt hat, gespeichert und letztendlich als Ergebnis der in Algorithmus 1 verwendeten Funktion `getOptimizedSVM` verwendet. Im Folgenden wird vorgestellt, welche Parameter wie optimiert werden.

Binäre Gruppierung der Zielklassen Um eine SVM zu trainieren, müssen alle Trainingsamples einer binären Klasse zugeordnet werden. Vereinfachend werden in dieser Arbeit alle Samples einer Zielklasse c auf dieselbe binäre Klasse $b(c) \in \{-1, 1\}$ abgebildet. Somit gilt es, eine optimale Abbildung $\mathbf{m} = \{b(1), b(2), \dots, b(C)\} \in \{-1, 1\}^C$ der Zielklassen auf die binären Klassen zu finden. Initial wird nur die Zielklasse mit den

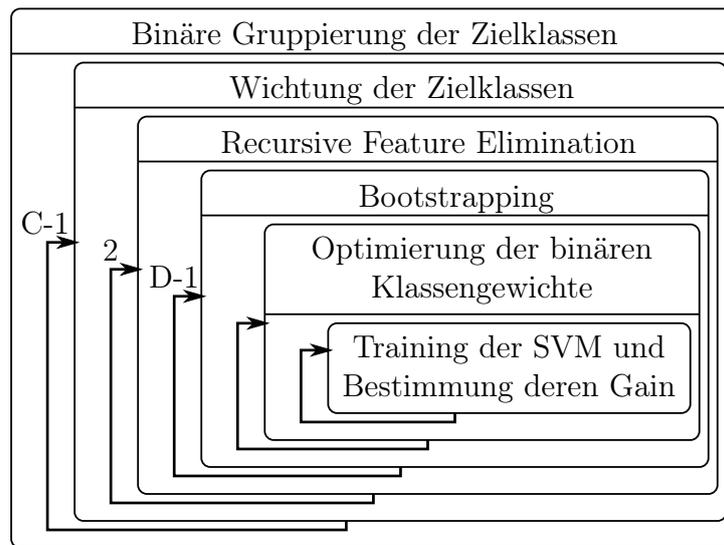


Abbildung D.1. Optimierungsschritte: Die Verarbeitungsschritte werden rekursiv durchgeführt um eine geeignete SVM zu finden. So wird beispielsweise für jede Gruppierung die Wichtung der Zielklassen durchgeführt etc.

meisten Samples $c' = \arg \max_c |S_c|$ auf das binäre Label 1 abgebildet und alle anderen Zielklassen auf das Label -1 . Anschließend wird m in jedem Optimierungsschritt iterativ modifiziert, indem eine der auf -1 abgebildeten Zielklassen c'' auf 1 abgebildet wird, bis nur noch eine einzige Zielklasse auf -1 abgebildet wird. Damit der Klassifikationsfehler der SVM möglichst gering ist, wird die Abbildung der Zielklasse $c'' = \arg \max_{\forall c, b(c)=-1} \frac{|S_c \cap S^1|}{|S_c|}$ geändert, von deren Samples der größte Anteil falsch klassifiziert wurde. Somit werden insgesamt $C - 1$ verschiedene Abbildungen untersucht. Nachdem die nachfolgenden Optimierungen für alle Abbildungen durchgeführt wurden, wird die SVM der Abbildung m verwendet, welche den höchsten Gain auf dem Validierungsdatensatz erreicht hat.

Wichtung der Zielklassen (Optional) Im Idealfall wären alle Samples bzgl. der binären Label separierbar. Meist ist dies aber nicht der Fall, und die SVM versucht die Trainingssamples möglichst gut bzgl. der binären Klassen zu separieren, ohne dass die Zielklassen berücksichtigt werden. Um einen hohen Gain zu erreichen, ist es jedoch wichtiger einzelne Zielklassen besonders gut zu separieren, selbst wenn dadurch der Fehler bzgl. der binären Klassifikation steigt (Abb. D.2). Um zu vermeiden, dass einzelne Zielklassen die Trennung der anderen Zielklassen negativ beeinflussen, wird von jeder binären Gruppe die Zielklasse mit dem geringsten Fehler bestimmt. Die Samples der übrigen Zielklassen werden abgewichtet oder gar nicht beim Training der SVM berücksichtigt. Nachdem die nachfolgenden Verarbeitungsschritte einmal mit und einmal ohne Abwichtung der Zielklassen durchgeführt wurden, wird der Gain der beiden resultierenden SVMs auf dem Validierungsdatensatz verglichen und die bessere SVM verwendet.

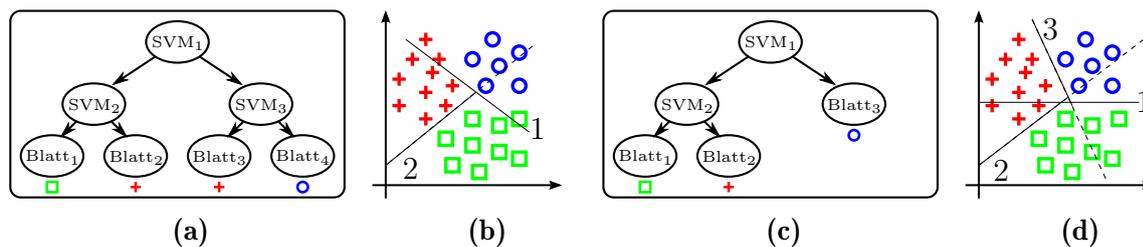


Abbildung D.2. Einfluss der Wichtung: Die Auswirkungen der Gewichtsreduktion der Zielklasse, welche durch + dargestellt wird, ist in (c,d) gezeigt. Im Gegensatz zu (a,b) ist der Margin von SVM_1 zwischen □ und o breiter geworden und somit wird eine bessere Generalisierung erreicht. Dafür wird jedoch auch eine weitere SVM zur Trennung von + und o benötigt.

Recursive Feature Elimination (Optional) Um eine Überanpassung der SVMs zu vermeiden, sollen irrelevante Merkmale aus dem Trainingsdatensatz entfernt werden. Dies ist bei linearen SVMs sehr leicht mittels Recursive Feature Elimination (RFE) möglich [Guyon, 2002]. Dazu werden alle Merkmale bzgl. ihres Einflusses auf die Größe des Margins bewertet. Danach wird eine bestimmte Anzahl der Merkmale, welche einen niedrigen Rang haben, eliminiert und die SVM wird erneut trainiert. Weitere Merkmale können iterativ entfernt werden. Bei einer linearen SVM ergibt sich der Rang $rank_d$ der Merkmalsdimension d ausgehend von Gleichung D.1 direkt aus dem Gewichtsvektor \mathbf{w} : $rank(d) = |\mathbf{w}_d|$. Da während der Eliminierung einzelner Merkmale immer wieder der Information Gain $G(SVM)$ (Gleichung 4.2) auf dem Validierungsdatensatz ermittelt wird, steigt die Generalisierungsfähigkeit so lange irrelevante Merkmale entfernt werden und sinkt wieder, wenn relevante Merkmale entfernt werden. Die Merkmalsselektion wird für jeden Knoten und jede Gruppierung durchgeführt, da je nach Klassifikation verschiedene Merkmale relevant sein können.

Bootstrapping Die notwendige Berechnungszeit für das Training der SVMs hängt von der Anzahl an Trainingssamples ab. Allerdings haben einige Samples einen sehr geringen Einfluss auf die resultierende SVM. Um zu verhindern, dass diese Samples den Berechnungsaufwand unnötig erhöhen, wird ein iteratives Bootstrapping angewendet. Initial wird eine suboptimale SVM auf einer Teilmenge der Trainingsdaten trainiert. Danach wird diese SVM verwendet, um eine weitere Teilmenge der Trainingssamples zu klassifizieren und die falsch klassifizierten Samples werden zu der Trainingsmenge hinzugefügt. Dies wird so lange wiederholt, bis entweder alle Samples im Trainingsdatensatz enthalten sind oder richtig klassifiziert wurden.

Optimierung der binären Klassengewichte Nachdem eine SVM trainiert wurde, kann deren Arbeitspunkt auf der ROC-Kurve (Receiver Operating Characteristic) durch

eine Veränderung des Bias b in Gleichung D.1 verändert werden. Dies entspricht einer Verschiebung der Hyperebene entlang ihrer Normalen im Merkmalsraum. Damit während der Suche nach einem geeigneten Arbeitspunkt auf der ROC-Kurve auch die Lage der Hyperebene optimiert wird, kann bereits beim Training ein Verhältnis für die binären Klassengewichte angegeben werden. Der Standardwert für dieses Verhältnis ist 1,0. In diesem Optimierungsschritt wird dieses Verhältnis so lange adaptiert wie die False Positive Rate (FPR) oder die False Negative Rate (FNR) 1,0 ist. Dieser Optimierungsschritt ist nur selten notwendig. Das ist z. B. der Fall, wenn eine Klasse im Merkmalsraum durch die Andere eingeschlossen ist und nur durch wenige Trainingssamples repräsentiert wird (Abb. D.3). Dies tritt jedoch insbesondere bei hochdimensionalen Merkmalsräumen sehr selten auf.

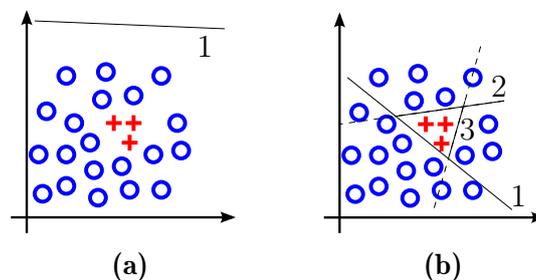


Abbildung D.3. Einfluss des Gewichtsverhältnisses der binären Klassen: Beim Training einer SVM auf zwei nicht linear separierbaren Klassen wird durch (a) eine gleiche Wichtung der Klassen erreicht, dass keine Trennung der Daten vorgenommen wird. (b) Durch Erhöhung des Gewichts der Klasse + wird erreicht, dass eine Separation des Merkmalsraumes durch Hyperebene 1 vorgenommen wird. Durch zwei weitere Knoten des Entscheidungsbaumes wäre eine korrekte Klassifikation der Daten möglich.

D.4. Evaluation des SVM Entscheidungsbaumes

Um die Leistungsfähigkeit des SVM Entscheidungsbaumes mit alternativen Verfahren (Abschnitt D.2) zu vergleichen, wurden neben dem für diese Arbeit relevanten Datensatz auch weitere Datensätze, wie sie in [Chang, 2010] verwendet wurden, zum Benchmarking verwendet. Diese Datensätze können aus dem UCI Machine Learning Repository [Bache, 2013] heruntergeladen werden.

Der im Rahmen dieser Arbeit erzeugte Oberkörperdatensatz mit HOG-Merkmalen (HOG-UB), welcher, wie in Abschnitt 4.2.3 beschrieben, aufgenommen wurde, umfasst 1.623.701 Bildausschnitte, welche bzgl. der neun Klassen gelabelt sind. Jeder Bildausschnitt wurde auf 72 x 72 Pixel skaliert und es wurde ein HOG-Deskriptor mit 16 x 16 Pixel Blöcken angewendet. Jeder Block besteht aus vier 8 x 8 Pixel großen Zellen und für jede Zelle wird ein Histogramm über 9 Gradientenorientierungen gebildet. Daraus ergeben sich 36

Histogrammwerte für einen Block und da die Blöcke mit einem Offset von 8 Pixeln in dem Bildausschnitt platziert sind, ergibt sich ein 2.304 dimensionaler Vektor pro Bildausschnitt.

Eine Charakterisierung aller zur Validierung verwendeten Datensätze ist in Tabelle D.1 gezeigt. Für jeden Datensatz ist die Anzahl zu unterscheidender Klassen, der Merkmalsdimensionen und der Trainings-, Validierungs- und Testsamples aufgelistet.

Tabelle D.1. Evaluierungsdatensätze: Öffentliche Datensätze aus [Chang, 2010] und der HOG-UB Datensatz dieser Arbeit (Übernommen aus [Weinrich, 2012])

	#Klassen	#Dimensionen	#Training	#Validierung	#Test	\sum Samples
PHW	10	16	7,227	1,872	1,893	10,992
Letter	26	16	13,294	3,336	3,370	20,000
Shuttle	7	9	38,664	9,573	9,763	58,000
Poker	10	10	16,674	4,165	4,171	25,010
CI	2	14	30,148	7,537	7,537	45,222
Forest	7	54	387,343	96,835	96,834	581,012
PPI	2	14	836,544	206,635	206,635	1,249,814
KDD	5	41	3,265,623	816,405	816,403	4,898,431
HOG-UB	9	2304	1,411,880	141,185	70,636	1,623,701

Die Accuracy der kleineren Datensätzen ist in Tabelle D.2 gezeigt, wobei SVM Tree das in dieser Arbeit beschriebene Verfahren bezeichnet. Die zur Klassifikation aller Testsamples benötigten CPU-Zyklen sind in Tabelle D.3 aufgelistet. Der SVM Tree erreicht bei ähnlichem Rechenaufwand auf allen Datensätzen eine bessere Accuracy als der ID3 Tree und die 1-v-R SVM. Die Accuracy des DTSVM wird durch den SVM Tree nicht erreicht, allerdings beträgt der Rechenaufwand nur einen Bruchteil.

Tabelle D.2. Klassifikationsgüte I: Accuracy[%] der weniger umfangreichen Evaluierungsdatensätze (Übernommen aus [Weinrich, 2012])

	PHW	Letter	Shuttle	Poker	CI
DTSVM[Chang, 2010]	99.52	97.66	99.89	56.75	84.81
SVM Tree [Weinrich, 2012]	98.42	89.14	99.98	53.97	84.22
ID3 Tree[Quinlan, 1986]	95.51	87.18	99.95	49.72	80.97
1-v-R SVM [Vapnik, 1998]	91.86	70.77	91.23	49.94	83.29

Die Tabellen D.4 und D.5 zeigen dementsprechend die Accuracy und den Rechenaufwand der größeren Datensätze. Für den HOG-UB Datensatz, welcher im Rahmen dieser Arbeit erzeugt wurde, wird die Balanced Accuracy (BAC) statt der Accuracy (ACC) angegeben, da dieser Datensatz bzgl. der Hintergrundklasse stark unbalanciert ist.

