Technische Universität Ilmenau
Fakultät für Informatik und Automatisierung
Fachgebiet Neuroinformatik und Kognitive Robotik

Bestimmung von optimalen Positionen zur Beobachtung des Nutzers im häuslichen Umfeld

Masterarbeit zur Erlangung des akademischen Grades Master of Science

David Kudlek

Betreuer: Thanh Quang Trinh (M.Sc.)
Verantwortlicher Hochschullehrer:
Prof. Dr. H.-M. Groß, FG Neuroinformatik und Kognitive Robotik

Die Masterarbeit wurde am 18.01.2017 bei der Fakultät für Informatik und Automatisierung der Technischen Universität Ilmenau eingereicht.

Erklärung: „Hiermit versichere ich, dass ich diese Masterarbeit selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel verwendet habe. Alle von mir aus anderen Veröffentlichungen übernommenen Passagen sind als solche gekennzeichnet.“

Ilmenau, 18.01.2017 .....

David Kudlek

Inhaltsverzeichnis

1 Einleitung **1**

1.1 Motivation 1

1.2 Bedeutung für Projekte am Fachgebiet 2

1.3 Ziel 3

1.4 Kapitelübersicht 4

2 Grundlagen und -begriffe **7**

2.1 Eingesetzter Roboter 7

2.2 Umweltrepräsentation (Karten) 8

2.2.1 Gridkarte 9

2.2.2 Normal Distributions Transform (NDT) 14

2.3 Weitere Begrifflichkeiten 15

2.3.1 Sichtbarkeit 15

2.3.2 Soziale und sensortechnische Abstände 16

2.3.3 Partikelschwarmoptimierung (PSO) 16

3 Anforderungen **19**

3.1 Situationsbeschreibungen 19

3.1.1 Ruhende Person 19

3.1.2 Sich bewegende Person 20

3.1.3 Unauffindbare Person 20

3.1.4 Spielfreudige Person 21

3.1.5 Gestürzte Person 22

3.2 Aufgaben 22
3.2.1 Beobachte 23
3.2.2 Handle 25
3.2.3 Lerne 29

4 Stand der Technik 31
4.1 Übersicht 31
4.2 Verwandte Probleme 34
4.3 Positionssuche durch Optimierung 36
4.3.1 Autonomous Robot Cameraman 36
4.3.2 Finding a good Position to observe a Person 39
4.4 Positionssuche durch Klassifikation 41
4.5 Zusammenfassung und Ausblick 43

5 Optimierung der Beobachtungsposition 47
5.1 Aufbau und Ablauf 48
5.2 Kostenfunktionen 49
5.2.1 Erreichbarkeit 51
5.2.2 Sichtbarkeit 51
5.2.3 Kumulative Kartenkosten 55

6 Beobachtungsverhalten 59
6.1 Zustandsmaschine 61
6.2 Feedback 62

7 Experimentelle Untersuchung 63
7.1 Testumgebung 63
7.2 Laufzeit der Kostenfunktionen 64
7.3 Integrationstest 67
7.4 Akzeptanztests 70

8 Zusammenfassung und Ausblick 73
8.1 Zusammenfassung 73

8.2 Ausblick 74

A Weitere Abbildungen **77**

Literaturverzeichnis **87**

Kapitel 1

Einleitung

1.1 Motivation

Wir leben in einer Welt mitten im Umbruch der digitalen Revolution. Stück für Stück zieht die Technik in unseren Alltag ein: Von Stift und Papier zum Computer, vom Brief zur Kurznachricht und vom Auskunftsschalter zum Internet. Wir sind ständig verfügbar, vernetzt – wir sind ständig *online*. Neue Informationen erreichen uns im Sekundentakt. Angestoßen von der digitalen Revolution werden die Geräte immer intelligenter: Vom stationären Telefon zum mobilen Smartphone und von der Kutsche zum selbstfahrenden Auto. Immer mehr Assistenzsysteme finden ihren Weg in unsere Hand, um uns im Alltag zu unterstützen und um uns Arbeit abzunehmen.

Auch in den eigenen vier Wänden hält dieser Trend an. Die Hausautomation erfreut sich steigender Beliebtheit. Neben dem Komfort bei der Steuerung des Hauses und der Optimierung beim Energieverbrauch ergeben sich auch neue Möglichkeiten für Interaktionen mit dem Eigenheim. So wird man beim Verlassen des Hauses informiert, wenn noch Fenster geöffnet sind. Das vernetzte Zuhause eignet sich auch für die Einführung von sozialen Assistenzfunktionen. Da die System stationär an die Stromversorgung angebunden werden, sind sie ständig verfügbar und können rechenintensive Funktionen bieten.

Den Höhepunkt dieser Revolution nimmt die Entwicklung von Robotern ein. Sie arbeiten bereits Seite an Seite mit uns am Fließband oder saugen autonom den Staub

im Haus. Ein weiteres dieser Anwendungsfelder ist die Assistenzrobotik. Die Roboter zeichnen sich dabei durch ihre Mobilität und Interaktionsmöglichkeiten aus. Diese Möglichkeiten umfassen z. B. Tonausgaben, Bewegungen, Berührungen, Ausgaben über Bildschirme oder die Steuerung mit einer Fernbedienung.

Wir leben auch in einer Welt inmitten des demographischen Wandels. Durch diesen Wandel und die steigende Lebenserwartung wird es eine wachsende Anzahl an älteren Menschen geben. Gleichzeitig gibt es immer weniger Großfamilien, wodurch eine Betreuung und Pflege durch die eigenen Kinder nicht mehr gesichert ist. An dieser Stelle helfen Hausroboter oder Hausautomation den Älteren dabei, dass sie noch länger ein selbstständiges und selbstbestimmtes Leben führen können. Sie bieten z. B. Erinnerungsfunktionen für Termine, Geburtstage und Medikamente. Ergänzt wird das Spektrum durch Videotelefonie, um mit der Familie, den Behörden und den Ärzten in Kontakt zu bleiben. Ein Roboter kann durch seine Anwesenheit auch gegen ein Gefühl der Einsamkeit helfen.

Mit der Anzahl der Älteren steigt auch die der pflegebedürftigen Menschen, während das Pflegepersonal im Vergleich immer weniger wird. Mit dem Einsatz von sozialen Assistenzsystemen kann das Personal unterstützt und entlastet werden. Damit haben sie mehr Zeit um sich z. B. um Personen zu kümmern, die einen höheren Pflegebedarf benötigen.

1.2 Bedeutung für Projekte am Fachgebiet

Diese Arbeit gliedert sich in das aktuelle Projekt SYMPARTNER ([Gross et al.]) des Fachgebiets *Neuroinformatik und kognitive Robotik* (NIKR) ein. In diesem sollen nun ein System zur Hausautomation (*PAUL*) und ein Roboter unter dem Aspekt der sozialen Assistenz verschmolzen werden (Abb. 1.1). Daraus ergibt sich ein leistungsstarkes Duo. Die Hausautomation liefert dabei ein stationäres System, welches den Nutzer rund um die Uhr und überall im Haus betreuen kann. Zum Beispiel weiß das System durch Bewegungssensoren immer, wo sich der Nutzer aufhält. Damit bildet es das starke Rückgrat in dieser Kombination. Der Roboter fungiert als weiteres Gesicht, bzw. als *Avatar* des Systems. Durch seine Mobilität und Gestalt soll er vom Nutzer

Abbildung 1.1: SYMPARTNER Projekt
*Zu sehen ist ein Mitarbeiter des Fachgebiets NIKR mit einem Roboter und mit einem Tablet PC zur Bedienung des Systems PAUL. Quelle: [*Gross *et al.]*

als Lebewesen wahrgenommen werden. Damit kann er Gesellschaft leisten und Berührungsängste gegenüber der Technik abbauen. Weiterhin ermöglicht dies auch eine einfache Bedienung mittels natürlicher Sprache oder Gestik. Er ist das Gesicht und die Brücke zur Technik.

1.3 Ziel

Der Nutzer wird nicht ständig mit dem Roboter arbeiten, so dass dieser in einer geeigneten Position auf weitere Interaktionen warten kann. Dabei soll er den Nutzer möglichst dauerhaft oder zumindest periodisch mittels Kamera oder Sensoren sehen. Sollte der Nutzer für längere Zeit nicht gesehen worden sein, so muss der Roboter nach ihm suchen, um auf einen Notfall reagieren zu können. Dieses Beobachtungsverhalten soll den Nutzer jedoch nicht stören, z. B. indem der Roboter in sehr kurzen Zeitabständen nach dem Rechten sieht. Für eine gute Warteposition spielen viele Kriterien eine Rolle. Im Laufe dieser Arbeit werden einige Ansätze aus früheren Arbeiten des Fachgebiets wieder aufgegriffen. Es wird eine Fähigkeit zur Optimierung der Beobachtungsposition (*OptimalPoseFinder*) (Kap. 5), sowie ein Beobachtungsverhalten (*AttendanceBehavior*) (Kap. 6) entwickelt.