Tabelle D.3. Rechenaufwand I: CPU-Zyklen zur Klassifikation der weniger umfangreichen Evaluierungsdatensätze (Übernommen aus [Weinrich, 2012])

	PHW	Letter	Shuttle	Poker	CI
DTSVM[Chang, 2010]	$1,4 \cdot 10^8$	$7,6 \cdot 10^9$	$1,8 \cdot 10^7$	$1,0 \cdot 10^9$	$7,0 \cdot 10^8$
SVM Tree [Weinrich, 2012]	$2,7 \cdot 10^6$	$5,6 \cdot 10^6$	$5,4 \cdot 10^6$	$7,3 \cdot 10^6$	$1,3 \cdot 10^7$
ID3 Tree[Quinlan, 1986]	$2,4 \cdot 10^6$	$5,1 \cdot 10^6$	$5,5 \cdot 10^6$	$9,5 \cdot 10^6$	$1,8 \cdot 10^7$
1-v-R SVM [Vapnik, 1998]	$3,6 \cdot 10^6$	$8,8 \cdot 10^6$	$1,7 \cdot 10^7$	$7,7 \cdot 10^7$	$1,3 \cdot 10^7$

Tabelle D.4. Klassifikationsgüte II: Accuracy bzw. Balanced Accuracy der umfangreichen Datensätze (Übernommen aus [Weinrich, 2012])

	Forest ACC[%]	PPI ACC[%]	KDD ACC[%]	HOG-UB BAC[%]
DTSVM[Chang, 2010]	94.59	92.29	99.99	51.61*
SVM Tree [Weinrich, 2012]	94.41	90.99	99.99	64.82
ID3 Tree[Quinlan, 1986]	93.31	88.11	99.99	33.07
1-v-R SVM [Vapnik, 1998]	71.50	87.42	99.81	55.23

Tabelle D.5. Rechenaufwand II: CPU-Zyklen zur Klassifikation der umfangreichen Evaluierungsdatensätze (Übernommen aus [Weinrich, 2012])

	Forest	PPI	KDD	HOG-UB
DTSVM[Chang, 2010]	$4,2 \cdot 10^9$	$3,0 \cdot 10^{10}$	$2,3 \cdot 10^9$	$1,3 \cdot 10^{11}$
SVM Tree [Weinrich, 2012]	$6,8 \cdot 10^8$	$1,9 \cdot 10^9$	$1,5 \cdot 10^9$	$2,6 \cdot 10^9$
ID3 Tree[Quinlan, 1986]	$6,5 \cdot 10^8$	$1,9 \cdot 10^9$	$1,9 \cdot 10^9$	$1,0 \cdot 10^{10}$
1-v-R SVM [Vapnik, 1998]	$2,6 \cdot 10^8$	$3,1 \cdot 10^8$	$2,5 \cdot 10^9$	$8,8 \cdot 10^9$

Bei den fremden umfangreichen Evaluierungsdatensätzen verhält sich die Accuracy und der Rechenaufwand ähnlich den kleineren Datensätzen. Bei dem hochdimensionalen HOG-UB Datensatz werden jedoch die Vorteile der linearen SVM deutlich. Sowohl die 1-v-R SVM als auch der SVM Tree erreichen eine sehr hohe BAC bei geringem Rechenaufwand, wobei sich der SVM Tree noch einmal deutlich abhebt. Die Verwendung nicht-linearer SVMs wie im DTSVM ist bei einem solchen Datensatz nicht geeignet. Selbst unter Verwendung von 128 GB RAM konnte das Verfahren auf nur 10 % des HOG-UB Datensatzes trainiert werden. Dieses Training dauerte trotz Nutzung eines 3.5 GHz CPU über 45 Stunden, wohingegen für das Training des SVM Tree auf dem gesamten Datensatz nur 1.4 Stunden benötigt wurden. Im Gegensatz zu den anderen Datensätzen übertrifft der SVM Tree auf dem hochdimensionalen HOG-UB Datensatz auch den DTSVM in der Klassifikationsgüte und dem Rechenaufwand. Weitere Experimente zur Orientierungsschätzung mittels SVM Entscheidungsbaum sind in Abschnitt 4.2.5 beschrieben.

D.5. Transformation von 3D-Hypothesen mit Unsicherheitsfortpflanzung

Der Personentracker (Abschnitt 4.3) fusioniert die Hypothesen des laserbasierten und des kamerabasierten Detektors. Allerdings sind die Hypothesen der beiden Detektoren relativ zu dem Koordinatensystem des entsprechenden Sensors spezifiziert und müssen vor der Fusion in dasselbe Koordinatensystem transformiert werden. Damit das Tracking unabhängig von der Bewegung des Roboters ist, werden die Hypothesen durch den Tracker in ein statisches Koordinatensystem transformiert. Während dieser Transformation müssen die Unsicherheiten der Detektoren mit der Lokalisationsungenauigkeit des Roboters relativ zu dem statischen Koordinatensystem verrechnet werden.

Die Transformation von unsicheren Hypothesen zwischen verschiedenen Koordinatensystemen, deren relative Lage selbst mit Unsicherheiten behaftet ist, ist jedoch nicht nur für den Tracker relevant. In der Middleware MIRA¹ werden die Koordinatensysteme und deren relative Lage zueinander in einem Transformationsframework verwaltet. Damit die Verrechnung der Unsicherheiten jeder Mira-Unit transparent zur Verfügung steht, wurde diese Verrechnung im Rahmen dieser Arbeit in das Transformationsframework integriert. Es folgt eine Beschreibung des Transformationsframeworks. Für die mathematischen Details zur Verrechnung der Unsicherheiten sei auf [Blanco, 2010] verwiesen.

D.5.1. Euklidische Transformation im 3D

Eine Pose \mathbf{p} im dreidimensionalen Raum \mathbb{R}^3 hat 3 translatorische und 3 rotatorische Freiheitsgrade. Die in MIRA verwendeten Posen sind als 3D-Translation und 4D-Quaternion parametrisiert:

$$\mathbf{p} = \left[x \quad y \quad z \quad q_r \quad q_x \quad q_y \quad q_z \right]^T \quad (\text{D.3})$$

Quaternionen haben einen redundanten Freiheitsgrad, denn es gilt die Einschränkung $q_r^2 + q_x^2 + q_y^2 + q_z^2 = 1$. Die Eulerwinkel als alternative Parametrierung der Rotation werden nur zur benutzerfreundlichen Ein- und Ausgabe verwendet. Intern kommen Eulerwinkel wegen der Problematik des Gimbal Locks nicht zum Einsatz. Denn bei der Verwendung von Eulerwinkeln führt ein Pitch von $\pm 90^\circ$ dazu, dass die Drehachsen von Yaw und Roll

¹www.mira-project.org

zusammenfallen und dadurch ein Freiheitsgrad verloren geht. Auch homogene 4 x 4 Transformationsmatrizen werden intern wegen deren hohen Redundanz nicht verwendet.

Repräsentation von Unsicherheiten Um Unsicherheiten im Transformationsframework darzustellen, sind diese als mehrdimensionale Normalverteilung über eine 7D Pose (Formel D.3) mit dem Mittelwert $\bar{\mathbf{p}}$ und der 7 x 7 Kovarianzmatrix $\text{cov}(\mathbf{p}) = \mathbf{C}_p$ parametrisiert.

D.5.2. Transformationen zwischen Koordinatensystemen

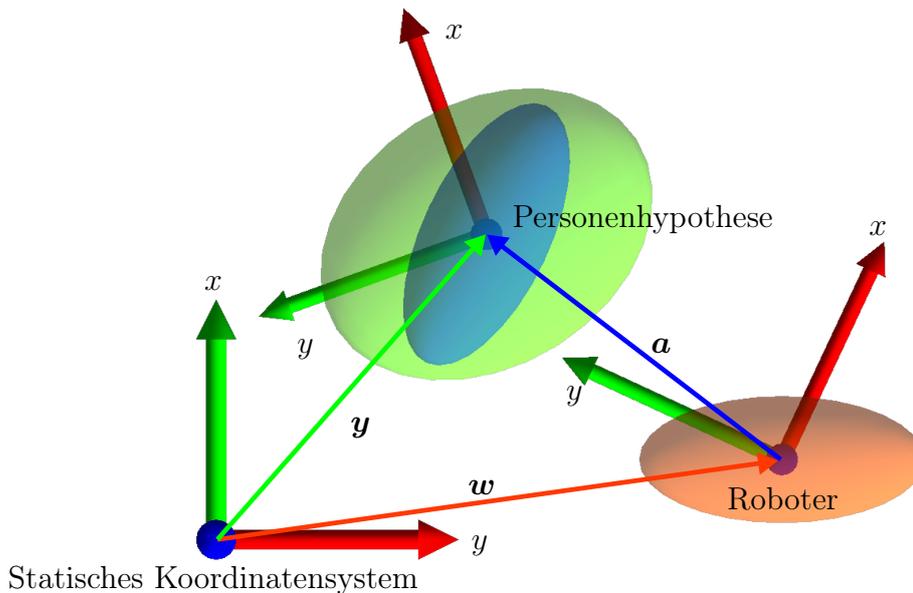


Abbildung D.4. Verrechnung der Unsicherheiten bei der Transformation zwischen Koordinatensystemen: Das Beispiel zeigt die unsichere Pose a einer Personenhypothese relativ zum Koordinatensystem des Roboters (blau). Die unsichere Pose des Roboters w ist wiederum relativ zum statischen Koordinatensystem spezifiziert (orangefaben). Durch die Verknüpfung dieser beiden unsicheren Transformationen zu y ergibt sich die Unsicherheit der Personenhypothese relativ zum statischen Koordinatensystem (grün).

Wie bereits oben erwähnt verwaltet das Transformationsframework die Lage verschiedener Koordinatensysteme zueinander. Dazu werden die Koordinatensysteme in einer baumartigen Struktur gespeichert, in welcher jedes Koordinatensystem durch eine euklidische Transformation relativ zu einem übergeordneten Referenz-Koordinatensystem spezifiziert ist. Das Framework ermöglicht die Berechnung der Transformation zwischen

Inverse Transformation Die Inverse einer Transformation beschreibt die Lage des Referenz-Koordinatensystems in dem Relativ-Koordinatensystem. Angenommen in Abb. D.4 seien die Transformation \mathbf{y} und \mathbf{a} gegeben und die Transformation \mathbf{w} sei gesucht, so müsste \mathbf{a} invertiert werden, damit die Verknüpfung $\mathbf{w} = \mathbf{y} \cdot \mathbf{a}^{-1}$ berechnet werden kann. Soll die Inverse \mathbf{p}^{-1} einer euklidischen Transformation \mathbf{p} bestimmt werden, so berechnet sich der Mittelwert $\bar{\mathbf{p}}^{-1}$ und die Varianz \mathbf{C}_p^{-1} der resultierenden Transformation wie folgt.

$$\mathbf{p}^{-1} = \begin{pmatrix} -\mathbf{p}_x + 2 \left((\mathbf{p}_{q_y}^2 + \mathbf{p}_{q_z}^2) \mathbf{p}_x - (\mathbf{p}_{q_x} \mathbf{p}_{q_y} - \mathbf{p}_{q_r} \mathbf{p}_{q_z}) \mathbf{p}_y - (\mathbf{p}_{q_r} \mathbf{p}_{q_y} + \mathbf{p}_{q_x} \mathbf{p}_{q_z}) \mathbf{p}_z \right) \\ -\mathbf{p}_y + 2 \left(-(\mathbf{p}_{q_r} \mathbf{p}_{q_z} + \mathbf{p}_{q_x} \mathbf{p}_{q_y}) \mathbf{p}_x + (\mathbf{p}_{q_x}^2 + \mathbf{p}_{q_z}^2) \mathbf{p}_y - (\mathbf{p}_{q_y} \mathbf{p}_{q_z} - \mathbf{p}_{q_r} \mathbf{p}_{q_x}) \mathbf{p}_z \right) \\ -\mathbf{p}_z + 2 \left(-(\mathbf{p}_{q_x} \mathbf{p}_{q_z} - \mathbf{p}_{q_r} \mathbf{p}_{q_y}) \mathbf{p}_x - (\mathbf{p}_{q_r} \mathbf{p}_{q_x} + \mathbf{p}_{q_y} \mathbf{p}_{q_z}) \mathbf{p}_y + (\mathbf{p}_{q_x}^2 + \mathbf{p}_{q_y}^2) \mathbf{p}_z \right) \\ \mathbf{p}_{q_r} \\ -\mathbf{p}_{q_x} \\ -\mathbf{p}_{q_y} \\ -\mathbf{p}_{q_z} \end{pmatrix} \quad (\text{D.11})$$

$$\mathbf{C}_p^{-1} = \mathbf{J} \mathbf{C}_p \mathbf{J}^T \quad (\text{D.12})$$

$$\mathbf{J} = \frac{\delta \mathbf{p}^{-1}}{\delta \mathbf{p}} \quad (\text{D.13})$$

$$\mathbf{J} = \begin{pmatrix} \mathbf{J}_{3 \times 3} & & \mathbf{J}_{3 \times 4} & & \\ & 1 & 0 & 0 & 0 \\ \mathbf{0}_{4 \times 3} & 0 & -1 & 0 & 0 \\ & 0 & 0 & -1 & 0 \\ & 0 & 0 & 0 & -1 \end{pmatrix} \quad (\text{D.14})$$

$$\mathbf{J}_{3 \times 3} = \begin{pmatrix} 2(q_y q_y + q_z q_z) - 1 & -2(q_r q_z + q_x q_y) & 2(q_r q_y - q_x q_z) \\ 2(q_r q_z - q_x q_y) & 2(q_x q_x + q_z q_z) - 1 & -2(q_r q_x + q_y q_z) \\ -2(q_r q_y + q_x q_z) & 2(q_r q_x - q_y q_z) & 2(q_x q_x + q_y q_y) - 1 \end{pmatrix} \quad (\text{D.15})$$

$$\mathbf{J}_{3 \times 4} = \begin{pmatrix} -2(q_y z - q_z y) & -2(q_y y + q_z z) & -2(q_x y - 2q_y x + q_r z) & -2(q_x z - q_r y - 2q_z x) \\ -2(q_z x - q_x z) & -2(q_y x - 2q_x y - q_r z) & -2(q_x x + q_z z) & -2(q_r x - 2q_z y + 2q_y z) \\ -2(q_x y - q_y x) & -2(q_z x + q_r y - 2q_x z) & -2(q_z y - q_r x - 2q_y z) & -2(q_x x + q_y y) \end{pmatrix} \quad (\text{D.16})$$

E

Erscheinungsbasierte Oberkörpermodellie- rung

In Abschnitt 5.2 wurde ein erscheinungsbasiertes Modell beschrieben. Dieses wird zur Bewertung der Übereinstimmung $f(\mathbf{I}, \boldsymbol{\theta})$ des Bildes \mathbf{I} mit dem Modell entsprechend der Modellparameter $\boldsymbol{\theta}$ verwendet. Die Bewertung, welche auf verschiedenen Merkmalen basiert, wird nachfolgend genauer erklärt.

E.1. Shader des Kantenmodells

Wie in Abschnitt 5.2 beschrieben, vergleicht das Kantenmodell die erwarteten Kanten des gelenkigen 3D-Modells entsprechend der Modellparameter $\boldsymbol{\theta}$ mit den tatsächlichen Gradienten des aufgenommenen Bildes. Dieser Vergleich wird hauptsächlich auf der GPU berechnet (Abb. E.1). Zuerst wird das 3D-Modell inklusive Gelenkstellungen entsprechend der Modellparameter $\boldsymbol{\theta}$ transformiert. Ein spezieller Vertex-Shader berechnet zu jedem Vertex v (Eckpunkt von dreidimensionalen Objekten) die transformierte Vertexposition \mathbf{p}'_v , die modellierte Gradientenorientierung o_v und die entsprechende Magnitude m_v . Dazu wird die Normale \mathbf{n}_v des Vertex in die Bildebene projiziert und diese Projektion wird als erwartete Gradientenorientierung o_v interpretiert. Die erwartete Magnitude m_v resultiert aus dem Skalarprodukt von \mathbf{n}_v mit der Blickrichtung von der Kamera zu der Vertexposition \mathbf{p}_v . Dementsprechend ist die Magnitude m_v maximal, wenn die Normale \mathbf{n}_v senkrecht zur Blickrichtung orientiert ist. Das Verfahren ist soweit dem Cel Shading

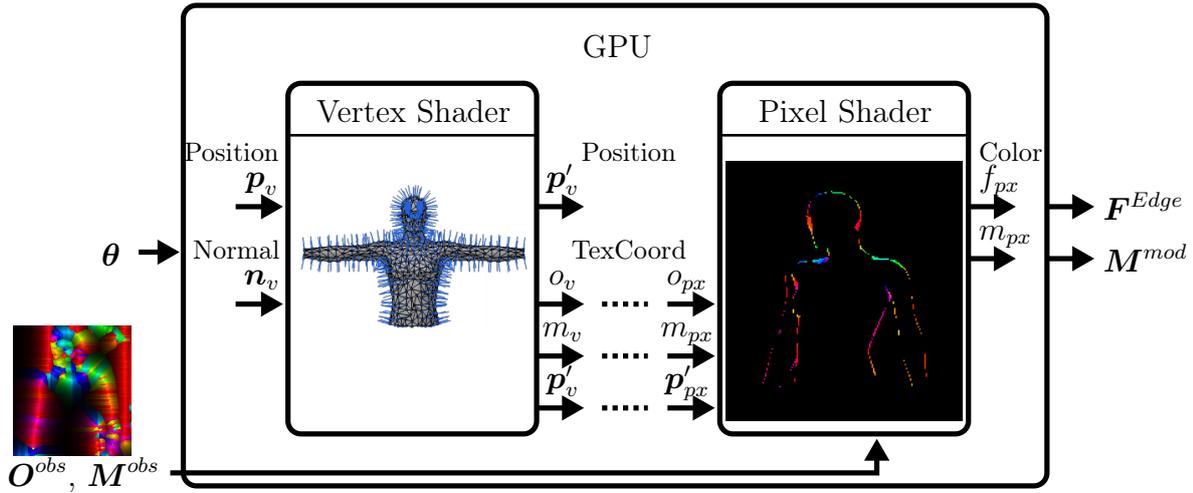


Abbildung E.1. Shader des Kantenmodells: Der Vertex-Shader berechnet für jeden Vertex v auf Basis dessen Normale \mathbf{n}_v die Orientierung o_v und Magnitude m_v des Gradienten, welcher beim Rendern des Modells entstehen würde. Nach der Rasterisierung wird durch den Pixel-Shader die Übereinstimmung f_{px} dieser Werte mit der entsprechenden Orientierung $\mathbf{O}^{obs}(\mathbf{p}'_{px})$ und Magnitude $\mathbf{M}^{obs}(\mathbf{p}'_{px})$ des aufgenommenen Bildes bestimmt.

sehr ähnlich. Im Gegensatz zum Cel Shading werden die modellierten Gradienten jedoch nicht gerendert. Stattdessen wird durch den Pixel-Shader direkt die Übereinstimmung f_{px} zwischen modellierter Orientierung o_{px} und der entsprechenden Orientierung $\mathbf{O}^{obs}(\mathbf{p}'_{px})$ in dem aufgenommenem Bild berechnet. Zu diesem Zweck werden, wie in Abschnitt 5.2 beschrieben, die Gradientenorientierungen \mathbf{O}^{obs} und -magnituden \mathbf{M}^{obs} des aufgenommenen Bildes \mathbf{I} auf der CPU berechnet und auf die Grafikkarte übertragen. In die Berechnung der Übereinstimmung fließen auch die modellierte Magnitude m_{px} und die aufgenommene $\mathbf{M}^{obs}(\mathbf{p}'_{px})$ Magnitude ein:

$$f_{px} = \exp\left(-\frac{1}{2}\left(\frac{o_{px} - \mathbf{O}^{obs}(\mathbf{p}'_{px})}{\sigma}\right)^2\right) \cdot m_{px} \cdot \mathbf{M}^{obs}(\mathbf{p}'_{px}) \quad (\text{E.1})$$

Die Übereinstimmung f_{px} aller Pixel wird in die Textur \mathbf{F}^{Edge} gerendert. Auf der CPU wird die Summe der Übereinstimmung aller Pixel berechnet und mit der Summe aller modellierten Magnituden normiert.

$$f_{Edg}(\mathbf{I}, \theta) = \frac{\sum_{px} f_{px}}{\sum_{px} m_{px}} \quad (\text{E.2})$$

E.2. Shader des Farbmodells

Wie in Abschnitt 5.2 beschrieben, besteht ein wesentlicher Verarbeitungsschritt des Farbmodells in der Ermittlung der Oberkörpertextur, welche bei gegebenen Modellparametern θ zu dem aufgenommenen Kamerabild I geführt hätte. Allgemeiner formuliert sollen im Gegensatz zum typischen Rendern nicht Szenenattribute genutzt werden, um ein realistisches Bild zu erzeugen, sondern es soll ausgehend von dem aufgenommenen Bild und gegebenen Modellparametern θ die Oberkörpertextur ermittelt werden. Dies wird im folgenden als inverses Rendern bezeichnet.

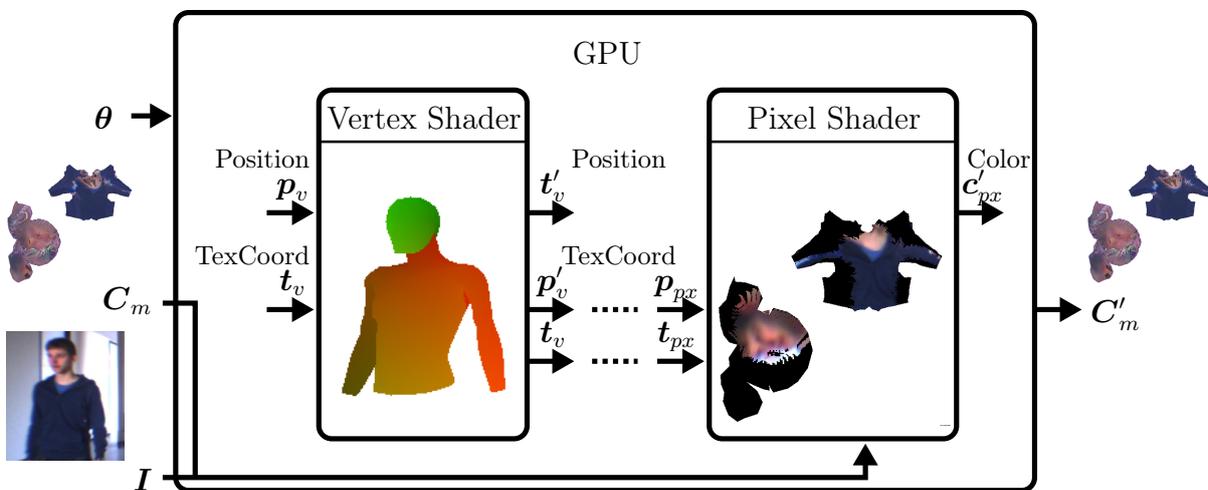


Abbildung E.2. Shader des Farbmodells: Die Besonderheit des inversen Renderns ist, dass der Vertex-Shader die Texturkoordination t'_v als Renderposition ausgibt. Die transformierte Vertexposition p'_v , welche typischerweise als Renderposition ausgegeben wird, wird hier als Texturkoordinate berechnet. Auf diese Weise operiert der Pixel-Shader über den Texturkoordinaten, greift dabei jedoch auf die transformierten Vertexpositionen des aufgenommenen Bildes $I(p_{px})$ und das entsprechende Texel des Farbmodells $C_m(t_{px})$ zu.

Der entscheidende Unterschied zum typischen Rendern besteht in dem Vertex-Shader (Abb. E.2). Wie beim Vertex-Shader des Kantenmodells besteht dessen Hauptaufgabe typischerweise darin, die 3D-Position des jeweiligen Vertex in die Renderposition des Vertex zu transformieren. Zum inversen Rendern wird jedoch die Renderposition völlig unabhängig von der Vertexposition p_v bestimmt. Stattdessen wird die Texturkoordinate t_v als Renderposition t'_v ausgegeben. Auf diese Weise wird der Pixel-Shader nach der Rasterisierung die Texturpositionen jedes Dreiecks bearbeiten. Damit dem Pixel-Shader auch die entsprechende Szenenposition zur Verfügung steht, wird auch die in Kamerako-

ordinaten transformierte Vertexposition \mathbf{p}'_v durch den Vertex-Shader berechnet und als Texturkoordinate (TexCoord) ausgegeben.

In Abbildung E.2 ist der Pixel-Shader zur Adaption eines Farbmodells \mathbf{C}_m dargestellt. Der Pixel-Shader berechnet den neuen Wert des Pixels des Farbmodells \mathbf{c}'_{px} indem das entsprechende Pixel des bestehenden Farbmodells $\mathbf{C}_m(\mathbf{t}_{px})$ mit der aktuellen Beobachtung in der realen Szene $\mathbf{I}(\mathbf{p}_{px})$ verrechnet wird.

Ebenso wird der gleiche Vertex-Shader mit einem anderen Pixel-Shader kombiniert, um die Übereinstimmung $f_{Col}(\mathbf{I}, \boldsymbol{\theta}, \mathbf{C}_m)$ eines Farbmodells \mathbf{C}_m mit der aktuellen Beobachtung \mathbf{I} bei Modellparametern $\boldsymbol{\theta}$ zu bestimmen.

E.3. Eigenschaften der Matching-Funktion

Um die Optimierung (mittels Particle Swarm Optimization) über der Matching-Funktion $f(\mathbf{I}, \mathbf{S}, \boldsymbol{\theta})$ zu begünstigen, sollte diese kontinuierlich und möglichst weitläufig ansteigen, während die Modellparameter $\boldsymbol{\theta}$ mit der tatsächlichen Pose der Person konvergieren. Die Funktion sollte also einerseits tolerant gegenüber Abweichungen der Modellparameter sein, andererseits sollte die Matching-Funktion ein eindeutiges Maximum bei korrekter Parameterwahl aufweisen.

In den Abbildungen E.3a bis E.3d ist eine exemplarische Matching-Funktion über jeweils zwei Dimensionen des Parameterraumes Θ gezeigt. Die verwendeten Funktionsparameter (Gleichung. 5.1) sind $\gamma = 0,1$, $\omega_{Edg} = 0,1$, $\omega_{Seg} = 1,0$, $\omega_{Col} = 1,0$, $\omega_{HOG} = 0,1$ und $\omega_{Leg} = 0,1$. Die korrekten Posenparameter befinden sich im Zentrum der visualisierten Parameterkonfiguration.

Abbildung E.3a und E.3c zeigen, dass die Matching-Funktion deutliche Gradienten bzgl. der Personenposition aufweist. Andererseits zeigen Abbildung E.3d und E.3b, dass die Matching-Funktion weniger sensitiv bzgl. der Oberkörperorientierung ist und sich die tatsächliche Orientierung nur leicht abhebt.

Wenn anstatt des universellen Farbmodells das an die Person angepasste Farbmodell verwendet wird, ergibt sich die in den Abbildungen E.4a bis E.4d dargestellte Matching-Funktion. Durch die Verwendung des spezifischen Farbmodells ist der Maximalwert der Matching-Funktion (0,263) gegenüber dem Maximalwert bei Verwendung des unspezifischen Farbmodells (0,228) deutlich angestiegen, und somit erfüllt das Farbmodell den Zweck der Wiedererkennung. Darüber hinaus hebt sich durch das personenspezifische Farbmodell die korrekte Pose insbesondere bzgl. der Oberkörperorientierung deutlicher

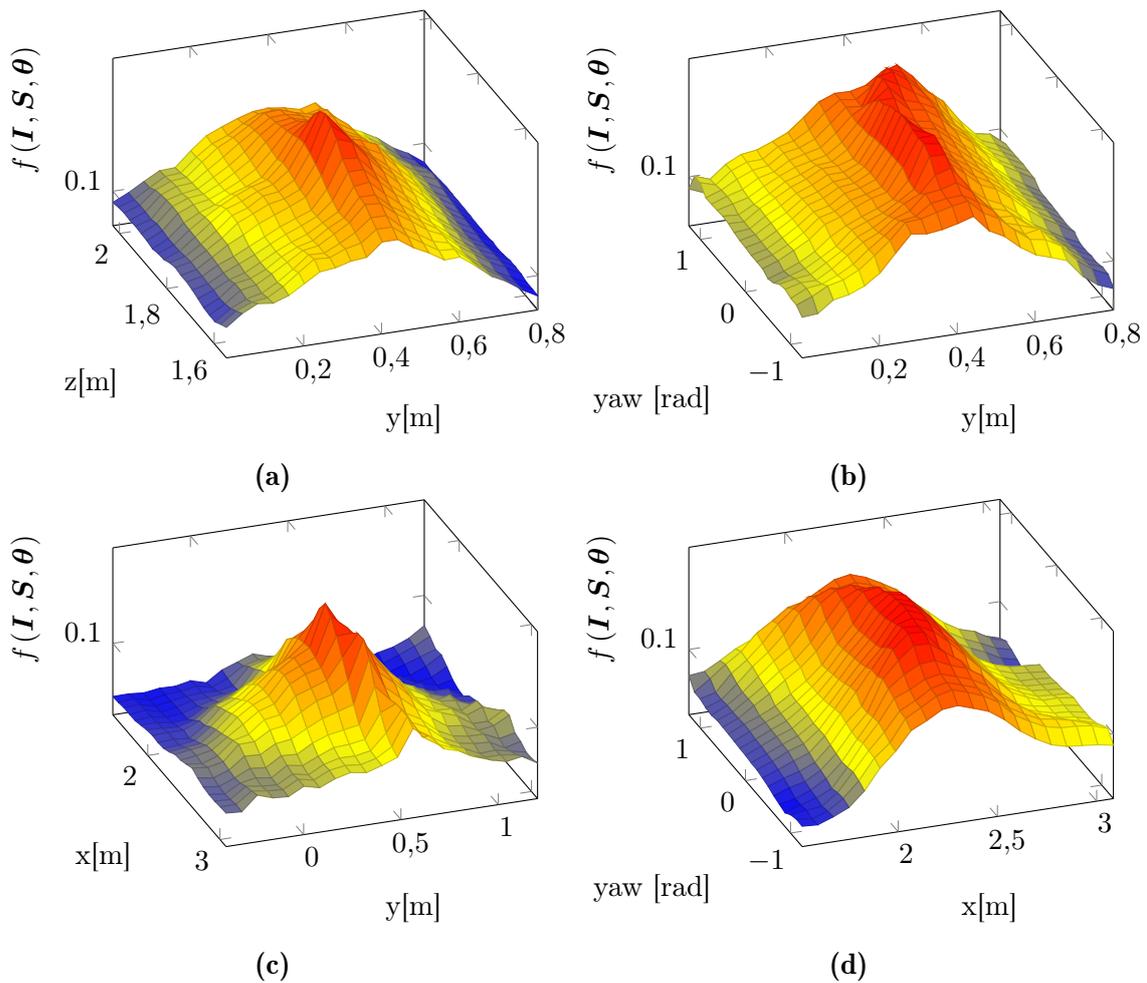


Abbildung E.3. Universelle Matching-Funktion: Visualisierung der Matching-Funktion $f(I, S, \theta)$ mit universellem Farbmodell c_0 über jeweils zwei der 14 Parameter des 3D-Modells. Die korrekte Parameterkonfiguration befindet sich im Zentrum des visualisierten Parameterraumes.

ab. Es sei angemerkt, dass trotz falsch geschätzten Oberkörperorientierungen während der Adaption des personenspezifischen Modells, das gelernte Modell zu einer besseren Schätzung der Oberkörperorientierung führen kann. Trägt eine Person beispielsweise einfarbige Kleidung, würde trotz falscher Orientierungsschätzung die richtige Farbe der Kleidung gelernt. Ein einfarbiges Farbmodell führt, wie in Abbildung E.4d und E.4b dargestellt, zu einer verbesserten Orientierungsschätzung, weil das 3D-Modell einer Person nicht rotationsinvariant ist und ein spezifisches Farbmodell nur bei passender Form eine hohe Übereinstimmung erreicht.