1.4 Kapitelübersicht

Kapitel 2: In diesem Kapitel werden die Grundlagen gelegt. Es wird auf verschiedene Datentypen und Modelle eingegangen, die in dieser Arbeit verwendet werden. Dadurch soll dem Leser ein Verständnis für die Terminologie näher gebracht werden.

Kapitel 3: Dieses Kapitel behandelt die allgemeinen Anforderungen an ein Beobachtungsverhalten. Dem Leser soll damit ein Überblick über die möglichen Aspekte gegeben werden. Eine Teilmenge dieser wird in der Arbeit realisiert. Sie bietet gleichzeitig einen Ausblick auf zukünftige Entwicklungen.

Kapitel 4: Hier werden nun die Ergebnisse der Literaturrecherche präsentiert. Die Arbeiten mit einem besonderen Bezug zum Thema werden hervorgehoben. Darunter werden auch einige Projekte des Fachgebiets NIKR aufgearbeitet, auf die später aufgebaut wird.

Kapitel 5: In diesem Kapitel wird die optimale Positionssuche dargestellt. Dabei verzichtet es auf die Nutzung des aktuellen Aufenthaltsorts der Person. Stattdessen werden neue Kostenfunktionen entwickelt, die auf einer Aufenthaltswahrscheinlichkeit des zu beobachtenden Nutzers beruhen. Damit ist die Positionssuche unabhängig von der Person und betrachtet Positionen, bei denen der Nutzer erst in naher Zukunft sichtbar sein wird. Erstmalig wird auch die Nutzung von Feedback berücksichtigt, die eine bessere Anpassung gewährleisten soll.

Kapitel 6: Um die Positionssuche zu nutzen wird ein entsprechendes Verhalten auf dem Roboter gebraucht. Mit genau diesem beschäftigt sich dieses Kapitel. Dem Leser wird ein neues Beobachtungsverhalten präsentiert, bei dem sich der Roboter nur selten neu positionieren muss. Dabei werden die Zusammenhänge zu anderen Funktionen und Verhalten auf dem Roboter verdeutlicht. Es wird auch der besondere Einsatz von Feedback hervorgehoben.

Kapitel 7: In diesem Kapitel werden die verschiedenen Tests dargestellt. Dabei wurden Funktions- und Integrationstest durchgeführt. Ein kurzer Akzeptanztest gibt Aufschluss über ein weiteres Vorgehen. Aufgrund der geringen Teilnehmeranzahl ist dieser jedoch nicht repräsentativ.

Kapitel 8: Abschließend werden alle Arbeiten und Ergebnisse zusammengefasst. Beruhend darauf wird ein Ausblick auf zukünftige Entwicklungen gegeben. Dabei wird besonderer Wert auf weitere Tests gelegt, die die Akzeptanz des neuen Verhaltens prüfen.

Kapitel 2

Grundlagen und -begriffe

In diesem Kapitel werden die Grundlagen geschaffen. Dabei verdeutlicht es die Begriffe und die Gegebenheiten, die für ein Verständnis notwendig sind. Es wird mit dem verwendeten Roboter angefangen. Danach folgen verschiedene Kartenrepräsentationen, sowie weitere Begrifflichkeiten.

2.1 Eingesetzter Roboter

Im laufe dieser Arbeit wurde ein Roboter SCITOS G3 (Abb. 2.1) der Firma MetraLabs eingesetzt. Dabei verfügt er über verschiedene Sensoren. Für die Hinderniserkennung nutzt er einen Laserscanner und einen Ring aus Sonarsensoren. Für die Kartierungen kommt neben dem Laserscanner eine Tiefenkamera (RGBD) des Typs Microsoft Kinect zum Einsatz. Darüber hinaus besitzt er eine Kamera mit Fischaugen-Objektiv zu Erkennung von Personen, sowie zur Videotelefonie. Sie dient optisch auch als Nase des Roboters. Aus Sicherheitsgründen verfügt er über einen Berührungssensor (*Bumper*), um Kollisionen zu registrieren und den Roboter zu stoppen. Weiterhin verfügt er über verschiedene Eingabemöglichkeiten. Serienmäßig handelt es sich dabei um ein Touch-Display. Dieses wird ergänzt durch eine Fernbedienung und einen textilen Berührungssensor. Auf dem Roboter stehen bereits einige Dienste zur Verfügung. Ein *PersonTracker* sorgt für eine aktuelle *Posenhypothese*. Dabei handelt es sich um eine dreidimensionale Normalverteilung, die die aktuelle Position einer Person beschreibt.

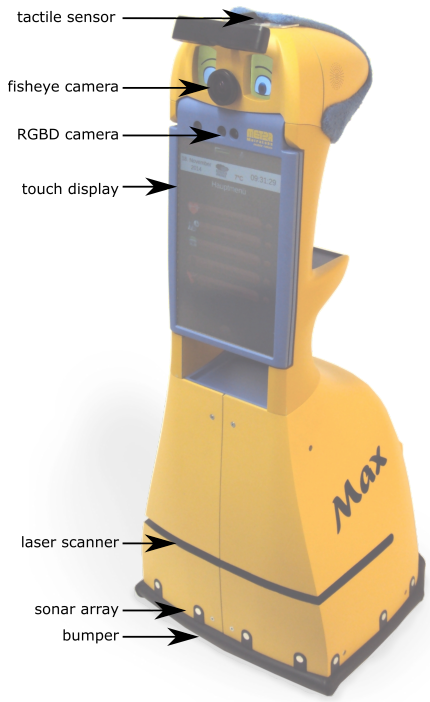


Abbildung 2.1: Eingesetzter Roboter
Im Bild ist ein Roboter der Firma MetraLabs dargestellt. Es werden dabei die vorhandenen Sensoren und Bedienmöglichkeiten gezeigt. Quelle: [Max]

Die Mittelwerte stehen für die Position und die Varianz für die Unsicherheit dieser Hypothese. Da der Tracker mehrere Personen erkennen kann, verfügt der Roboter über eine Wiedererkennung, die diese eindeutig unterscheidet. Weiterhin steht bereits ein *Pilot* für die Navigation zur Verfügung. Dieser realisiert die Fahrt zu einer Position, mit der er beauftragt wird. Dabei produziert er eine Kostenkarte (Kap. 2.2.1), die zur Fahrt unerlässlich ist. Zuletzt steht auch ein Verhalten zur Personensuche (*SearchBehavior*) zur Verfügung. Wird dies aktiviert, so sucht es zuerst Positionen mit hoher Aufenthaltswahrscheinlichkeit (Kap. 2.2.1) ab. Trifft der Roboter auf eine erkannte Person, so erwartet er eine Rückmeldung in Form einer Eingabe über die Fernbedienung.

2.2 Umweltrepräsentation (Karten)

In diesem Kapitel stehen nun Umweltrepräsentationen im Mittelpunkt. Im späteren Verlauf werden sie die Hauptrolle bei der Realisierung einer optimalen Positionssuche spielen.

2.2.1 Gridkarte

Die Gridkarte stellt ein Basismodell für die Abbildung der Umwelt dar. Dabei wird die Welt in eine zwei- oder dreidimensionale Matrix unterteilt. Jede Zelle entspricht dabei einem diskreten Stück der Umwelt. Hat eine Zelle z. B. eine Auflösung on $0,01m$ so bedeutet das, dass eine zweidimensionale Fläche von $10cm \cdot 10cm$ auf eine Zelle abgebildet wird. Zusätzlich zu dieser Unterteilung haben die Zellen einen Wert. Bei einer Hinderniskarte (*Occupancy Grid Map*) beschreibt dieser Wert, mit welcher Wahrscheinlichkeit in einer Zelle ein Hindernis angetroffen wird. Auf Basis dieses Modells lassen sich weitere Karten ableiten. Eine Kostenkarte beschreibt, mit welcher Wahrscheinlichkeit eine Zelle in der Karte mit dem Roboter angefahren werden kann. Die Aufenthaltswahrscheinlichkeit zeigt die Position, an der sich eine Person aufhalten wird. Eine Feedbackkarte beschreibt, wie gut eine bestimmte Position ist.