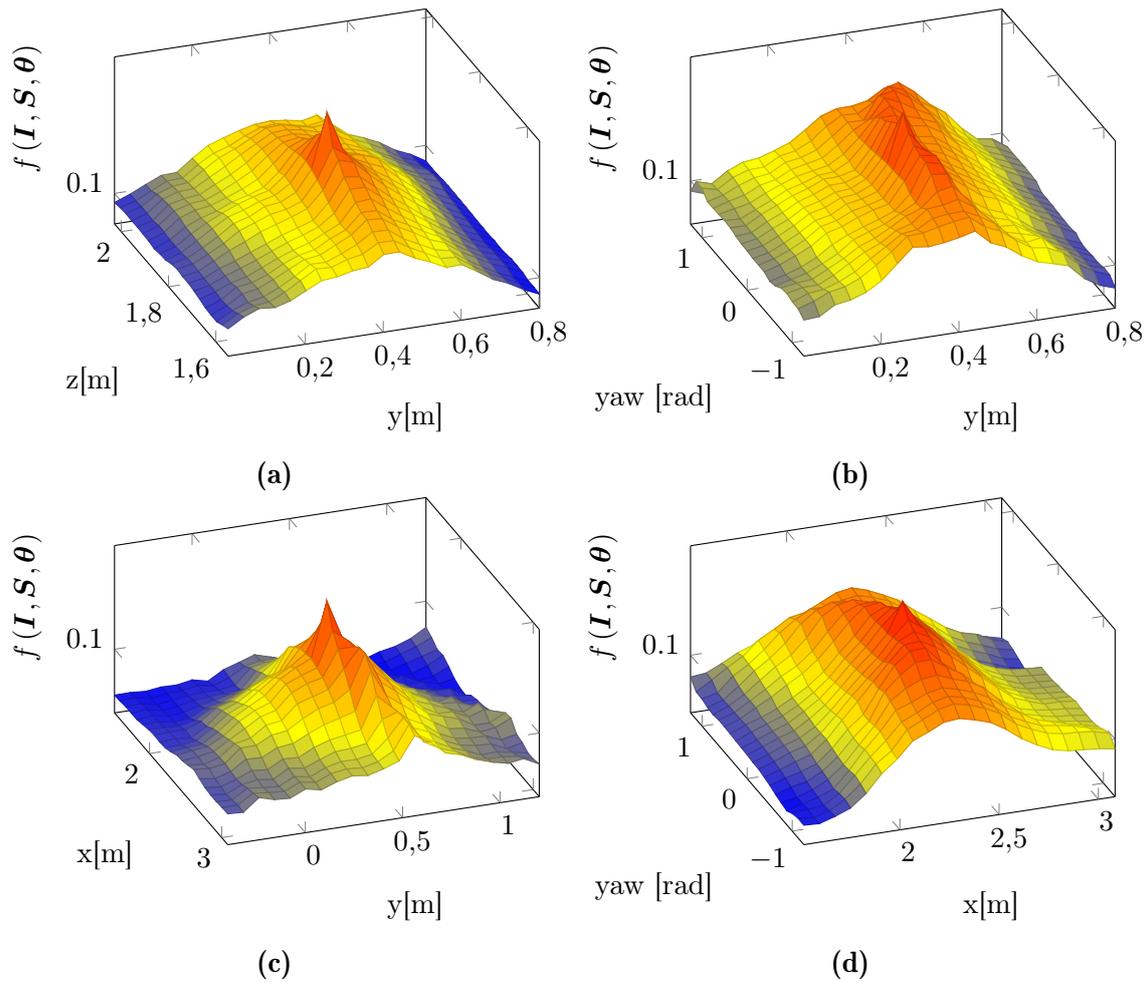


Abbildung E.4. Matching-Funktion mit personenspezifischem Farbmodell: Visualisierung der Matching-Funktion $f(\mathbf{I}, \mathbf{S}, \theta)$ mit spezifischem Farbmodell $C_{m>0}$ über jeweils zwei der 14 Parameter des 3D-Modells. Die korrekte Parameterkonfiguration befindet sich im Zentrum des visualisierten Parameterraumes.

F

On-line Lernen menschlicher Bewegungsmuster

F.1. Gelernte Prädiktion zum reaktiven Ausweichen

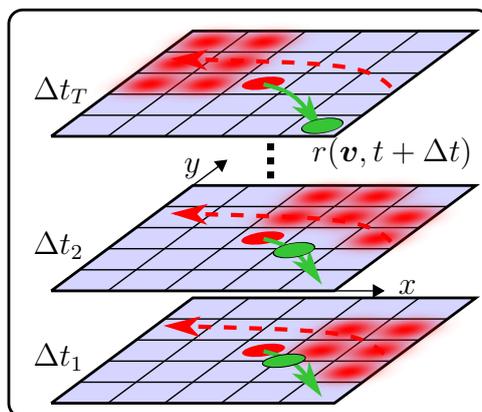


Abbildung F.1. Distanzbasierte Kosten $\text{cost}_{\text{PS}}^{\text{bel,d}}$: Wie in Abb. 6.2 dargestellt, liefert das Verfahren zur Bewegungsprädiktion für jede Personenhypothese und dem Prädiktionshorizont Δt eine Belief Distribution Map, welche die zukünftige Aufenthaltswahrscheinlichkeit der Person darstellt. Um ausgehend von der Verteilung der Aufenthaltswahrscheinlichkeit die Verletzung des Personal Space der Person zu berechnen, wird für jede Zelle (x, y) der Personal Space durch eine 2D-Gaußfunktion (rot), welche mit der entsprechenden Aufenthaltswahrscheinlichkeit skaliert ist, repräsentiert. Die Gesamtkosten $\text{cost}_{\text{PS}}^{\text{bel,d}}$ ergeben sich durch die Überlagerung der Funktionswerte aller Gaußfunktionen an der prädizierten Position des Roboters (grüner Kreis).

In Abschnitt 3.3.1 ist im Kontext des reaktiven Ausweichens die Formel 3.1 zur Bewertung der Verletzung des Personal Space für zukünftige Zeitpunkte Δt aufgeführt. Diese Formel basiert auf einer abstrakten Funktion $\text{cost}_{\text{PS}}^*(\mathbf{h}_i, t, \Delta t, \mathbf{v})$, welche die Nähe und Geschwindigkeit des Roboters gegenüber der zukünftigen Position einer Personenhypothese bewertet. Für die konkrete Realisierung dieser abstrakten Funktion können verschiedene Prädiktionsmethoden eingesetzt werden. In Kapitel C.2 wird dazu die Kostenfunktion $\text{cost}_{\text{PS}}^{\text{lin}}(\mathbf{h}_i, t, \Delta t, \mathbf{v})$ mit linearer Prädiktion als Referenzmethode beschrieben. Die Kostenfunktion $\text{cost}_{\text{PS}}^{\text{bel}}(\mathbf{h}_i, t, \Delta t, \mathbf{v})$, welche die gelernte Prädiktion aus Kapitel 6 verwendet, ist nachfolgend aufgeführt:

$$\text{cost}_{\text{PS}}^{\text{bel}}(\mathbf{h}_i, t, \Delta t, \mathbf{v}) = \sum_{x,y} \text{cost}_{\text{PS}}^{\text{bel,d}}(\mathbf{h}_i, t, \Delta t, \text{pos}(\mathbf{r}, t, \Delta t, \mathbf{v}), x, y) \cdot \left(1,0 + \alpha^{\text{app}} \cdot \max\left(0, \text{cost}_{\text{PS}}^{\text{bel,v}}(t, \Delta t, \mathbf{v}, x, y)\right)\right) \quad (\text{F.1})$$

$$\text{cost}_{\text{PS}}^{\text{bel,d}}(\mathbf{h}_i, t, \Delta t, \mathbf{r}', x, y) = \exp\left(-\frac{1}{2} \left(\frac{|\text{dist}^{\text{bel}}(t, \Delta t, \mathbf{r}', x, y)|}{\sigma}\right)^2\right) \cdot \frac{\text{Bel}(c(\mathbf{s}_i(t)), \Delta t, x, y)}{\sum_{x',y'} \text{Bel}(c(\mathbf{s}_i(t)), \Delta t, x', y')} \quad (\text{F.2})$$

$$\text{cost}_{\text{PS}}^{\text{bel,v}}(t, \Delta t, \mathbf{v}, x, y) = \frac{\text{dist}^{\text{bel}}(t, \Delta t, \text{pos}(\mathbf{r}, t, \Delta t, \mathbf{v}), x, y)}{|\text{dist}^{\text{bel}}(t, \Delta t, \text{pos}(\mathbf{r}, t, \Delta t, \mathbf{v}), x, y)|} \cdot v_{\text{trans}}(\mathbf{r}, t, \Delta t, \mathbf{v}) \quad (\text{F.3})$$

$$\text{dist}^{\text{bel}}(t, \Delta t, \mathbf{r}', x, y) = (x, y)^T - \mathbf{r}' \quad (\text{F.4})$$

Ein entscheidender Unterschied zwischen der gelernten Prädiktion und der linearen Prädiktion ist, dass bei der linearen Prädiktion für jeden Prädiktionszeitpunkt Δt eine eindeutige Position der Person ermittelt wird. Im Gegensatz dazu steht bei der gelernten Prädiktion zu jedem Zeitpunkt Δt eine Wahrscheinlichkeitsverteilung über der entsprechenden Belief Distribution Map (Abb. F.1) zur Verfügung. Dementsprechend besteht der Unterschied zwischen der linearen Kostenfunktion (C.3) und der prädizierten Kostenfunktion (F.1) darin, dass die Kosten zu allen Gridzellen der Belief Distribution Map berechnet werden und mit der entsprechenden Wahrscheinlichkeit gewichtet aufsummiert werden.

F.2. Korrelation zwischen Prädiktion und tatsächlicher Bewegung

In Abschnitt 6.3 wurde beschrieben, dass die linear prädizierte Kostenfunktion (Gleichung C.3) und die Kostenfunktion F.1, welche auf der gelernten Prädiktion basiert, mittels Kreuzkorrelation mit der Kostenfunktion $\text{cost}_{\text{PS}}^{\text{gt,d}}$, welche auf den tatsächlichen Personenpositionen basiert, evaluiert werden.

Wie nachfolgend für die Ground-Truth Kostenfunktion gezeigt, eignen sich die Kostenfunktionen zur Evaluation mittels Kreuzkorrelation, weil der durch eine Gaußfunktion modellierte Personal Space eine glättende Wirkung auf die prädizierten bzw. beobachteten Personenpositionen $\text{pos}(\mathbf{h}_i, t)$ hat.

$$\text{cost}_{\text{PS}}^{\text{gt,d}}(\mathbf{r}', t', \mathbf{h}_i) = \exp\left(-\frac{1}{2} \left(\frac{|\mathbf{r}' - \text{pos}(\mathbf{h}_i, t')|}{\sigma}\right)^2\right) \quad (\text{F.5})$$

Das Integral des Produktes aus der Ground-Truth Kostenfunktion $\text{cost}_{\text{PS}}^{\text{gt,d}}$ und der linear prädizierten Kostenfunktion $\text{cost}_{\text{PS}}^{\text{lin,d}}$ bzw. der gelernten Kostenfunktion $\text{cost}_{\text{PS}}^{\text{bel,d}}$ dient als Maß der Korrelation zwischen den beiden Funktionen. Zur Normierung wird durch das Integral des Produktes der Ground-Truth Kostenfunktion mit sich selbst dividiert.

$$cc^{\text{lin}}(t, \Delta t, \mathbf{h}_i) = \frac{\int_{\mathbf{r}' \in \mathbb{R}^2} \text{cost}_{\text{PS}}^{\text{lin,d}}(\mathbf{h}_i, t, \Delta t, \mathbf{r}') \cdot \text{cost}_{\text{PS}}^{\text{gt,d}}(\mathbf{r}', t + \Delta t, \mathbf{h}_i) d\mathbf{r}'}{\int_{\mathbf{r}' \in \mathbb{R}^2} \text{cost}_{\text{PS}}^{\text{gt,d}}(\mathbf{h}_i, t, \Delta t, \mathbf{r}') \cdot \text{cost}_{\text{PS}}^{\text{gt,d}}(\mathbf{r}', t + \Delta t, \mathbf{h}_i) d\mathbf{r}'} \quad (\text{F.6})$$

$$cc^{\text{bel}}(t, \Delta t, \mathbf{h}_i) = \frac{\sum_{x,y} \int_{\mathbf{r}' \in \mathbb{R}^2} \text{cost}_{\text{PS}}^{\text{bel,d}}(\mathbf{h}_i, t, \Delta t, \mathbf{r}', x, y) \cdot \text{cost}_{\text{PS}}^{\text{gt,d}}(\mathbf{r}', t + \Delta t, \mathbf{h}_i) d\mathbf{r}'}{\int_{\mathbf{r}' \in \mathbb{R}^2} \text{cost}_{\text{PS}}^{\text{gt,d}}(\mathbf{h}_i, t, \Delta t, \mathbf{r}') \cdot \text{cost}_{\text{PS}}^{\text{gt,d}}(\mathbf{r}', t + \Delta t, \mathbf{h}_i) d\mathbf{r}'} \quad (\text{F.7})$$

Da alle drei Kostenfunktionen durch Gauß-Funktionen repräsentiert werden, ist das Produkt zweier Kostenfunktionen auch eine Gauß-Funktion, welche analytisch bestimmt werden kann. Das Integral dieser Gauß-Funktion lässt sich ebenso analytisch bestimmen. Zur Berechnung des Produktes brauchen die Gaußfunktionen nur in die kanonische Form überführt zu werden. Diese Überführung ist in Abschnitt F.3 beschrieben. Ausgehend von der kanonischen Form lässt sich das Produkt der beiden Funktionen, wie in Abschnitt F.4 beschrieben, berechnen.