Kostenkarte (Erreichbarkeit)

Die Erreichbarkeit stellt das wichtigste Kriterium für eine gute Positionswahl dar. Wird sie nicht erfüllt, so ist die gewählte Position nutzlos. Sie definiert sich dadurch, dass es einen Pfad von einem Startpunkt (dem Roboter) zu einem gewählten Zielpunkt gibt. Dies lässt sich gut mit einer zweidimensionalen Karte darstellen. Dabei wird diese in erreichbare und nicht erreichbare Zellen unterteilt. Eine Karte der Erreichbarkeit ist jedoch immer nur eine Momentaufnahme, da sie in dieser Form keine dynamischen Hindernisse bedenkt. Sie muss demnach kontinuierlich aktualisiert werden, um von Nutzen zu sein. So kann der Weg zu einer guten Position versperrt werden oder eine bessere Position wird plötzlich erreichbar. Die Kostenkarte setzt sich dabei aus verschiedenen anderen Karten zusammen (Abb. 2.2).

Die Basis bildet eine globale Hinderniskarte. Sie enthält alle Hindernisse, die von den Sensoren erfasst wurden. Dabei wird sie vor Beginn der Nutzung in der Einsatzumgebung erstellt. Es handelt sich dabei um eine statische Karte, da sie nicht mehr verändert wird. In diesem Zusammenhang gibt es auch eine lokale Hinderniskarte. In ihr werden aktuelle Änderungen gespeichert wie z. B. ein geschlossene Tür oder eine Person. Sie ergänzt die Karte um dynamische Hindernisse. Darauf folgt eine manuell

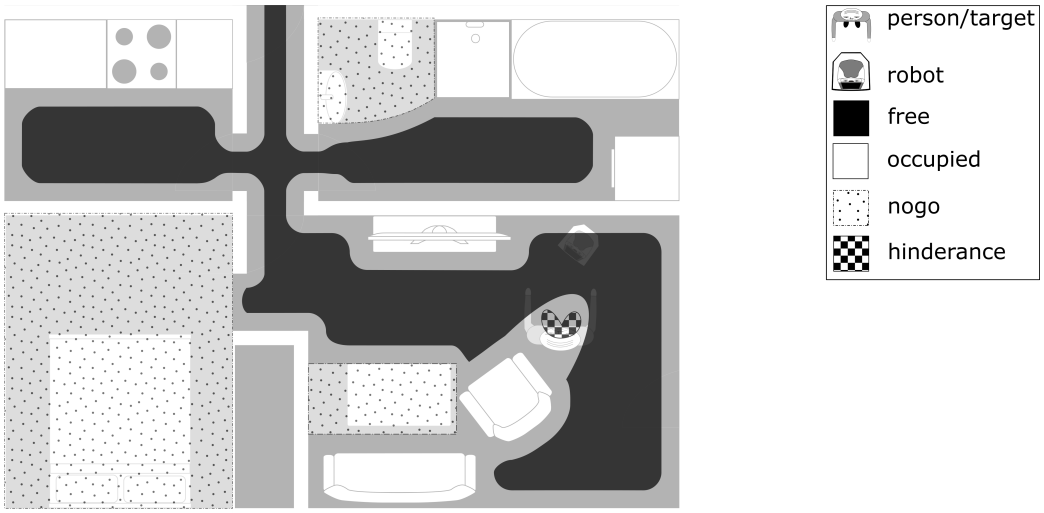


Abbildung 2.2: Erreichbarkeit

In diesem Bild wird eine berechnete Erreichbarkeit für alle Zellen der Karte gezeigt. Die Nogo-Karte verhindert hier z. B. das Befahren des Schlafzimmers und des Waschbeckenbereichs im Badezimmer. Die aktuelle Position der Person wird in einer lokalen Hinderniskarte berücksichtigt. Als unterste Schicht wird die globale Hinderniskarte verwendet, die Möbel und Wände kartiert hat. Auf dieser kombinierten Karte wird nun eine Dilatation eingesetzt, die den Umfang des Roboters berücksichtigt. In der finalen Karte sind alle schwarzen Zellen erreichbar.

erstellte *No-Go-Karte*. Sie erfüllt zwei Aufgaben. Die erste Aufgabe ist die Einschränkung der Karte an Stellen, die von den Sensoren nicht wahrgenommen werden. Darunter fallen bodentiefe Glasflächen oder Treppenaufgänge. In ihrer zweiten Aufgabe erfüllt sie nutzerspezifische Restriktionen. Dieser möchte z. B. nicht, dass der Roboter das Schlafzimmer betritt, während er dort schläft.

Im Moment entsprechen die freien Zellen in der Karte noch allen sichtbaren Zellen. Dabei wird jedoch nicht der Umfang des Roboters berücksichtigt, so dass er bei einer Fahrt mit einem Hindernis kollidieren könnte. Deshalb wird nun eine Dilatation auf die kombinierten Karten angewandt. Dabei wird um jede belegte Zelle in der Originalkarte ein Kreis mit dem Umfang des Roboters als belegte Zellen hinzugefügt. Bildlich gesprochen wird der Roboter als Punkt in der neuen Karte dargestellt. Alle freien Zellen in der neuen Karte sind nun erreichbare Positionen. Damit steht nun die Kostenkarte (*cost map*) zur Verfügung. Es wird dadurch sichergestellt, dass der Roboter rangieren und navigieren kann. Die fertige Karte hat jedoch noch den Mangel, da es einige eingeschlossene Bereiche gibt, die fälschlicherweise als erreichbar erkannt werden. Diese müssen für eine korrekte Ausführung der Positionssuche noch beseitigt werden.

Aufenthaltswahrscheinlichkeit

Mit dieser Fähigkeit wird die Wahrscheinlichkeit beschrieben, mit der sich die Zielperson an einer bestimmten Position aufhält. Dabei sind die Begriffe *person occurrence map* ([Volkhardt und Gross, 2013]) und *person occupany distribution* (POD) ([Kessler et al., 2012]) geläufig. Sie lässt sich in zwei Kategorien aufteilen: Die Schätzung von Aufenthaltsorten (Abb. 2.3 links) und Laufwegen (Abb. 2.3 rechts). Die Aufenthaltsorte können bei der Suche nach einer geeigneten Position als Ersatz für eine Posenhypothese genutzt werden. Insbesondere, wenn es eine zeitliche Abhängigkeit gibt wie z. B. dass sich die Person zwischen 19 und 20 Uhr stets im Wohnzimmer aufhält. Damit berücksichtigt man den zukünftigen Aufenthaltsort. Auch bei der Personensuche kann diese Karte eingesetzt werden. In diesem Fall fährt der Roboter alle Orte nacheinander ab, bis er die Person findet. Laufwege lassen sich ebenfalls bei der

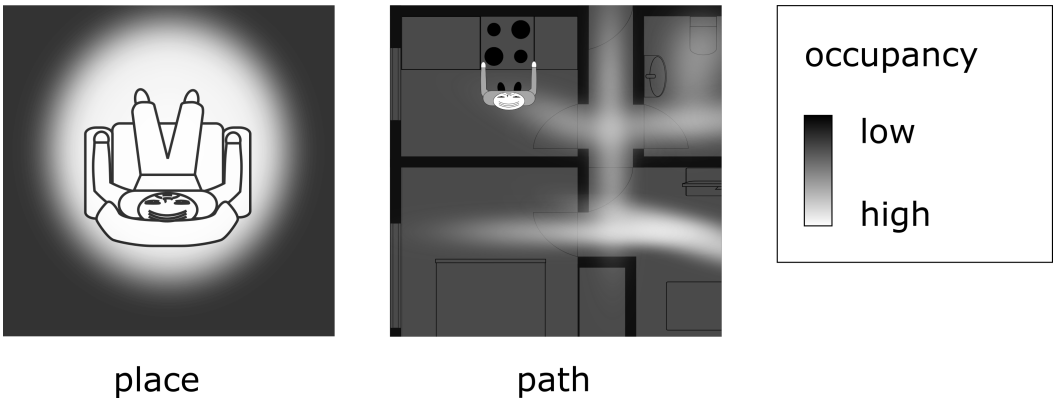


Abbildung 2.3: Aufenthaltswahrscheinlichkeit
Im Bild zu sehen sind die zwei Ziele einer Aufenthaltswahrscheinlichkeit. Einerseits lassen sich aus ihr mögliche Aufenthaltsorte ermitteln (links) und anderseits häufig genutzte Wege (rechts).