F.3. Momentumform und kanonische Form einer Normalverteilung

In dieser Arbeit werden D-dimensionale Normalverteilungen in zwei verschiedenen Formen beschrieben. Sehr stark verbreitet ist die Momentumform:

$$N(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{\frac{D}{2}} |\boldsymbol{\Sigma}|^{\frac{1}{2}}} \exp \left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right\} \quad (\text{F.8})$$

Durch den Normalisierungsfaktor η wird erreicht, dass das Integral der Funktion gleich 1.0 ist.

$$\eta = \frac{1}{(2\pi)^{\frac{D}{2}} |\boldsymbol{\Sigma}|^{\frac{1}{2}}} \quad (\text{F.9})$$

Für verschiedene Operationen ist jedoch die kanonische Form besser geeignet:

$$N(\mathbf{x}|g, \mathbf{h}, \boldsymbol{\Lambda}) = \exp \left\{ g + \mathbf{h}^T \mathbf{x} - \frac{1}{2} \mathbf{x}^T \boldsymbol{\Lambda} \mathbf{x} \right\} \quad (\text{F.10})$$

Um die Momentumform in die kanonische Form zu überführen, ergeben sich die Parameter wie folgt:

$$\boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1} \quad (\text{F.11})$$

$$\mathbf{h} = \boldsymbol{\Lambda} \boldsymbol{\mu} \quad (\text{F.12})$$

$$g = \log \eta - \frac{1}{2} \boldsymbol{\mu}^T \boldsymbol{\Lambda} \boldsymbol{\mu} \quad (\text{F.13})$$

Umgekehrt ergeben sich die Parameter zur Überführung der kanonischen Form in die Momentumform:

$$\boldsymbol{\Sigma} = \boldsymbol{\Lambda}^{-1} \quad (\text{F.14})$$

$$\boldsymbol{\mu} = \boldsymbol{\Sigma} \mathbf{h} \quad (\text{F.15})$$

F.4. Produkt zweier Normalverteilungen

Zwei normalverteilte Zufallsvariablen \mathbf{X} und \mathbf{Y} seien wie folgt in kanonischer Form gegeben:

$$\mathbf{X} \sim N(\mathbf{x}|g_X, \mathbf{h}_X, \mathbf{\Lambda}_X) \quad (\text{F.16})$$

$$\mathbf{Y} \sim N(\mathbf{y}|g_Y, \mathbf{h}_Y, \mathbf{\Lambda}_Y) \quad (\text{F.17})$$

Das Produkt dieser Zufallsvariablen $\mathbf{Z} \sim N(\mathbf{x}|g_X, \mathbf{h}_X, \mathbf{\Lambda}_X) \cdot N(\mathbf{y}|g_Y, \mathbf{h}_Y, \mathbf{\Lambda}_Y)$ ist ebenfalls normalverteilt und es gilt:

$$\mathbf{Z} \sim N(\mathbf{z}|g_Z, \mathbf{h}_Z, \mathbf{\Lambda}_Z) \quad (\text{F.18})$$

$$g_Z = g_X + g_Y \quad (\text{F.19})$$

$$\mathbf{h}_Z = \mathbf{h}_X + \mathbf{h}_Y \quad (\text{F.20})$$

$$\mathbf{\Lambda}_Z = \mathbf{\Lambda}_X + \mathbf{\Lambda}_Y \quad (\text{F.21})$$

Weitere Details zu diesen Berechnungen sind in [Bromiley, 2003] zu finden.

Abbildungsverzeichnis

1.1. Aufbau dieser Arbeit	9
2.1. Experimentierplattform Cora	20
2.2. Systemarchitektur	24
3.1. Navigationskomponenten der drei Planungsebenen	28
3.2. Adaptives dynamic Window	30
3.3. Ermittlung der Interaktionspose	36
3.4. Reaktives und taktisches Ausweichen	38
3.5. Anpassung der Roboterbewegung entsprechend des Rechtsverkehrs	40
3.6. Reaktives Guide Objective	42
3.7. Aspekte des Begleitens von Personen auf taktischer Ebene	45
4.1. Arbeiten zu Personendetektion in Laserdaten	49
4.2. Verarbeitungsschritte	52
4.3. Erscheinung von Personen in Tiefendaten	54
4.4. Merkmalsextraktion aus Tiefendaten	56
4.5. Extrahierte Merkmale am Beispiel der durchschnittlichen Distanz	57
4.6. Einsatzumgebungen	59
4.7. Evaluation mit <i>STATIC</i> Datensatz	61
4.8. Evaluation mit HOME und REHA Datensatz	63
4.9. Weitere Evaluation auf REHA Datensatz	63
4.10. Trainingsdaten	66
4.11. Arbeiten zur kamerabasierten Personendetektion mit Orientierungsschätzung	67
4.12. Grundprinzip der Referenzverfahren [Enzweiler, 2010] und [Andriluka, 2010] zur Orientierungsschätzung	70
4.13. Verarbeitungsschritte	72
4.14. Ersetzung des Hintergrundes	74

4.15. Topologie des Entscheidungsbaumes zur Personendetektion und Orientierungsschätzung	75
4.16. SVM Entscheidungsbaum zur Mehrklassenunterscheidung	76
4.17. Histogramm über absoluten Orientierungsfehler des SVM Tree	79
4.18. Beispielhafte Darstellung der Personendetektion mit Orientierungsschätzung	80
4.19. Hypothesen der einzelnen Detektoren und des Trackers	83
5.1. Arbeiten zu kamerabasierter Personendetektion	88
5.2. Einordnung der erscheinungsbasierten Optimierung	91
5.3. Erscheinungsbasiertes Modell und inverses Rendern	93
5.4. Kanten	94
5.5. Segmentierung	95
5.6. Versuchsaufbau	99
6.1. Arbeiten zur Prädiktion menschlicher Aufenthaltswahrscheinlichkeiten . .	104
6.2. Belief über zukünftigen Aufenthalt der Personen	107
6.3. Versuchsumgebung	109
6.4. Roboter- und Personentrajektorien	109
6.5. Evaluation der prädizierten Kostenfunktionen I	110
6.6. Belief Distribution Maps	112
6.7. Gegenseitiges Ausweichen durch Person und Roboter	113
6.8. Umweg des Probanden beim Ausweichen des Roboters	114
6.9. Evaluation der prädizierten Kostenfunktionen II	115
7.1. Arbeiten zur Evaluation von Navigationsverhalten	118
7.2. Krümmung von Beispieltrajektorien	121
7.3. Versuchsaufbau zur respektvollen Navigation	122
7.4. Beispieltrajektorien zur respektvollen Navigation	124
7.5. Einsatzumgebung Rehaklinik	128
7.6. Simuliertes Rechtsfahrverhalten	129
7.7. Reales Rechtsfahrverhalten	130
7.8. Rechtsfahrverhalten in der Einsatzumgebung des Lotsenroboters	131
7.9. Versuchsaufbau zum Lotsen	133
7.10. Personengeschwindigkeit und Abstand zum Roboter	134
7.11. Vorsprung, Relativgeschwindigkeit und Beschleunigung beim Lotsen . . .	136
7.12. Versuchsaufbau zum Begleiten	137
7.13. Durchschnittsgeschwindigkeit des Probanden und des Roboters	139
7.14. Geschwindigkeit und Vorsprung der Probanden	139

B.1. Kontrollparadigmen	157
D.1. Optimierungsschritte	173
D.2. Einfluss der Wichtung	174
D.3. Einfluss des Gewichtsverhältnisses der binären Klassen	175
D.4. Verrechnung der Unsicherheiten bei der Transformation zwischen Koordinatensystemen	179
E.1. Shader des Kantenmodells	184
E.2. Shader des Farbmodells	185
E.3. Universelle Matching-Funktion	187
E.4. Matching-Funktion mit personenspezifischem Farbmodell	188
F.1. Distanzbasierte Kosten $\text{cost}_{\text{PS}}^{\text{bel,d}}$	189

Literaturverzeichnis

- [Agarwal, 2006] A. Agarwal, B. Triggs, *Recovering 3D human pose from monocular images*. In IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 28(1):44–58, 2006 (Zitiert auf den Seiten 88 und 89)
- [Akanyeti, 2007] O. Akanyeti, U. Nehmzow, C. Weinrich, T. Kyriacou, S. Billings, *Programming Mobile Robots by Demonstration through System Identification*. In Proc. of European Conference on Mobile Robots (ECMR), pp. 192–197, Freiburg, Germany, 2007 (Zitiert auf Seite 12)
- [Andriluka, 2010] M. Andriluka, S. Roth, B. Schiele, *Monocular 3D Pose Estimation and Tracking by Detection*. In Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 623–630, 2010 (Zitiert auf den Seiten 67, 69, 70, 71, 88, 90 und 195)
- [Argyle, 1976] M. Argyle, M. Cook, *Gaze and mutual gaze*. Cambridge University Press, 1976 (Zitiert auf Seite 16)
- [Arras, 2008] K. Arras, S. Grzonka, M. Luber, W. Burgard, *Efficient people tracking in laser range data using a multi-hypothesis leg-tracker with adaptive occlusion probabilities*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 1710–1715, 2008 (Zitiert auf Seite 49)
- [Arras, 2007] K. Arras, O. Mozos, W. Burgard, *Using Boosted Features for the Detection of People in 2D Range Data*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 3402–3407, 2007 (Zitiert auf den Seiten 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61 und 65)
- [Azad, 2008] P. Azad, T. Asfour, R. Dillmann, *Robust real-time stereo-based markerless human motion capture*. In Proc. of IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids), pp. 700–707, 2008 (Zitiert auf den Seiten 88 und 89)
- [Bache, 2013] K. Bache, M. Lichman, *UCI Machine Learning Repository*. <http://archive.ics.uci.edu/ml>, abgerufen: 11. Dezember 2014, 2013 (Zitiert auf Seite 175)
- [Bailey, 2004] D. G. Bailey, *An efficient euclidean distance transform*. In Proc. of Int. Workshop on Combinatorial Image Analysis (IWCIA), pp. 394–408, 2004 (Zitiert auf Seite 94)
- [Bales, 1951] R. F. Bales, *Interaction Process Analysis: A Method for the Study of Small Groups*. Addison-Wesley Press, 1951 (Zitiert auf Seite 4)

- [Bastioni, 2008] M. Bastioni, S. Re, S. Misra, *Ideas and methods for modeling 3D human figures: the principal algorithms used by MakeHuman and their implementation in a new approach to parametric modeling*. In Proc of Bangalore Annual Compute Conf. (COMPUTE), pp. 1–6, 2008 (Zitiert auf Seite 92)
- [Bennewitz, 2005] M. Bennewitz, W. Burgard, G. Cielniak, S. Thrun, *Learning Motion Patterns of People for Compliant Robot Motion*. In Int. Journal of Robotics Research (IJRR), 24(1):1–30, 2005 (Zitiert auf den Seiten 39 und 104)
- [Bennewitz, 2003] M. Bennewitz, B. W., S. Thrun, *Adapting Navigation Strategies Using Motions Patterns of People*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), 2003 (Zitiert auf Seite 39)
- [Bente, 2001] G. Bente, N. C. Krämer, *Psychologische Aspekte bei der Implementierung und Evaluation nonverbal agierender Interface Agenten*. In Mensch & Computer, Teubner, 2001 (Zitiert auf Seite 5)
- [Bernardin, 2008] K. Bernardin, R. Stiefelhagen, *Evaluating multiple object tracking performance: the CLEAR MOT metrics*. In EURASIP Journal on Image and Video Processing (JIVP), 2008:1–10, 2008 (Zitiert auf Seite 99)
- [Bhattacharyya, 1946] A. Bhattacharyya, *On a Measure of Divergence between Two Multinomial Populations*. In Sankhya: The Indian Journal of Statistics (1933-1960), 7(4):401–406, 1946 (Zitiert auf Seite 95)
- [Bitgood, 2006] S. Bitgood, S. Dukes, *Not another step! Economy of movement and pedestrian choice point behavior in shopping malls*. In Environment and Behavior, 38(3):394–405, 2006 (Zitiert auf Seite 39)
- [Blanco, 2010] J.-L. Blanco, *A tutorial on SE(3) transformation parameterizations and on-manifold optimization*. Technical report, University of Malaga, 2010 (Zitiert auf Seite 178)
- [Bohlmann, 2012] K. Bohlmann, A. Beck-Greinwald, S. Buck, H. Marks, A. Zell, *Autonomous Person Following with 3D LIDAR in Outdoor Environments*. In Proc. of Int. Workshop on Perception for Mobile Robots Autonomy (PEMRA), Poznan, Poland, 2012 (Zitiert auf Seite 44)
- [Bösel, 1987] R. Bösel, *Physiologische Psychologie: Einführung in die biologischen und physiologischen Grundlagen der Psychologie. 2. Auflage*. Walter de Gruyter Verlag, Berlin, 1987 (Zitiert auf Seite 121)
- [Boucsein, 1992] W. Boucsein, *Electrodermal Activity*. Advances in Archaeological and Museum Science, Plenum Press, ISBN 9780306442148, 1992 (Zitiert auf Seite 122)
- [Boulay, 2006] B. Boulay, F. Brémond, M. Thonnat, *Applying 3D human model in a posture recognition system*. In Pattern Recognition Letters (PRL), 27(15):1788–1796, 2006 (Zitiert auf Seite 88)
- [Brandl, 2013] C. Brandl, A. Mertens, I. Blotenberg, M. Lüdtkke, T. Jacobs, C. Bröhl, M. P. Mayer, C. M. Schlick, *Acceptance of distances in human-robot interaction while the approach of a service robot with different velocities*. In Proc. of German Conf. on Ambient Assisted Living (AAL), 2013 (Zitiert auf Seite 16)

- [Brauer, 2012] J. Brauer, W. Hübner, M. Arens, *Generative 2D and 3D Human Pose Estimation with Vote Distributions*. In Proc. of Int. Symposium on Advances in Visual Computing (ISVC), pp. 470–481, 2012 (Zitiert auf Seite 89)
- [Breiman, 1984] L. Breiman, J. H. Friedman, R. A. Olshen, C. J. Stone, *Classification and Regression Trees*. Wadsworth International Group, Belmont, CA, 1984 (Zitiert auf Seite 57)
- [Bromiley, 2003] P. Bromiley, *Products and Convolutions of Gaussian Distributions*. Internal Report 2003-003, TINA Vision, 2003 (Zitiert auf Seite 193)
- [Buhmann, 1995] J. Buhmann, W. Burgard, A. B. Cremers, D. Fox, T. Hofmann, F. E. Schneider, J. Strikos, S. Thrun, *The mobile robot Rhino*. In Artificial Intelligence (AI), 16(2):31, 1995 (Zitiert auf den Seiten 16 und 21)
- [Burgard, 1998] W. Burgard, A. B. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, S. Thrun, *The Interactive Museum Tour-Guide Robot*. In Proc. of Conf. on Artificial Intelligence (AAAI), pp. 11–18, 1998 (Zitiert auf Seite 150)
- [Butler, 2001] J. T. Butler, A. Agah, *Psychological Effects of Behavior Patterns of a Mobile Personal Robot*. In Autonomous Robots (AR), 10:185–202, 2001 (Zitiert auf Seite 153)
- [Calisi, 2009] D. Calisi, D. Nardi, *Performance evaluation of pure-motion tasks for mobile robots with respect to world models*. In Autonomous Robots (AR), 27(4):465–481, 2009 (Zitiert auf den Seiten 118 und 119)
- [Carballo, 2008] A. Carballo, A. Ohya, S. Yuta, *Fusion of double layered multiple laser range finders for people detection from a mobile robot*. In Proc. of IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems (MFI), pp. 677–682, 2008 (Zitiert auf den Seiten 49 und 50)
- [Carreira-Perpinan, 2000] M. Carreira-Perpinan, *Mode-finding for mixtures of Gaussian distributions*. In IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 22(11):1318–1323, 2000 (Zitiert auf Seite 69)
- [Cassell, 1999] J. Cassell, T. Bickmore, M. Billinghurst, L. Campbell, K. Chang, H. Vilhjalmsson, H. Yan, *Embodiment in conversational interfaces*. In Proc. of ACM Conf. on Human Factors in Computing Systems (CHI), pp. 520–527, 1999 (Zitiert auf Seite 5)
- [Chang, 2010] F. Chang, C.-Y. Guo, X.-R. Lin, C.-J. Lu, *Tree Decomposition for Large-Scale SVM Problems*. In Journal of Machine Learning Research (JMLR), 11:2935–2972, 2010 (Zitiert auf den Seiten 78, 171, 175, 176 und 177)
- [Comon, 1994] P. Comon, *Independent component analysis, A new concept?* In Signal Processing, 36:287–314, 1994 (Zitiert auf Seite 69)
- [Cootes, 1998] T. Cootes, G. Edwards, C. Taylor, *Active Appearance Models*. In Proc. of European Conf. on Computer Vision (ECCV), volume 2, pp. 484–498, Springer, 1998 (Zitiert auf Seite 7)
- [Cootes, 1995] T. F. Cootes, C. J. Taylor, D. H. Cooper, J. Graham, *Active shape models: their training and application*. In Computer Vision and Image Understanding (CVIU), 61(1):38–59, 1995 (Zitiert auf den Seiten 7 und 69)