Positionssuche berücksichtigen, denn Türdurchgänge und Flure haben dort zum Beispiel eine höhere Aufenthaltswahrscheinlichkeit. Er sollte also dort nach Möglichkeit nicht stehen. Dies ist besonders im Zusammenhang mit einer langfristigen Beobachtung relevant. Angenommen der Roboter steht im einzigen Durchgang, der aus dem Wohnzimmer herausführt. Der Nutzer wird dann irgendwann an ihm vorbei müssen, um z. B. auf Toilette zu gehen.

Die Aufenthaltswahrscheinlichkeit lässt sich nur durch lange Beobachtungen der Posenhypothese schätzen, wobei typische Bewegungs- und Gewohnheitsmuster dies unterstützen. Aufenthaltsorte lassen sich wiederum einfacher bestimmen als Laufwege. Das begründet sich darin, dass die Person sich lange an einem Ort aufhält und es dementsprechend viele Beobachtungen gibt. Laufwege hingegen werden deutlich seltener komplett beobachtet, da sich die Person auch außerhalb der Sicht des Roboters durch die Wohnung bewegt.

Ergänzend zur Aufenthaltswahrscheinlichkeit existiert auch die Raumvorhersage. Sie versucht ebenfalls Muster in den Bewegungen zu erkennen. Aber statt dies nur mit der Zeit zu verknüpfen, spielen hier zuvor besuchte Räume eine Rolle ([Gellert und Vintan, 2006]). Geht eine Person frühs vom Schlafzimmer zum Bad, wird sie danach

z. B. in die Küche gehen und dort frühstücken. Durch solch eine Verkettungen lässt sich vorhersagen, welchen Raum die Person als nächstes betreten wird.

Feedback

Die Feedback-Karte bietet dem Nutzer die Möglichkeit, das Verhalten des Roboters gemäß seiner Vorlieben anzupassen. Der Wert einer Gridzelle ist umso niedriger, je mehr positives Feedback sie bekommt. Während es sich bei der No-Go-Karte um das strikte Erlauben und Verbieten handelt, wird hier ein fließender Übergang genutzt. Bei der No-Go-Karte dürfen verbotene Bereiche nicht durchfahren werden, damit der Roboter dort z. B. nicht mit einem Hindernis kollidiert. Bei der Feedback-Karte ist das jedoch unbedenklich. Hierbei muss der Roboter über eine entsprechende Funktionalität zur Feedback-Gewinnung verfügen, z. B. durch einen berührungsempfindlichen Sensor (Streichelfell) oder durch eine Fernbedienung. Dabei lassen sich drei Arten der Nutzung unterscheiden: Statisch, manuell und automatisch (Abb. 2.4).

Beim statischen Feedback kann der Nutzer bei der Erstinbetriebnahme des Roboters seine Wünsche äußern, die dann während der Laufzeit berücksichtigt werden. Sollte ihm die Position des Roboters nicht gefallen, so hat er zu diesem Zeitpunkt keine Möglichkeit mehr, das zu ändern. Beim manuellen Feedback greift der Nutzer ständig ein, um sowohl positive wie auch negative Ereignisse zu bewerten. Bei negativer Rückmeldung kann der Roboter eine neue Positionssuche starten und dabei seine neuen Erkenntnisse mit einfließen lassen. Auch wenn die Nutzung einer Fernbedienung ein Komfortelement darstellt, muss der Nutzer ständig weiter bewerten. Deshalb gibt es eine letzte Variante, mit der sich auch automatisch Feedback generieren lässt. Fährt der Roboter eine Position an und erhält dort keine negative Rückmeldung, so kann er sich selbst ein geringes positives Feedback vermerken. Dadurch erhöht sich der Komfort, da der Nutzer nur noch negative Ereignisse bewertet, die ihn wirklich stören. Auch eine automatisierte, negative Rückmeldung ist denkbar. Erreicht der Roboter eine Beobachtungsposition und kann die Person dort bis zu einem Timeout nicht sehen, so gibt er sich selbst ein geringes negatives Feedback.

An dieser Stelle ist es auch eine Überlegung wert, ob es einen zeitlichen Zusammen-

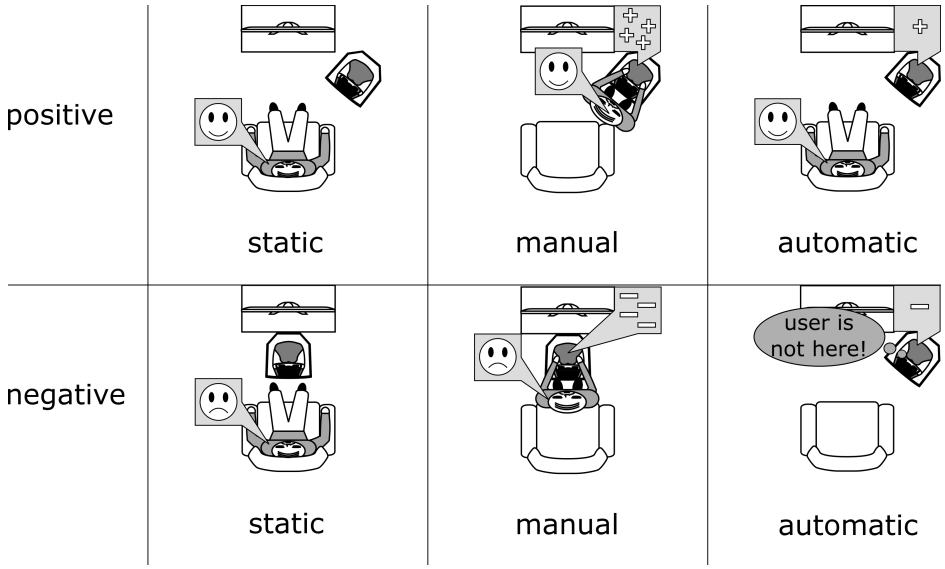


Abbildung 2.4: Varianten des Feedback

Der Einsatz von Feedback kann auf drei verschiedene Arten geschehenn. Auf der linken Seite ist eine statische Karte gegeben, so dass der Nutzer das Ergebnis nur akzeptieren kann. In der Mitte hat der Nutzer die Möglichkeit, durch Interaktion positive und negative Impulse zu setzen. Auf der rechten Seite ist die automatische Feedbackgenerierung zu sehen. Wird ein Roboter an einer Position nicht bestraft, so bekommt er positives Feedback. Beobachtet der Roboter von einer Position aus und hat die Person nie gesehen, so bekommt er negatives Feedback.

hang zwischen Feedback und Position gibt. Ähnlich wieder der Aufenthaltswahrscheinlichkeit kann auch diese von bestimmten Verhaltensmustern abhängen. So kann eine Position am Morgen noch negativ, aber mittags positiv bewertet werden.

2.2.2 Normal Distributions Transform (NDT)

Hierbei handelt es sich um ein effizientes Umweltmodell insbesondere für den dreidimensionalen Raum. Durch Sensoren wie Tiefenkameras werden bei der Kartierung Punkte erzeugt. Diese einzeln zu speichern ist jedoch zu aufwendig, weshalb eine Abstraktion stattfindet. Doch zuerst wird die Umwelt in dreidimensionale Zellen (*Voxel*) gleicher Größe unterteilt. Dadurch entsteht eine Ähnlichkeit zu den Gridkarten. Doch

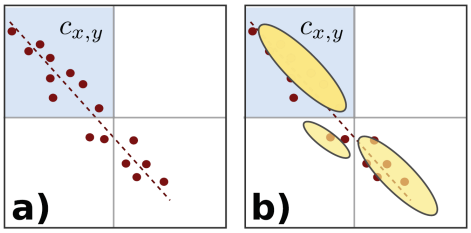


Abbildung 2.5: Entstehen einer NDT-Zelle
*In den beiden Ausschnitten ist je ein zweidimensionales Grid aus zwei mal zwei Zellen abgebildet. Links werden die Entfernungswerte (z. B. durch einen Laserscan) in das Grid eingetragen. Mit Hilfe dieser Punkte wird im rechten Bild die Normalverteilung bestimmt. Quelle: Ausschnitt aus [*Saarinen *et al., 2013].*

im Unterschied zu ebendiesen wird in der Zelle nicht ein Wert, sondern eine mehrdimensionale Normalverteilung gespeichert. Diese Verteilung ist unter dem Begriff *Normal Distributed Transform* (NDT) bekannt ([Biber und Strasser, 2003]). Die genaue Position innerhalb der Zelle wird durch die Mittelwerte und die entsprechende Kovarianzmatrix abstrahiert. In Abbildung 2.5 wird dieses Vorgehen im zweidimensionalen Raum dargestellt.

2.3 Weitere Begrifflichkeiten

Im folgenden Abschnitt werden weitere Begriffe erklärt, wobei die ersten zwei Unterkapitel mit Kostenfunktionen zusammenhängen und zuletzt ein besonderer Optimierer vorgestellt wird.

2.3.1 Sichtbarkeit

Bei der Suche nach einer Beobachtungsposition beschreibt man damit in erster Linie die *Sichtbarkeit der Person*, also dessen Oberkörperausrichtung, Gesicht, Mimik und Gestik. Unter anderem arbeitet man hier mit der Posenhypothese und der Ausrichtung des Roboters zu ebendieser. Dadurch wird sichergestellt, dass die Kamera in die richtige Richtung zeigt. Des Weiteren nutzt man das Kamerabild um einerseits festzustellen,

ob ein Gesicht erkannt wird und anderseits wie groß es im Verhältnis zur Bildgröße ist. Das spielt z. B. eine Rolle bei der Videotelefonie. Auch für die Erkennung von Mimik wird eine bestimme Auflösung des Gesichtes benötigt.

2.3.2 Soziale und sensortechnische Abstände

Technischen Abstände sind ein Kriterium der praktischen Natur. Sie beschreiben die Limits für die verwendeten Sensoren. Das umfasst zum Beispiel die Entfernung, in der das Gesicht oder die Posenhypothese noch sicher erkannt wird. Sie stellen den Roboter in den Vordergrund (Abb. 2.6).

Beim *sozialen Abstand* handelt es sich um ein Kriterium der menschlichen bzw. tierischen Natur. Welchen Einfluss bestimmte Entfernung zwischen Lebewesen haben, wurde schon in [Hall, 1966] ergründet. Bezogen auf die Menschen kann man sagen, das die Distanz um so geringer ist, je besser sich zwei Personen kennen. Auf Grundlage dessen bilden sich in der Robotik drei Bereiche für die Interaktion mit Menschen heraus (Vgl. Abb. 2.7). Dabei ist der Übergang zwischen den einzelnen Bereichen fließend und kann sich auch von Mensch zu Mensch unterscheiden. Im ersten Bereich ist der Roboter aus Sicht des Menschen zu nah und löst Unbehagen aus, insbesondere wenn diese nicht an den Umgang mit mobilen Assistenzsystemen gewöhnt sind. Dies wurden auch in Studien von [Walters et al., 2007] und [Koay et al., 2007] gezeigt. Im letzten Bereich ist der Roboter zu weit von der Person entfernt, so dass diese nicht mehr zur Interaktion mit dem System angeregt wird. In der Mitte wird die Balance zwischen beiden Bereichen geschafft.

2.3.3 Partikelschwarmoptimierung (PSO)

Bei der PSO handelt es sich um ein Optimierungsverfahren für dynamische, mehrdimensionale Probleme. Ursprünglich entwickelt, um die Bewegung von Vogelschwärmen zu simulieren, leistet sie heute einen wichtigen Beitrag zur Optimierung ([Kennedy und Eberhart, 1995]). Sie arbeitet dabei mit einer Menge an Partikeln, ähnlich einem Partikelfilter. Diese Partikel werden zufällig in einem Umkreis um eine Startposition initialisiert, diese könnte z. B. die Roboterposition sein. Im Gegensatz zum Partikel-

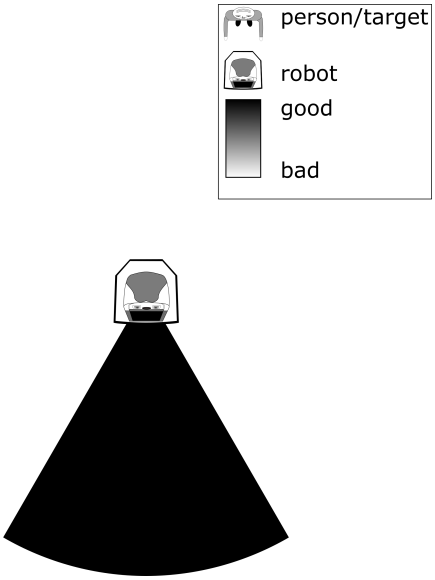


Abbildung 2.6: Technische Distanz

Bei dieser Kostenfunktion steht der Roboter im Mittelpunkt. Die Charakteristiken seiner Sensoren sind von Bedeutung. Hier z. B. die Reichweite und der Öffnungswinkel einer Kamera.

Abbildung 2.7: Soziale Distanz

Zwischen den Menschen gibt es eine soziale Distanz. Zu sehen ist die fließende Unterteilung im Kreis um die Person. Für eine Suche sind Positionen, die zu nah oder zu entfernt sind, ungünstig.

filter wird hier jedoch eine künstliche Bewegung der Partikel eingeführt, sogenannte *Geschwindigkeiten*. Das Update der Geschwindigkeiten ist dabei in Gleichung (2.1) für eine Dimension zu sehen ([Kessler et al., 2012]).

$$v_t^{(i)} = K\Big[v_{t-1}^{(i)} + c_1 r_1 (p_t^{(i)} - x_t^{(i)}) + c_2 r_2 (g_t - x_t^{(i)})\Big] \tag{2.1}$$

Die innere Summe besteht dabei u. a. aus der Geschwindigkeit des vorherigen Zeitschrittes. Danach kommt der Abstand zum aktuellen lokalen Optimum des Partikels hinzu. Dieser wird mit einem Zufallsfaktor $r_1 \in [0, 1]$ und einer Konstanten c_1 multipliziert. Gleiches trifft in Bezug auf den Abstand zum globalen Optimum zu. Die Wahl von c_1 und c_2 bestimmt, ob das lokale oder das globale Optimum bevorzugt werden sollte.

$$K = \frac{2}{|2 - \theta - \sqrt{\theta^2 - 4\theta}|}, \text{ mit } \theta = c_1 + c_2, \quad \theta > 4 \tag{2.2}$$

Um die Konvergenz zu sichern wird die Summe mit einer Konstanten $K < 1$ multipliziert (Gl. (2.2)).In jedem Schritt bewegt sich das Partikel also entweder in die Richtung seines lokalen oder des globalen Optimums. Dabei kann es sich auch in der Richtung über das Optimum hinaus bewegen. Nachdem sich alle Partikel bewegten, haben sie eine neue, möglicherweise optimale, Position erreicht. Es werden nun die lokalen und das globale Optimum neu bestimmt. Die zuvor genannte Konstante K sorgt dafür, dass die Geschwindigkeit immer weiter reduziert wird, bis sich die Partikel nicht mehr bewegen. Gleichzeitig bewirkt es, dass sich die Partikel in der näheren Umgebung optimieren und sich so auf das Optimum einpendeln können.

Kapitel 3

Anforderungen

Dieses Kapitel befasst sich mit den Anforderungen an eine optimale Beobachtungsposition und dem Nutzen. Zuerst werden verschiedene Situationen mit einer Person präsentiert und wie das Verfahren damit umgehen wird. Darauf folgen die Teilaufgaben, die dabei zu bewältigen sind. Dabei werden auch Aufgaben genannt, die den Rahmen dieser Arbeit übersteigen. Damit wird ein Spielraum für zukünftige Arbeiten gegeben. Es wird Überschneidungen mit bereits bestehenden oder sich in Arbeit befindlichen Projekten geben. Es wird Bezug auf die Möglichkeiten in Kombination mit Hausautomation und Fitness-Trackern genommen, in der Arbeit selbst wird sich auf die Nutzung des Roboters und dessen Sensoren beschränkt. Im Kapitel 4.5 werden die Einschränkungen und Gegebenheiten konkret gefasst.

3.1 Situationsbeschreibungen

In diesem Abschnitt werden nun verschiedene Situationen im Umgang mit einer Person präsentiert. Dabei wird auch das erwünschte Verhalten des Roboters beschrieben.

3.1.1 Ruhende Person

Der Benutzer führt eine Tätigkeit aus, bei der er über lange Zeit an einem Ort bleibt, z. B. beim Lesen oder Fernsehen. Der Roboter soll dann eine geeignete Position finden

und von dort aus beobachten. Bis sich die Person dem Roboter wieder zuwendet, wartet er.