- [Cortes, 1995] C. Cortes, V. Vapnik, *Support-Vector Networks*. In Machine Learning (ML), pp. 273–297, 1995 (Zitiert auf den Seiten 50 und 69)
- [Cui, 2005] J. Cui, H. Zha, H. Zhao, R. Shibasaki, *Tracking multiple people using laser and vision*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 2116–2121, 2005 (Zitiert auf Seite 51)
- [Dalal, 2005] N. Dalal, B. Triggs, *Histograms of Oriented Gradients for Human Detection*. In Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 886–893, 2005 (Zitiert auf den Seiten 67, 68, 69, 72, 73, 74, 79, 81, 85, 92, 93 und 170)
- [Dalal, 2006] N. Dalal, B. Triggs, C. Schmid, *Human detection using oriented histograms of flow and appearance*. In Proc. of European Conf. on Computer Vision (ECCV), 2006 (Zitiert auf den Seiten 67 und 68)
- [Dautenhahn, 2004] K. Dautenhahn, *Robots we like to live with?! - a developmental perspective on a personalized, life-long robot companion*. In Proc. of Int. Workshop on Robot and Human Interactive Communication (RO-MAN), pp. 17–22, 2004 (Zitiert auf Seite 5)
- [Dautenhahn, 2006] K. Dautenhahn, M. Walters, S. Woods, K. L. Koay, C. L. Nehaniv, A. Sisbot, R. Alami, T. Siméon, *How may I serve you?: a robot companion approaching a seated person in a helping context*. In Proc. of ACM/IEEE Int. Conf. on Human-Robot Interaction (HRI), pp. 172–179, 2006 (Zitiert auf den Seiten 4 und 16)
- [Dollar, 2012] P. Dollar, C. Wojek, B. Schiele, P. Perona, *Pedestrian Detection: An Evaluation of the State of the Art*. In IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 34(4):743–761, 2012 (Zitiert auf Seite 68)
- [Dondrup, 2014] C. Dondrup, C. Lichtenthäler, M. Hanheide, *Hesitation Signals in Human-robot Head-on Encounters: A Pilot Study*. In Proceedings of the 2014 ACM/IEEE International Conference on Human-robot Interaction, HRI '14, pp. 154–155, ACM, 2014 (Zitiert auf den Seiten 118, 119, 121 und 132)
- [Droeschel, 2011] D. Droeschel, S. Behnke, *3D Body Pose Estimation Using an Adaptive Person Model for Articulated ICP*. In Proc. of Int. Conf. on Intelligent Robotics and Applications (ICIRA), pp. 157–167, 2011 (Zitiert auf den Seiten 88 und 89)
- [Einhorn, 2012] E. Einhorn, T. Langner, R. Stricker, C. Martin, H.-M. Gross, *MIRA - middleware for robotic applications*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 2591–2598, 2012 (Zitiert auf den Seiten 21 und 26)
- [Eisenbach, 2012] M. Eisenbach, A. Kolarow, K. Schenk, K. Debes, H. Gross, *View Invariant Appearance-Based Person Reidentification Using Fast Online Feature Selection and Score Level Fusion*. In Proc. of Int. Conf. on Advanced Video and Signal-Based Surveillance (AVSS), pp. 184–190, IEEE Computer Society, 2012 (Zitiert auf Seite 101)
- [Eisenbach, 2015] M. Eisenbach, A. Vorndran, S. Sorge, H.-M. Gross, *User Recognition for Guiding and Following People with a Mobile Robot in a Clinical Environment*. In IROS, submitted, 2015 (Zitiert auf den Seiten 140, 141 und 147)

- [Elhayek, 2012] A. Elhayek, C. Stoll, N. Hasler, K. Kim, H. Seidel, C. Theobalt, *Spatio-temporal motion tracking with unsynchronized cameras*. In Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 1870–1877, 2012 (Zitiert auf Seite 88)
- [Enzweiler, 2010] M. Enzweiler, D. Gavrila, *Integrated pedestrian classification and orientation estimation*. In Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 982–989, 2010 (Zitiert auf den Seiten 67, 69, 70, 71 und 195)
- [Feil-Seifer, 2005] D. Feil-Seifer, M. J. Mataric, *Defining socially assistive robotics*. In Proc. of Int. Conf. on Rehabilitation Robotics (ICORR), pp. 465–468, IEEE, 2005 (Zitiert auf Seite 4)
- [Felzenszwalb, 2008] P. Felzenszwalb, D. McAllester, D. Ramanan, *A discriminatively trained, multiscale, deformable part model*. In Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 1–8, 2008 (Zitiert auf den Seiten 67, 68, 88 und 89)
- [Felzenszwalb, 2005] P. F. Felzenszwalb, D. P. Huttenlocher, *Pictorial Structures for Object Recognition*. In Int. Journal of Computer Vision (IJCV), 61(1):55–79, 2005 (Zitiert auf Seite 69)
- [Fergus, 2003] R. Fergus, P. Perona, A. Zisserman, *Object class recognition by unsupervised scale-invariant learning*. In Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 264–271, 2003 (Zitiert auf Seite 74)
- [Ferrari, 2008] V. Ferrari, M. Marin-Jimenez, A. Zisserman, *Progressive search space reduction for human pose estimation*. In Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 1–8, 2008 (Zitiert auf den Seiten 88 und 89)
- [Fod, 2002] A. Fod, A. Howard, M. J. Mataric, *Laser-Based People Tracking*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 3024–3029, 2002 (Zitiert auf den Seiten 48 und 49)
- [Fong, 2003] T. W. Fong, I. Nourbakhsh, K. Dautenhahn, *A survey of socially interactive robots*. In Robotics and Autonomous Systems (RAS), 42:143–166, 2003 (Zitiert auf Seite 5)
- [Fox, 1997] D. Fox, W. Burgard, S. Thrun, *The dynamic window approach to collision avoidance*. In Robotics & Automation Magazine (RAM), 4(1):23–33, 1997 (Zitiert auf den Seiten 25, 29 und 30)
- [Freund, 1995] Y. Freund, R. E. Schapire, *A decision-theoretic generalization of on-line learning and an application to boosting*. In Proc. of European Conf. on Computational Learning Theory (EuroCOLT), pp. 23–37, 1995 (Zitiert auf Seite 56)
- [Fukunaga, 1975] K. Fukunaga, L. Hostetler, *The estimation of the gradient of a density function, with applications in pattern recognition*. In IEEE Transactions on Information Theory (TIT), 21(1):32–40, 1975 (Zitiert auf Seite 79)
- [Gandhi, 2008] T. Gandhi, M. M. Trivedi, *Image based estimation of pedestrian orientation for improving path prediction*. In Proc. of IEEE Intelligent Vehicles Symposium (IVS), pp. 506–511, IEEE, 2008 (Zitiert auf den Seiten 67 und 69)
- [Gavrila, 2007] D. Gavrila, *A Bayesian, Exemplar-Based Approach to Hierarchical Shape Matching*. In IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 29(8):1408–1421, 2007 (Zitiert auf den Seiten 67 und 68)

- [Germa, 2009] T. Germa, F. Lerasle, N. Ouadah, V. Cadenat, M. Devy, *Vision and RFID-based person tracking in crowds from a mobile robot*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 5591–5596, 2009 (Zitiert auf den Seiten 22 und 153)
- [Gharpure, 2008] C. Gharpure, V. Kulyukin, *Robot-assisted shopping for the blind: issues in spatial cognition and product selection*. In Intelligent Service Robotics (ISR), 1(3):237–251, 2008 (Zitiert auf Seite 152)
- [Glas, 2007] D. Glas, T. Miyashita, H. Ishiguro, N. Hagita, *Laser tracking of human body motion using adaptive shape modeling*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 602–608, 2007 (Zitiert auf den Seiten 48 und 49)
- [Go, 2013] A. S. Go, D. Mozaffarian, V. L. Roger, E. J. Benjamin, J. D. Berry, W. B. Borden, D. M. Bravata, S. Dai, E. S. Ford, C. S. Fox, ..., *Heart disease and stroke statistics-2013 update a report from the American Heart Association*. In Circulation, 127(1):e6–e245, 2013 (Zitiert auf Seite 14)
- [Gockley, 2007] R. Gockley, J. Forlizzi, R. Simmons, *Natural person-following behavior for social robots*. In Proc. of ACM/IEEE Int. Conf. on Human-Robot Interaction (HRI), pp. 17–24, 2007 (Zitiert auf den Seiten 43 und 154)
- [Goffman, 1963] E. Goffman, *Stigma*. Prentice-Hall, Englewood Cliffs, New Jersey, 1963 (Zitiert auf Seite 16)
- [Goffman, 1971] E. Goffman, *Verhalten in sozialen Situationen - Strukturen und Regeln der Interaktion im öffentlichen Raum*. Fachverlag Bertelsmann, Gütersloh, 1971 (Zitiert auf Seite 16)
- [Göller, 2014] M. Göller, *Behavior-based Control for Service Robots inspired by Human Motion Patterns: a Robotic Shopping Assistant*. Ph.d. thesis, Karlsruhe Institute of Technology, 2014 (Zitiert auf Seite 159)
- [Göller, 2009] M. Göller, T. Kerscher, J. Zöllner, R. Dillmann, M. Devy, T. Germa, F. Lerasle, *Setup and control architecture for an interactive Shopping Cart in human all day environments*. In Proc. of Int. Conf. on Advanced Robotics (ICAR), pp. 1–6, 2009 (Zitiert auf Seite 153)
- [Göller, 2010] M. Göller, F. Steinhardt, T. Kerscher, J. Marius Zollner, R. Dillmann, *Proactive avoidance of moving obstacles for a service robot utilizing a behavior-based control*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 5984–5989, 2010 (Zitiert auf den Seiten 39 und 153)
- [Goodrich, 2007] M. A. Goodrich, A. C. Schultz, *Human-robot interaction: a survey*. In Foundations and Trends in Human-Computer Interaction (FTHCI), 1(3):203–275, 2007 (Zitiert auf Seite 3)
- [Goodwin, 1981] C. Goodwin, *Conversational organization: Interaction between speakers and hearers*. Academic Press, New York, 1981 (Zitiert auf Seite 16)
- [Gross, 2009] H.-M. Gross, H.-J. Böhme, C. Schröter, S. Müller, A. König, E. Einhorn, C. Martin, M. Merten, A. Bley, *TOOMAS: Interactive Shopping Guide Robots in Everyday Use - Final Implementation and Experiences from Long-term Field Trials*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 2005–2012, 2009 (Zitiert auf den Seiten 6 und 151)

- [Gross, 2014] H.-M. Gross, K. Debes, E. Einhorn, S. Mueller, A. Scheidig, C. Weinrich, A. Bley, C. Martin, *Mobile Robotic Rehabilitation Assistant for Walking and Orientation Training of Stroke Patients: A Report on Work in Progress*. In Proc. of IEEE Int. Conf. on Systems, Man, and Cybernetics (SMC), pp. 1880–1887, 2014 (Zitiert auf den Seiten 4, 12, 15, 28 und 148)
- [Gu, 2009] C. Gu, J. Lim, P. Arbelaez, J. Malik, *Recognition using regions*. In Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 1030–1037, 2009 (Zitiert auf den Seiten 67 und 68)
- [Guyon, 2002] I. Guyon, J. Weston, S. Barnhill, V. Vapnik, *Gene Selection for Cancer Classification using Support Vector Machines*. In Machine Learning (ML), 46:389–422, 2002 (Zitiert auf Seite 174)
- [Hall, 1963] E. T. Hall, *A System for the Notation of Proxemic Behavior*. In American Anthropologist, 65(5):1003–1026, 1963 (Zitiert auf den Seiten 5, 6, 16, 17, 36, 103 und 120)
- [Hart, 1968] P. Hart, N. Nilsson, B. Raphael, *A Formal Basis for the Heuristic Determination of Minimum Cost Paths*. In IEEE Transactions on Systems Science and Cybernetics, 4(2):100–107, 1968 (Zitiert auf Seite 33)
- [Helbing, 1995] D. Helbing, P. Molnár, *Social force model for pedestrian dynamics*. In Physical Review E (PRE), 51(5):4282–4286, 1995 (Zitiert auf den Seiten 104 und 105)
- [Henry, 2010] P. Henry, C. Vollmer, B. Ferris, D. Fox, *Learning to navigate through crowded environments*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 981–986, 2010 (Zitiert auf den Seiten 104 und 105)
- [Hofmann, 2012] M. Hofmann, D. Gavrilu, *Multi-view 3D Human Pose Estimation in Complex Environment*. In Int. Journal of Computer Vision (IJCV), 96(1):103–124, 2012 (Zitiert auf Seite 88)
- [Holthaus, 2010] P. Holthaus, I. Lütkebohle, M. Hanheide, S. Wachsmuth, *Can I help you?: a spatial attention system for a receptionist robot*. In Proceedings of the 2th international conference on Social robotics, ICSR'10, pp. 325–334, Springer-Verlag, Berlin, Heidelberg, 2010 (Zitiert auf Seite 16)
- [Hüttenrauch, 2006] H. Hüttenrauch, K. Eklundh, A. Green, E. Topp, *Investigating Spatial Relationships in Human-Robot Interaction*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 5052–5059, 2006 (Zitiert auf den Seiten 5, 16, 17, 103 und 153)
- [ISO 8373, 2012] ISO 8373, *Robots and robotic devices – Vocabulary*. ISO 8373:2012, International Organization for Standardization, Geneva, Switzerland, 2012 (Zitiert auf Seite 3)
- [Jensen, 2005] B. Jensen, N. Tomatis, L. Mayor, A. Drygajlo, R. Siegwart, *Robots meet Humans-interaction in public spaces*. In IEEE Transactions on Industrial Electronics (TIE), 52(6):1530–1546, 2005 (Zitiert auf Seite 151)
- [Julier, 1997] S. Julier, J. Uhlmann, *A non-divergent estimation algorithm in the presence of unknown correlations*. In Proc. of American Control Conference (ACC), volume 4, pp. 2369–2373 vol.4, 1997 (Zitiert auf Seite 58)
- [Kanda, 2009] T. Kanda, D. Glas, M. Shiomi, N. Hagita, *Abstracting Peoples Trajectories for Social Robots to Proactively Approach Customers*. In IEEE Transactions on Robotics (TRA), 25(6):1382–1396, 2009 (Zitiert auf Seite 22)