3.1.2 Sich bewegende Person

Bei einer Person, die sich im Raum oder zwischen verschiedenen Räumen bewegt ist es sinnvoll diese entsprechend ihre Abwesenheitsdauer zu unterscheiden.

Kurze Abwesenheit

Die Bezugsperson ist für einen kurzweiligen Moment nicht im Sichtbereich, z. B. beim Benutzen von Sanitäranlagen, beim Holen eines Erfrischungsgetränks oder beim Herrichten einer kleinen Mahlzeit. Sie erwartet, dass der Roboter nicht plötzlich in der Tür steht. Er soll an seiner letzten Position auf die Rückkehr des Nutzers warten. Dieser Fall beschreibt den Aspekt, dass der Roboter im Moment nicht gebraucht wird und nicht stören soll.

Lange Abwesenheit

Die Bezugsperson ist für einen längeren Zeitabschnitt nicht im Sichtbereich, z. B. beim Zubereiten einer ausgiebigeren Mahlzeit oder beim Baden. Nach einer gewissen Zeit muss der Roboter aktiv werden und nach der Person suchen. Dies geschieht im Einklang mit der Aufgabe, dass er den Nutzer trotz Warteposition weiter beobachten soll. Das gleiche Verhalten trifft zu, wenn sich die Person innerhalb des Raums oft bewegt und nicht mehr sichtbar ist.

3.1.3 Unauffindbare Person

Die Bezugsperson ist für einen längeren Zeitabschnitt nicht im Sichtbereich. Das kann passieren, wenn die Person ein Bad nimmt. Jedoch kann sie auch das Haus verlassen haben, obwohl sie dafür eine Begleitung bräuchte. In diesen Fällen wird Roboter eine Suche starten und erfolglos beenden. Daraus ergeben sich zwei mögliche Handlungen.

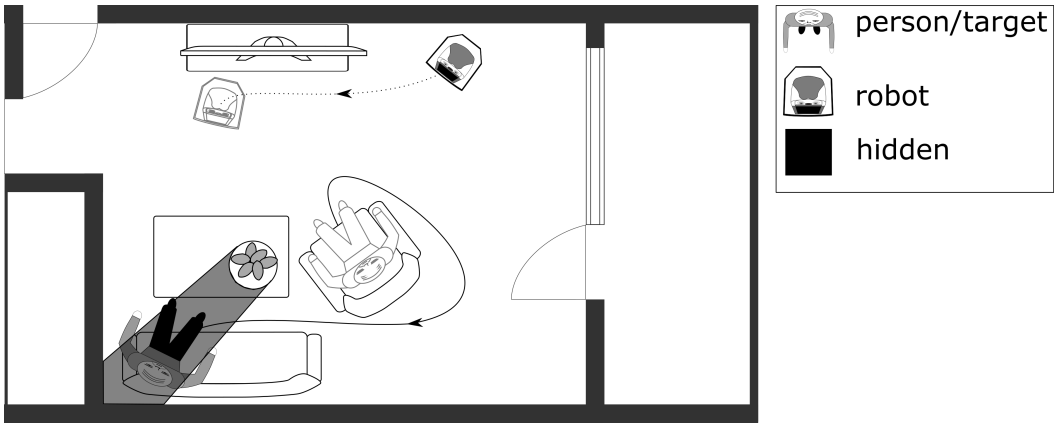


Abbildung 3.1: Spielfreudige Person

Die Person versteckt sich bewusst außerhalb des Sichtbereiches des Roboters. Sie hofft, dass der Roboter schnell eine neue Position einnimmt und sie sich wieder verstecken kann.

Der Roboter soll in eine Ruhemodus wechseln, da die Person nicht beobachtbar ist oder er muss einen Alarm auslösen, da sich die Person in einer Notlage befinden kann.

3.1.4 Spielfreudige Person

Der Nutzer versteckt sich, um den Roboter gezielt zu einer Reaktion zu bewegen. Zum Beispiel könnte sich die Person auf einem Sofa immer ein Stück aus dem Sichtbereich bewegen (Abb. 3.1). Die Erwartung wäre, dass der Roboter ebenfalls etappenweise eine neue Position sucht, von der er die Person wieder sehen kann, so dass dieses Spiel öfter wiederholt werden kann. In diesem Fall wird der Roboter nicht den Erwartungen an ein Spielverhalten gerecht. Das Verhalten ist unter Berücksichtigung der anderen Situationen zu träge, um einen Spielfluss zu ermöglichen. Des Weiteren versucht der Roboter eine Position zu finden, bei der er den Nutzer klar und möglichst unbeeinträchtigt im Bild hat. Ein weiterer Punkt, der gegen ein Spielverhalten spricht, ist das Suchverhalten, welches bei Verlust der Person aktiviert wird. Mit den hier entwickelten Positionssuche ist ein Spielverhalten in der oben beschrieben Weise nicht realisierbar. Ein solches Verhalten muss demnach separat erarbeitet werden.

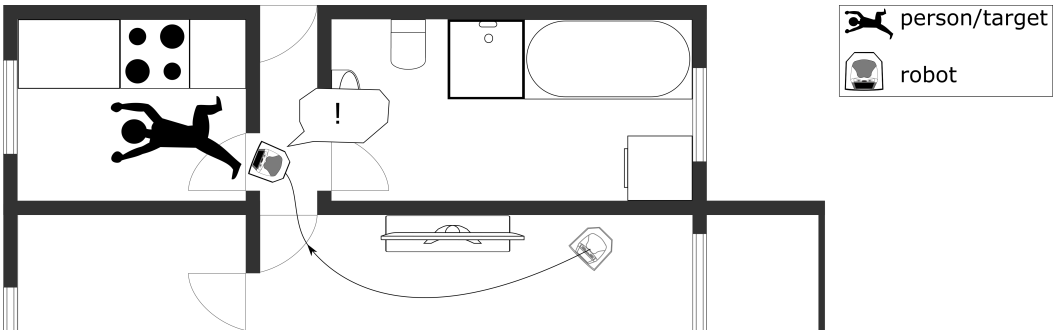


Abbildung 3.2: Gestürzte Person
Die Person ist außerhalb des Sichtbereiches gestürzt und benötigt Hilfe. Der Roboter sucht zeitnah nach der Person.

3.1.5 Gestürzte Person

Eine Person stürzt außerhalb des Sichtbereiches des Roboters und benötigt medizinische Hilfe (Abb. 3.2). Der Roboter muss ihr zeitnah zu Hilfe kommen. Bei der Suche nach der Person muss auch eine Notsituation erkannt werden können. Alternativ muss der Roboter im Beobachtungsverhalten auf Hilferufe oder ähnliches reagieren können.

3.2 Aufgaben

Das Thema umfasst weit mehr als nur das Finden und Navigieren zu einer geeigneten Position. Das Beobachten des Nutzers selbst stellt ebenfalls ein facettenreiches Verhalten dar. Denn im Gegensatz zum Standby muss der Roboter hier aktiv bleiben und auf Veränderungen reagieren. Die Person kann sich z. B. nicht im gewünschten Zielbereich aufhalten, wodurch die Beobachtung nicht möglich ist. Wir unterscheiden an dieser Stelle drei Aspekte: Beobachte (observe), handle (act) und lerne (adapt) (Abb. 3.3). Das Beobachten umfasst alle Bereiche, die der Roboter durch Sensoren wahrnehmen und bewerten können. Es umfasst hier auch die Suche nach und Anfahrt zur optimalen Beobachtungsposition. Das beschränkt sich nicht auf die eingebauten Sensoren und kann durch Hausautomation oder intelligente Geräte erweitert werden. Das Handeln beschreibt Fähigkeiten und Aktionen, mit denen wir auf Veränderun-

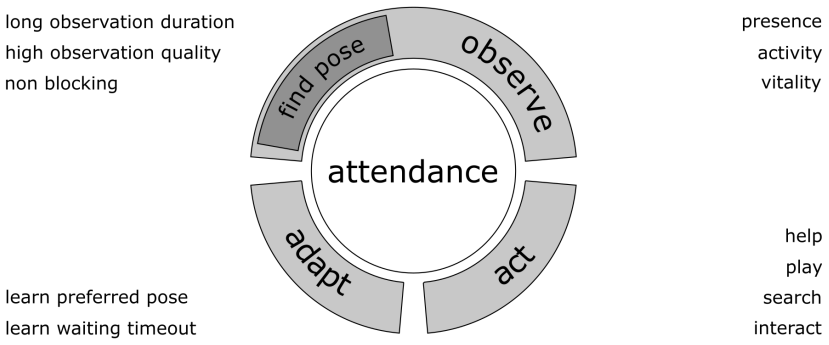


Abbildung 3.3: Funktionale Aspekte eines Beobachtungsverhaltens
Im Bild zu sehen ist das Beobachtungsverhalten (attendance) mit seinen grundlegenden Komponenten und Fähigkeiten. Das Finden einer optimalen Position wird dabei dem Beobachte Aspekt zugeordnet.

gen reagieren müssen (z. B. Person vermisst/verloren). Das letztere Konzept umfasst die Fähigkeit, sich besser an den Nutzer anzupassen, dessen typisches Verhalten zu erlernen und damit seine Akzeptanz gegenüber dem Roboter zu erhöhen. In den entsprechenden Abschnitten wird auch auf das besondere Potential eingegangen, was eine plattformübergreifende Kommunikation zwischen Roboter, Hausautomation und anderen intelligenten Geräten mit sich bringt.

3.2.1 Beobachte (observe)

Das Beobachten beschreibt den Vorgang der Aktivierung des Verhaltens bis zum eigentlichen Beobachten. Dabei umfasst es sowohl die Suche nach einer optimalen Position als auch die Navigation zu ebendieser Pose.

Finde eine geeignete Beobachtungsposition

Eine gute Beobachtungsposition wird hier durch drei Eigenschaften charakterisiert: Eine lange Beobachtungsdauer, eine hohe Beobachtungsqualität und einen geringen Störfaktor für die Person (Abb. 3.3 oben links). Der Roboter soll eine Position finden, die möglichst dauerhaft genutzt werden kann, auch wenn die Person für einen

Zeitabschnitt nicht beobachtbar ist. Die Qualität der Beobachtung soll währenddessen gut genug sein, um die Person und ggf. deren Gestik zu erkennen. Zuletzt soll der Roboter so stehen, das er den Nutzer nicht weiter stört, d. h. zu nahe ist oder Wege blockiert. Der Nutzer erwartet hier, dass der Roboter ihm nicht ständig folgt, weshalb häufige Neupositionierungen vermieden werden sollen. Das Finden einer geeigneten Position ist dabei ein Problem, welches aus mehreren Kriterien besteht. Im Folgenden werden einige dieser Kriterien genant. Als erstes ist die Erreichbarkeit zu erwähnen. Sie beschreibt, ob eine Position angefahren werden kann. Als nächstes folgt die Sichtbarkeit der Person. Diese muss von der Position aus durch die Sensoren wahrnehmbar sein. Steht der Roboter beispielsweise mit seiner Kamera frontal vor einer Wand, dann kann er die Person nicht beobachten. Für die Sichtbarkeit wird eine Posenhypothese bzw. eine Aufenthaltswahrscheinlichkeit vorausgesetzt. Es folgen weitere Kriterien wie die Einhaltung einer sozialen Distanz oder die Ausrichtung von Person zu Roboter bzw. umgekehrt Weitere Details zu den Kostenfunktionen folgen im Stand der Technik (Kap. 4).

Fahre die Position an

Während der Anfahrt soll weiterhin nach einer guten Position gesucht werden. Veränderungen in der dynamischen Umwelt können die Erreichbarkeit beeinträchtigen oder bessere Positionen ermöglichen. Außer einer optimalen Positionssuche, benötigt man ein Navigationsverhalten (Pilot), welches den Roboter zur Zielposition fährt.

Beobachte die Person

Der Roboter soll die Bezugsperson weiter überwachen, auch wenn er im Moment nicht von ihr genutzt wird. Er wartet an seiner Beobachtungsposition, solange er innerhalb eines festen Zeitfensters die Aktivität der Person registriert. Die Erkennung einer Person erfordert im einfachsten Fall eine Posenhypothese. Statt nur die Anwesenheit zu beobachten, kann die Aktivität auch mit Gestik-/Mimikerkennung arbeiten. Damit könnte der Roboter u. a. unterscheiden, ob die Person gerade schläft oder welchen Gemütszustand sie im Moment hat. Es wäre wünschenswert, dass sich das Zeitfenster

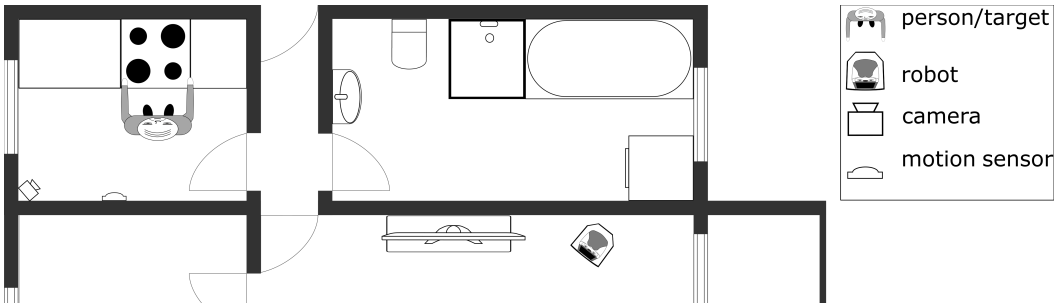


Abbildung 3.4: Beobachtung durch Hausautomation
Ist der Roboter im vernetzten Zuhause im Einsatz, so könnte er auch auf vorhandene Infrastruktur wie Kameras oder Bewegungssensoren für die Beobachtung zurückgreifen. So ist es vorstellbar, dass er während der Zeit in einen Standby-Betrieb übergeht und so den Energieverbrauch reduziert.

bei der Beobachtung verändern lässt. Dies könnte durch Lernen oder manuell durch den Nutzer geschehen. Damit sollen Situation vermieden werden, bei denen der Nutzer auf dem Rückweg von einer Tätigkeit ist und ihm der Roboter begegnet. Der Roboter könnte dann z. B. im Weg stehen.

Lasse die Person beobachten

Ist der Roboter ein Teil eines Smart Home Systems ([Gross et al.]), so soll er die Beobachtung auch in Kooperation mit diesem durchführen können (Abb. 3.4). Es ist auch vorstellbar, dass man in diesem Rahmen ein Fitnessarmband als weitete Informationsquelle hinzuzieht. Ist die Person nicht durch den Roboter sichtbar, so liefern Fitnessarmbänder Rückschlüsse über die tatsächliche Aktivität des Nutzers in dieser Zeit. Mit solchen Armbändern können Schrittzahlen, Vitalwerte (Puls) und Schlafinformationen erhoben werden.

3.2.2 Handle (act)

Der Beobachtungsmodus ist die einseitige Absicht des Nutzers, jetzt nicht mit dem Roboter interagieren oder arbeiten zu wollen. Jedoch gibt es Gründe, weshalb dieser Modus auch selbstständig verlassen werden sollte. Genau diese listet der folgende

Abschnitt in Form von Funktionen auf. Sein Verhalten ist damit in erster Linie die Reaktion auf geschehene Ereignisse. Zu den wichtigsten dieser Aktionen gehört die Suche nach der Person, wenn diese für die Dauer einer festgelegten Zeit nicht mehr gesichtet wurde.

Suche die Person

Der Roboter soll die Person suchen, wenn er sie für längerer Zeit nicht mehr beobachtet hat. Dabei ist der Nutzer nur unerwartet lange an einem anderen Ort oder befindet sich in einer Notlage. In jedem Fall wird eine Personensuche ausgelöst (Abb. 3.5). Diese Suche wird dann entweder erfolgreich oder erfolglos beendet. Liegt ein Notfall vor, so muss der Roboter der Person helfen. Ist der Zustand der Person in Ordnung, so startet Beobachtungsverhalten wieder bei der Positionssuche. Kann die Person nicht gefunden werden, so wechselt das System in einen Ruhemodus oder löst einen Alarm aus. Die Suche kann erfolglos enden, wenn die Person unerwartet die Wohnung verlässt oder ein längeres Bad nimmt. Beim Baden ist es ausreichend, in einen Ruhemodus zu wechseln oder in gewissen Zeitabständen mit dem Nutzer Kontakt aufzunehmen, z. B. durch Sprache. Fälle, bei denen der Roboter einen Alarm auslösen soll, müssen noch spezifiziert werden. Für das Suchverhalten ist die Erkennung einer Posenhypothese notwendig. Dabei wird auch eine Wiedererkennung benötigt, damit die richtige Person gesucht wird. Außerdem kann wird die Erkennung eines Sturzes benötigt, wie im Kapitel 3.1.5 zuvor beschrieben. Steht eine Aufenthaltswahrscheinlichkeit zur Verfügung, so kann der Roboter diese bei der Suche miteinbeziehen, indem er Orte mit hoher Wahrscheinlichkeit zuerst absucht.