- [Kennedy, 1995] J. Kennedy, R. Eberhart, *Particle swarm optimization*. In Proc. of IEEE Int. Conf. on Neural Networks (ICNN), volume 4, pp. 1942–1948, 1995 (Zitiert auf den Seiten 35, 91, 97 und 161)
- [Kessler, 2012] J. Kessler, D. Iser, H. Gross, *I'll keep you in sight: Finding a good position to observe a person*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 4392–4398, 2012 (Zitiert auf Seite 7)
- [Kleinehagenbrock, 2002] M. Kleinehagenbrock, S. Lang, J. Fritsch, F. Lömker, G. A. Fink, G. Sagerer, *Person Tracking with a Mobile Robot based on Multi-Modal Anchoring*. In Proc. of Int. Workshop on Robot and Human Interactive Communication (RO-MAN), pp. 423–429, 2002 (Zitiert auf den Seiten 49 und 50)
- [Kluge, 2001] B. Kluge, C. Kohler, E. Prassler, *Fast and robust tracking of multiple moving objects with a laser range finder*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), volume 2, pp. 1683–1688 vol.2, 2001 (Zitiert auf Seite 49)
- [Kolominsky-Rabas, 2006] P. L. Kolominsky-Rabas, P. U. Heuschmann, D. Marschall, M. Emmert, N. Baltzer, B. Neundörfer, O. Schöffski, K. J. Krobot, *Lifetime cost of ischemic stroke in germany: Results and national projections from a population-based stroke registry the erlangen stroke project*. In Stroke, 37(5):1179–1183, 2006 (Zitiert auf Seite 14)
- [Kreßel, 1999] U. H.-G. Kreßel, *Pairwise classification and support vector machines*, pp. 255–268. MIT Press, Cambridge, MA, USA, 1999 (Zitiert auf den Seiten 71 und 170)
- [Kruse, 2014] T. Kruse, A. Kirsch, H. Khambhaita, R. Alami, *Evaluating directional cost models in navigation*. In Proc. of ACM/IEEE Int. Conf. on Human-Robot Interaction (HRI), pp. 350–357, ACM, 2014 (Zitiert auf Seite 118)
- [Kuderer, 2012] M. Kuderer, H. Kretzschmar, C. Sprunk, W. Burgard, *Feature-Based Prediction of Trajectories for Socially Compliant Navigation*. In Proc. of Int. Conf. on Robotics: Science and Systems (RSS), 2012 (Zitiert auf den Seiten 104 und 105)
- [Lawrence, 2007] N. D. Lawrence, A. J. Moore, *Hierarchical Gaussian process latent variable models*. In Proc. of Int. Conf. on Machine Learning (ICML), pp. 481–488, ACM, 2007 (Zitiert auf Seite 90)
- [Leibe, 2005] B. Leibe, E. Seemann, B. Schiele, *Pedestrian detection in crowded scenes*. In Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR), volume 1, pp. 878–885 vol. 1, 2005 (Zitiert auf den Seiten 67 und 68)
- [Lu, 2013] D. V. Lu, D. B. Allan, W. D. Smart, *Tuning Cost Functions for Social Navigation*. In Proc. of Int. Conf. on Social Robotic (ICSR), International Conference on Social Robotics, 2013 (Zitiert auf Seite 164)
- [Luber, 2012] M. Luber, L. Spinello, J. Silva, K. O. Arras, *Socially Acceptable Robot Navigation: A Learning Approach*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 902–907, 2012 (Zitiert auf den Seiten 104 und 105)
- [MacQueen, 1967] J. B. MacQueen, *Some Methods for Classification and Analysis of MultiVariate Observations*. In Proc. of Berkeley Symposium on Mathematical Statistics and Probability (BSMSP), volume 1, pp. 281–297, 1967 (Zitiert auf Seite 107)

- [Martin, 2012] C. Martin, *Videobasierte Verfahren zur Schätzung des Interaktionsinteresses bei der Mensch-Roboter-Interaktion mittels Analyse durch Synthese*. Ph.D. thesis, Ilmenau University of Technology, 2012 (Zitiert auf den Seiten 7, 67 und 69)
- [Matarić, 2008] M. Matarić, F. Michaud, *Behavior-Based Systems*. In B. Siciliano, O. Khatib, editors, Springer Handbook of Robotics, pp. 891–909, Springer Berlin Heidelberg, 2008 (Zitiert auf Seite 158)
- [Mitzel, 2012] D. Mitzel, B. Leibe, *Taking Mobile Multi-object Tracking to the Next Level: People, Unknown Objects, and Carried Items*. In Proc. of European Conf. on Computer Vision (ECCV), pp. 566–579, 2012 (Zitiert auf den Seiten 88 und 89)
- [Montemerlo, 2002] M. Montemerlo, J. Pineau, N. Roy, S. Thrun, V. Verma, *Experiences with a mobile robotic guide for the elderly*. In Proc. of Conf. on Artificial Intelligence (AAAI), pp. 587–592, 2002 (Zitiert auf Seite 150)
- [Moon, 2011] A. Moon, C. Parker, E. Croft, H. Van der Loos, *Did you see it hesitate? - Empirically grounded design of hesitation trajectories for collaborative robots*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 1994–1999, 2011 (Zitiert auf Seite 119)
- [Morales, 2014] Y. Morales, T. Kanda, N. Hagita, *Walking Together: Side by Side Walking Model for an Interacting Robot*. In Proc. of ACM/IEEE Int. Conf. on Human-Robot Interaction (HRI), 3(2):51–73, 2014 (Zitiert auf den Seiten 43 und 152)
- [Mozos, 2009] O. M. Mozos, R. Kurazume, T. Hasegawa, *Multi-Layer People Detection Using 2D Range Data*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA) Workshop: People Detection and Tracking, 2009 (Zitiert auf den Seiten 49 und 50)
- [Müller, 2008] S. Müller, S. Hellbach, E. Schaffernicht, A. Ober, A. Scheidig, H.-M. Gross, *Whom to talk to? Estimating user interest from movement trajectories*. In Proc. of Int. Symposium on Robot and Human Interactive Communication (RO-MAN), pp. 532–538, 2008 (Zitiert auf Seite 151)
- [Müller, 2007] S. Müller, E. Schaffernicht, A. Scheidig, H.-J. Böhme, H.-M. Gross, *Are you still following me?* In Proc. of European Conference on Mobile Robots (ECMR), pp. 211–216, 2007 (Zitiert auf Seite 151)
- [Muñoz Ceballos, 2010] N. D. Muñoz Ceballos, J. A. Valencia, N. L. Ospina, *Quantitative Performance Metrics for Mobile Robots Navigation*. INTECH Open Access Publisher, 2010 (Zitiert auf den Seiten 118, 119 und 120)
- [Murphy, 2000] R. R. Murphy, *Introduction to AI Robotics*. MIT Press, Cambridge, MA, USA, 1st edition, 2000 (Zitiert auf den Seiten 155 und 157)
- [Mutlu, 2009] B. Mutlu, F. Yamaoka, T. Kanda, H. Ishiguro, N. Hagita, *Nonverbal leakage in robots: communication of intentions through seemingly unintentional behavior*. In Proc. of ACM/IEEE Int. Conf. on Human-Robot Interaction (HRI), pp. 69–76, 2009 (Zitiert auf den Seiten 16 und 152)
- [Navarro-Serment, 2010] L. E. Navarro-Serment, C. Mertz, M. Hebert, *Pedestrian Detection and Tracking Using Three-dimensional LADAR Data*. In Int. Journal of Robotics Research (IJRR), 29(12):1516 – 1528, 2010 (Zitiert auf den Seiten 49 und 50)

- [Nehmzow, 2007a] U. Nehmzow, O. Akanyeti, C. Weinrich, T. Kyriacou, S. Billings, *Learning by observation through system identification*. In Proc. Towards Autonomous Robotic Systems (TAROS), pp. 17–24, 2007 (Zitiert auf Seite 12)
- [Nehmzow, 2007b] U. Nehmzow, O. Akanyeti, C. Weinrich, T. Kyriacou, S. Billings, *Robot programming by demonstration through system identification*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 801–806, 2007 (Zitiert auf Seite 12)
- [Olsen, 2003] D. R. Olsen, M. A. Goodrich, *Metrics for evaluating human-robot interactions*. In Proc. of Workshop on Performance Metrics for Intelligent Systems, volume 2003, p. 5, 2003 (Zitiert auf den Seiten 118 und 119)
- [OpenNI, 2011] OpenNI, *OpenNI User Guide*. Website, available online at <http://www.openni.org/documentation> last visited on August 10th 2011, 2011 (Zitiert auf den Seiten 73, 81 und 146)
- [Osman, 2007] H. Osman, *Novel Multiclass SVM-Based Binary Decision Tree Classifier*. In Proc. of IEEE Int. Symposium on Signal Processing and Information Technology (ISSPIT), pp. 880–883, 2007 (Zitiert auf den Seiten 70, 71 und 172)
- [Pacchierotti, 2005] E. Pacchierotti, H. Christensen, P. Jensfelt, *Human-robot embodied interaction in hallway settings: a pilot user study*. In Proc. of Int. Workshop on Robot and Human Interactive Communication (RO-MAN), pp. 164–171, 2005 (Zitiert auf den Seiten 65 und 154)
- [Pacchierotti, 2006] E. Pacchierotti, H. Christensen, P. Jensfelt, *Evaluation of Passing Distance for Social Robots*. In Proc. of Int. Symposium on Robot and Human Interactive Communication (RO-MAN), pp. 315–320, 2006 (Zitiert auf Seite 17)
- [Pearson, 1901] K. Pearson, *On lines and planes of closest fit to systems of points in space*. In Philosophical Magazine (PM), 2(6):559–572, 1901 (Zitiert auf den Seiten 50 und 69)
- [Philippsen, 2005] R. Philippsen, R. Siegart, *An Interpolated Dynamic Navigation Function*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 3793–3800, 2005 (Zitiert auf den Seiten 33 und 164)
- [Pineau, 2003] J. Pineau, M. Montemerlo, M. Pollack, N. Roy, S. Thrun, *Towards robotic assistants in nursing homes: Challenges and results*. In Robotics and Autonomous Systems (RAS), 42(1):271–281, 2003 (Zitiert auf Seite 150)
- [Platt, 2000] J. Platt, N. Cristianini, J. Shawe-Taylor, *Large Margin DAGS for Multiclass Classification*. In Proc. of Int. Conf. on Advances in Neural Information Processing Systems (NIPS), pp. 547–553, 2000 (Zitiert auf den Seiten 71 und 170)
- [Pöschl, 2008] S. Pöschl, N. Döring, H.-J. Böhme, C. Martin, *Computergestützte Artikelsuche im Baumarkt. Formative Evaluation eines Artikelsuchsystems für mobile Shopping-Roboter*. In Zeitschrift für Evaluation, 7(1):113–135, 2008 (Zitiert auf Seite 151)
- [Pöschl, 2009] S. Pöschl, N. Döring, H.-J. Böhme, C. Martin, *Mensch-Roboter-Interaktion im Baumarkt-Formative Evaluation eines mobilen Shopping-Roboters*. In Zeitschrift für Evaluation, 8(1):27–58, 2009 (Zitiert auf den Seiten 6 und 151)

- [Pöschl, 2011] S. Pöschl, N. Döring, H.-M. Gross, A. Bley, C. Martin, H.-J. Böhme, *Roboter-gestützte Artikelsuche im Baumarkt. Eine Studie zu summativer Evaluation und Nutzerakzeptanz eines mobilen Shopping-Assistenten*. In *Zeitschrift für Evaluation*, 10(1):99–132, 2011 (Zitiert auf Seite 151)
- [Quinlan, 1986] J. R. Quinlan, *Induction of Decision Trees*. In *Machine Learning (ML)*, 1:81–106, 1986 (Zitiert auf den Seiten 77, 78, 172, 176 und 177)
- [Ramanan, 2005] D. Ramanan, D. Forsyth, A. Zisserman, *Strike a pose: tracking people by finding stylized poses*. In *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pp. 271–278 vol. 1, 2005 (Zitiert auf den Seiten 88 und 89)
- [Rother, 2004] C. Rother, V. Kolmogorov, A. Blake, *GrabCut: Interactive Foreground Extraction using Iterated Graph Cuts*. In *ACM Transactions on Graphics (TOG)*, 23:309–314, 2004 (Zitiert auf Seite 74)
- [Rybok, 2010] L. Rybok, M. Voit, H. Ekenel, R. Stiefelhagen, *Multi-view Based Estimation of Human Upper-Body Orientation*. In *Proc. of Int. Conf. on Pattern Recognition (ICPR)*, pp. 1558–1561, 2010 (Zitiert auf Seite 88)
- [Sardar, 2012] A. Sardar, M. Joosse, A. Weiss, V. Evers, *Don't stand so close to me: Users' attitudinal and behavioral responses to personal space invasion by a robot*. In *Proc. of ACM/IEEE Int. Conf. on Human-Robot Interaction (HRI)*, pp. 229–230, 2012 (Zitiert auf den Seiten 4, 16, 17 und 36)
- [Satake, 2009] S. Satake, T. Kanda, D. F. Glas, M. Imai, H. Ishiguro, N. Hagita, *How to approach humans?: Strategies for Social Robots to Initiate Interaction*. In *Proc. of ACM/IEEE Int. Conf. on Human-Robot Interaction (HRI)*, pp. 109–116, 2009 (Zitiert auf den Seiten 5, 104 und 152)
- [Schallert, 2003] T. Schallert, S. Fleming, M. Woodlee, *Should the injured and intact hemispheres be treated differently during the early phases of physical restorative therapy in experimental stroke or parkinsonism?* In *Physical Medicine & Rehabilitation Clinics of North America (PMRCNA)*, 14(1):27–46, 2003 (Zitiert auf Seite 14)
- [Scheidig, 2015a] A. Scheidig, K. Debes, S. Müller, C. Schröter, M. Volkhardt, H.-M. Gross, K. Richter, N. Döring, *SERROGA: Funktions-und Nutzertests Herangehensweise und Ergebnisse*. In *Proc. of German Conf. on Ambient Assisted Living (AAL)*, pp. 34–43, VDE VERLAG GmbH, 2015 (Zitiert auf Seite 7)
- [Scheidig, 2015b] A. Scheidig, E. Einhorn, C. Weinrich, M. Eisenbach, S. Müller, T. Schmiedel, T. Wengefeld, T. Trinh, H.-M. Gross, A. Bley, R. Scheidig, G. Pfeiffer, S. Meyer, S. Oelkers, *Robotischer Reha-Assistent zum Lauftraining von Patienten nach Schlaganfall: Erste Ergebnisse zum Laufcoach*. In *Proc. of German Conf. on Ambient Assisted Living (AAL)*, 2015 (Zitiert auf den Seiten 12, 127 und 130)
- [Scheidig, 2014] A. Scheidig, C. Schröter, M. Volkhardt, S. Müller, K. Debes, H.-M. Gross, N. Döring, K. Richter, *SERROGA: Servicerobotik für die Gesundheitsassistenten im nutzerzentrierten Entwurf*. In *Proc. of German Conf. on Ambient Assisted Living (AAL)*, VDE VERLAG GmbH, 2014 (Zitiert auf Seite 7)
- [Schenk, 2011] K. Schenk, M. Eisenbach, A. Kolarow, H.-M. Gross, *Comparison of laser-based person tracking at feet and upper-body height*. In *Proc. of German Conf. on Advances in Artificial Intelligence (KI)*, pp. 277–288, 2011 (Zitiert auf den Seiten 81, 98, 114 und 119)

- [Scheutz, 2004] M. Scheutz, J. McRaven, G. Cserey, *Fast, reliable, adaptive, bimodal people tracking for indoor environments*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 1347–1352, IEEE, 2004 (Zitiert auf Seite 51)
- [Schulz, 2001] D. Schulz, W. Burgard, D. Fox, A. Cremers, *Tracking multiple moving targets with a mobile robot using particle filters and statistical data association*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), volume 2, pp. 1665–1670 vol.2, 2001 (Zitiert auf Seite 49)
- [Schulz, 2003] D. Schulz, W. Burgard, D. Fox, A. B. Cremers, *People Tracking with Mobile Robots Using Sample-based Joint Probabilistic Data Association Filters*. In Int. Journal of Robotics Research (IJRR), 22(2):99–116, 2003 (Zitiert auf Seite 49)
- [Seidl, 2015] B. Seidl, *Framework zur Evaluation des Navigationsverhaltens mobiler Roboter gegenüber Menschen*. Master thesis, Ilmenau University of Technology, Germany, 2015 (Zitiert auf den Seiten 120, 121, 122, 123, 124, 134, 136 und 139)
- [Shi, 2008] D. Shi, J. Collins, E.G., B. Goldiez, A. Donate, X. Liu, D. Dunlap, *Human-aware robot motion planning with velocity constraints*. In Proc. of Int. Symposium on Collaborative Technologies and Systems (CTS), pp. 490–497, 2008 (Zitiert auf den Seiten 16 und 37)
- [Shiomi, 2009] M. Shiomi, T. Kanda, D. F. Glas, S. Satake, H. Ishiguro, N. Hagita, *Field trial of networked social robots in a shopping mall*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 2846–2853, 2009 (Zitiert auf Seite 152)
- [Shotton, 2011] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, A. Blake, *Real-time human pose recognition in parts from single depth images*. In Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 1297–1304, 2011 (Zitiert auf den Seiten 88 und 89)
- [Siegwart, 2003] R. Siegwart, K. O. Arras, S. Bouabdallah, D. Burnier, G. Froidevaux, X. Greppin, B. Jensen, A. Lorotte, L. Mayor, M. Meisser, R. Philippsen, R. Piguet, G. Ramel, G. Terrien, N. Tomatis, *Robox at Expo.02: A large-scale installation of personal robots*. In Robotics and Autonomous Systems (RAS), 42(3-4):203–222, 2003 (Zitiert auf Seite 151)
- [Sigal, 2012] L. Sigal, M. Isard, H. Haussecker, M. Black, *Loose-limbed People: Estimating 3D Human Pose and Motion Using Non-parametric Belief Propagation*. In Int. Journal of Computer Vision (IJCV), 98:15–48, 2012 (Zitiert auf Seite 88)
- [Song, 2010] X. Song, H. Zhao, J. Cui, X. Shao, R. Shibasaki, H. Zha, *Fusion of laser and vision for multiple targets tracking via on-line learning*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 406–411, 2010 (Zitiert auf den Seiten 49 und 51)
- [Spinello, 2010] L. Spinello, K. O. Arras, R. Triebel, R. Siegwart, *A Layered Approach to People Detection in 3D Range Data*. In Proc. of Conf. on Artificial Intelligence (AAAI), 2010 (Zitiert auf den Seiten 49 und 50)
- [Spinello, 2008] L. Spinello, R. Siegwart, *Human detection using multimodal and multidimensional features*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 3264–3269, 2008 (Zitiert auf den Seiten 49, 51, 53, 56, 57, 58, 60 und 61)

- [Steinfeld, 2006] A. Steinfeld, T. Fong, D. Kaber, M. Lewis, J. Scholtz, A. Schultz, M. Goodrich, *Common Metrics for Human-robot Interaction*. In Proc. of ACM/IEEE Int. Conf. on Human-Robot Interaction (HRI), HRI '06, pp. 33–40, ACM, New York, NY, USA, ISBN 1-59593-294-1, 2006 (Zitiert auf den Seiten 118 und 119)
- [Stentz, 1994] A. Stentz, *Optimal and efficient path planning for partially-known environments*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 3310–3317 vol.4, 1994 (Zitiert auf Seite 33)
- [Stricker, 2012] R. Stricker, S. Müller, E. Einhorn, C. Schröter, M. Volkhardt, K. Debes, H.-M. Gross, *Interactive mobile robots guiding visitors in a university building*. In Proc. of Int. Symposium on Robot and Human Interactive Communication (RO-MAN), pp. 695–700, 2012 (Zitiert auf Seite 13)
- [Syrdal, 2007] D. Syrdal, K. L. Koay, M. Walters, K. Dautenhahn, *A personalized robot companion? - The role of individual differences on spatial preferences in HRI scenarios*. In Proc. of Int. Symposium on Robot and Human Interactive Communication (RO-MAN), pp. 1143–1148, 2007 (Zitiert auf Seite 154)
- [Syrdal, 2006] D. S. Syrdal, K. Dautenhahn, S. Woods, M. Walters, K. Koay, *Doing the Right Thing Wrong - Personality and Tolerance to Uncomfortable Robot Approaches*. In Proceedings of the 15th IEEE International Symposium on Robot and Human Interactive Communication, pp. 183–188, Hertfordshire, 2006 (Zitiert auf Seite 5)
- [Takayama, 2009] L. Takayama, C. Pantofaru, *Influences on proxemic behaviors in human-robot interaction*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 5495–5502, 2009 (Zitiert auf Seite 17)
- [Teasell, 2005] R. W. Teasell, L. Kalra, *What's New in Stroke Rehabilitation Back to Basics*. In Stroke, 36(2):215–217, 2005 (Zitiert auf Seite 14)
- [Thompson, 2009] S. Thompson, T. Horiuchi, S. Kagami, *A probabilistic model of human motion and navigation intent for mobile robot path planning*. In Proc. of Int. Conf. on Autonomous Robots and Agents (ICARA), pp. 663–668, 2009 (Zitiert auf Seite 104)
- [Thrun, 1999a] S. Thrun, M. Bennewitz, W. Burgard, A. Cremers, F. Dellaert, D. Fox, D. Haehnel, G. Lakemeyer, C. Rosenberg, N. Roy, J. Schulte, D. Schulz, W. Steiner, *Experiences with two deployed interactive tour-guide robots*. In Proc. of Int. Conf. on Field and Service Robotics (FSR), 1999 (Zitiert auf Seite 150)
- [Thrun, 1999b] S. Thrun, M. Bennewitz, W. Burgard, A. Cremers, F. Dellaert, D. Fox, D. Hahnel, C. Rosenberg, N. Roy, J. Schulte, D. Schulz, *MINERVA: a second-generation museum tour-guide robot*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), volume 3, pp. 1999–2005 vol.3, 1999 (Zitiert auf Seite 150)
- [Thrun, 2005] S. Thrun, W. Burgard, D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005 (Zitiert auf Seite 7)
- [Topp, 2005] E. Topp, H. Christensen, *Tracking for following and passing persons*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 2321–2327, 2005 (Zitiert auf Seite 51)

- [Trautman, 2010] P. Trautman, A. Krause, *Unfreezing the robot: Navigation in dense, interacting crowds*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 797–803, 2010 (Zitiert auf den Seiten 39, 104 und 105)
- [Vapnik, 1998] V. N. Vapnik, *Statistical learning theory*. Wiley, 1 edition, 1998 (Zitiert auf den Seiten 71, 78, 170, 176 und 177)
- [Voit, 2009] M. Voit, R. Stiefelhagen, *A System for Probabilistic Joint 3D Head Tracking and Pose Estimation in Low-Resolution, Multi-view Environments*. In Proc. of Int. Conf. on Computer Vision Systems (ICVS), pp. 415–424, 2009 (Zitiert auf Seite 88)
- [Volkhardt, 2013a] M. Volkhardt, C. Weinrich, H.-M. Gross, *Multi-modal people tracking on a mobile companion robot*. In Proc. of European Conference on Mobile Robots (ECMR), pp. 288–293, 2013 (Zitiert auf Seite 11)
- [Volkhardt, 2013b] M. Volkhardt, C. Weinrich, H.-M. Gross, *People Tracking on a Mobile Companion Robot*. In Proc. of IEEE Int. Conf. on Systems, Man, and Cybernetics (SMC), pp. 4354–4359, 2013 (Zitiert auf den Seiten 11 und 82)
- [Volkhardt, 2009] M. Volkhardt, C. Weinrich, C. Schröter, H.-M. Gross, *A Concept for Detection and Tracking of People in Smart Home Environments with a Mobile Robot*. In 2nd Companion-Able Workshop co-located with the 3rd European Conf. on Ambient Intelligence (AmI), AmI '09, Springer-Verlag, Berlin, Heidelberg, 2009 (Zitiert auf Seite 11)
- [Walters, 2011] M. Walters, M. Oskoei, D. Syrdal, K. Dautenhahn, *A long-term Human-Robot Proxemic study*. In Proc. of Int. Symposium on Robot and Human Interactive Communication (RO-MAN), pp. 137–142, 2011 (Zitiert auf den Seiten 17, 103 und 106)
- [Walters, 2008] M. Walters, D. Syrdal, K. Koay, K. Dautenhahn, R. te Boekhorst, *Human approach distances to a mechanical-looking robot with different robot voice styles*. In Proc. of Int. Symposium on Robot and Human Interactive Communication (RO-MAN), pp. 707–712, 2008 (Zitiert auf Seite 154)
- [Wang, 2009] X. Wang, T. X. Han, S. Yan, *An HOG-LBP human detector with partial occlusion handling*. In Proc. of IEEE Int. Conf. on Computer Vision (ICCV), pp. 32–39, 2009 (Zitiert auf den Seiten 67, 68 und 170)
- [Weinrich, 2011] C. Weinrich, Y. Ludewig, K. Debes, N. Döring, H.-M. Gross, *OMERON: Optimierung der Mensch-Roboter-Interaktion durch sozialverträgliche Navigation bei mobilen Servicerobotern: Konzeption, Implementierung und Evaluation*. (unveröffentlicht), Forschungsantrag bei der Deutschen Forschungsgesellschaft (DFG), 2011 (Zitiert auf den Seiten 16 und 150)
- [Weinrich, 2010] C. Weinrich, S. Müller, H.-M. Gross, *Appearance-based Person Tracking and 3D Pose Estimation of Upper-Body and Head*. In Proc. of Int. Scientific Colloquium (IWK), pp. 606–611, Ilmenau, Germany, 2010 (Zitiert auf den Seiten 10 und 11)
- [Weinrich, 2013a] C. Weinrich, M. Volkhardt, E. Einhorn, H.-M. Gross, *Prediction of Human Avoidance Behavior by Lifelong Learning for Socially Compliant Robot Navigation*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 376–381, Karlsruhe (Germany), 2013 (Zitiert auf den Seiten 11, 104, 109 und 119)

- [Weinrich, 2013b] C. Weinrich, M. Volkhardt, H.-M. Gross, *Appearance-Based 3D Upper-Body Pose Estimation and Person Recognition on Mobile Robots*. In Proc. of IEEE Int. Conf. on Systems, Man, and Cybernetics (SMC), pp. 4384–4390, Manchester (UK), 2013 (Zitiert auf den Seiten 11 und 88)
- [Weinrich, 2012] C. Weinrich, C. Vollmer, H.-M. Gross, *Estimation of Human Upper Body Orientation for Mobile Robotics using an SVM Decision Tree on Monocular Images*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 2147–2152, Vilamoura (Portugal), 2012 (Zitiert auf den Seiten 10, 66, 67, 78, 81, 176 und 177)
- [Weinrich, 2014a] C. Weinrich, T. Wengefeld, C. Schröter, H.-M. Gross, *People Detection and Distinction of their Walking Aids in 2D Laser Range Data based on Generic Distance-Invariant Features*. In Proc. of Int. Symposium on Robot and Human Interactive Communication (RO-MAN), pp. 767–773, Edinburgh (UK), 2014 (Zitiert auf den Seiten 10, 49, 51, 54, 55, 56, 57, 59, 61 und 63)
- [Weinrich, 2014b] C. Weinrich, T. Wengefeld, M. Volkhardt, A. Scheidig, H.-M. Gross, *Generic Distance-Invariant Features for Detecting People with Walking Aid in 2D Laser Range Data*. In Proc. of Int. Conf. on Intelligent Autonomous Systems (IAS), Padova, Italy, 2014 (Zitiert auf Seite 10)
- [Wojek, 2008] C. Wojek, B. Schiele, *A Performance Evaluation of Single and Multi-feature People Detection*. In Proc. of DAGM Symposium on Pattern Recognition, pp. 82–91, Springer-Verlag, Berlin, Heidelberg, 2008 (Zitiert auf Seite 68)
- [Wu, 2011] J. Wu, C. Geyer, J. M. Rehg, *Real-Time Human Detection Using Contour Cues*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), 2011 (Zitiert auf den Seiten 67 und 68)
- [Xavier, 2005] J. Xavier, M. Pacheco, D. Castro, A. Ruano, U. Nunes, *Fast Line, Arc/Circle and Leg Detection from Laser Scan Data in a Player Driver*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 3930–3935, 2005 (Zitiert auf den Seiten 49 und 50)
- [Yamazaki, 2007] K. Yamazaki, M. Kawashima, Y. Kuno, N. Akiya, M. Burdelski, A. Yamazaki, H. Kuzuoka, *Prior-to-request and request behaviours within elderly day care: Implications for developing service robots for use in multiparty settings*. In Proceedings of the Tenth European Conference on Computer Supported Cooperative Work, pp. 61–78, 2007 (Zitiert auf Seite 16)
- [Zhao, 2005] H. Zhao, R. Shibasaki, *A novel system for tracking pedestrians using multiple single-row laser-range scanners*. In IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, 35(2):283–291, 2005 (Zitiert auf den Seiten 48, 49 und 51)
- [Ziebart, 2009] B. Ziebart, N. Ratliff, G. Gallagher, C. Mertz, K. Peterson, J. Bagnell, M. Hebert, A. Dey, S. Srinivasa, *Planning-based prediction for pedestrians*. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 3931–3936, 2009 (Zitiert auf den Seiten 104 und 105)
- [Zouba, 2008] N. Zouba, B. Boulay, F. Bremond, M. Thonnat, *Monitoring Activities of Daily Living (ADLs) of Elderly Based on 3D Key Human Postures*. In B. Caputo, M. Vincze, editors, Proc. of Int. Cognitive Vision Workshop (ICVW), Lecture Notes in Computer Science, pp. 37–50, Springer Berlin Heidelberg, 2008 (Zitiert auf den Seiten 67, 68 und 88)