Spiele mit der Person

Der Roboter soll auf spielerische Impulse reagieren können. Der Benutzer neigt durchaus dazu, die Fähigkeiten des Roboters auf die Probe zustellen. Wie im Kapitel 3.1.4 kann der Roboter nicht auf diesen Spielwunsch reagieren. Dieses spielerische Verhalten der Person ist jedoch auch eine eigene Art der Interaktion mit dem Roboter, die gefördert werden sollte. Dafür wird ein gesondertes Spielverhalten benötigt, dass den

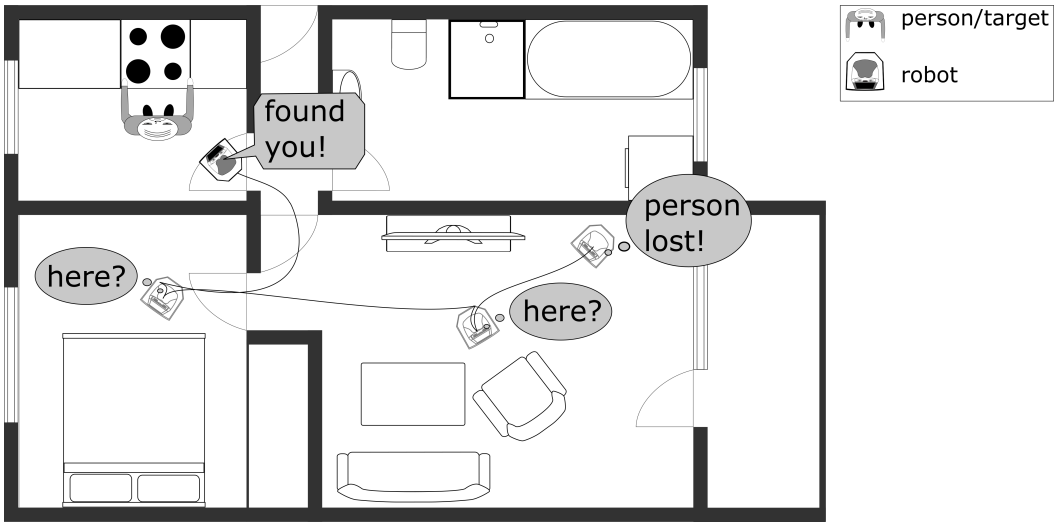


Abbildung 3.5: Suche nach der Person

Mit Hilfe einer Aufenthaltswahrscheinlichkeit werden zuerst Orte mit hoher Wahrscheinlichkeit abgesucht. In diesem Fall sucht der Roboter zuerst beim Sessel und dann im Schlafzimmer. Danach findet er die Person in der Küche.

Spielwunsch erkennt und ein ggf. vorprogrammiertes Spiel mit der Person spielt. Dabei muss es die Möglichkeit geben, dass der Wunsch auch im Beobachtungsmodus erkannt wird.

Fördere die Interaktion mit der Person

Der Roboter soll selbstständig mit der Person interagieren, wenn diese ihm über einen langen Zeitraum keine Aufmerksamkeit schenkt. Das steht zuerst einmal im Gegensatz zum Beobachtungsverhalten. Jedoch kann dieses Verhalten missbraucht werden, um sich der Technik zu entziehen. Die Person aktiviert ständig das Beobachtungsverhalten, damit sie sich nicht mit dem Roboter auseinandersetzen muss. Ein soziale Assistenzroboter ist jedoch kein Selbstzweck. Die Kooperation mit dem Menschen ist essentiell, da der Roboter im häuslichen Umfeld nur dem Wohl des Menschen dienen soll. Deswegen sollte diese Verhalten eine proaktive Komponente besitzen, damit der Nutzer zur Zusammenarbeit angeregt wird. Dadurch könnte die Eingewöhnungszeit bei der Person verkürzen werden. Indem das System die Initiative ergreift, kann er

den Nutzer leiten und so Vertrauen aufbauen. Entscheidend ist jedoch der richtige Zeitpunkt und die richtige Handlung. Ansonsten wird das Verhalten vom Nutzer nicht akzeptiert werden. Steht die Roboterplattform allein, so muss ein geeignetes Verfahren zur Erkennung von Langeweile entwickelt werden, z. B. über Gestik-/Mimikerkennung. es werden auch geeignete Unterhaltungs- und Spielverhalten benötigt.

Es ist sinnvoll, dass so ein Verhalten in Kombination mit der vernetzten Wohnung eingesetzt wird. So kann der Roboter z. B. über den SmartTV herausfinden, ob gerade die Lieblingsserie des Nutzers läuft oder nur Werbung.

Höre auf die Person

Der Roboter soll das Beobachtungsverhalten beenden und zum Nutzer fahren, wenn er gerufen wird. Wenn es sich dabei um einen Notfall handelt, kann der Roboter Hilfe leisten. Unter Umständen möchte der Nutzer auch einfach nur mit dem Roboter reden, ohne das er zu ihm fahren soll. Dafür werden Spracherkennung für die Eingabe, sowie Ton-/Sprachausgabe für eine Antwort benötigt

Informiere die Person Auch wenn das Beobachtungsverhalten aktiv ist, muss der Nutzer über eingehende Anrufe oder baldige Termine informiert werden. Dieser Informationsdienst kann auch auf medizinische Informationen übertragen werden. Verwendet die Person ein tragbares Pulsgerät wie ein neuartiges Fitnessband, so kann der Roboter die Person z. B. beei niedrigen Blutdruck warnen. Voraussetzung dafür ist die Verbindung des tragbaren Gerätes mit dem Roboter.

Umgang mit einer schlafenden Person Es kann vorkommen dass die Bezugsperson z. B. im Sessel vor dem Fernseher einschläft. Auf einen solchen Fall sollte der Roboter entsprechend reagieren können. Ist das System mit einer Hausautomation verbunden, so könnte es den Fernseher ausschalten und das Licht dimmen. Verfügt die Person über tragbare Geräte zu Schlafanalyse kann eine geeignete leichte Schlafphase abgewartet werden, um den Nutzer möglichst sanft zu wecken. So hat die Person die Möglichkeit seinen Schlaf im Bett fortzusetzen.

3.2.3 Lerne (adapt)

Um die Anpassung an den Nutzer zu verbessern, sollte der Roboter sein Verhalten und seine Positionssuche anpassen können. Für diesen Zweck werden im Folgenden einige Funktionen vorgestellt.

Lerne vom Nutzer gut akzeptierte Positionen

Der Roboter soll Positionen erlernen, die bei der Bezugsperson hohen Zuspruch genießen. Jeder Mensch hat seine eigenen Kriterien, mit denen er eine gute Position bewerten kann. Auch die Vertrautheit mit dem Roboter hat einen großen Einfluss auf den Umgang mit dem System. Nutzer mit wenig Erfahrung können sich eher vom Verhalten des Roboters bedroht fühlen und möchten deshalb mehr Abstand. Erfahrenere Nutzer lassen auch zu, dass der Roboter außerhalb des Sichtbereichs rangiert und steht, ohne sich ständig Gedanken zu machen. Außerdem gibt es Wartepositionen, die der Nutzer nicht akzeptiert. Vor dem eingeschalteten Fernseher zu parken sollte vermieden werden. Wenn die Person merkt, dass der Roboter von ihm lernt, kann das das Vertrauen in die Technik erhöhen und Hemmschwellen in der Nutzung abbauen. Des Weiteren vermittelt die Technik damit kein Handeln gegen den Willen des Nutzers. Um dieses Lernen zu ermöglichen benötigt man einerseits eine entsprechende Kostenfunktion in der Positionssuche und anderseits eine Eingabemöglichkeit für Kritik (*Feedback*). Diese kann z. B. durch Berührung, Sprache oder Fernbedienung geschehen.

Lerne ein gutes Zeitintervall bevor nach der Person gesucht wird

Der Roboter soll das Zeitfenster bis zur Aktivierung der Suche nach der Person variabel verändern können. Da Menschen unterschiedlich viel Zeit für ähnliche Tätigkeiten benötigen, muss auch die Wartezeit entsprechend angepasst werden können. Des Weiteren kann es Situationen geben, in denen der Roboter keine passende Position mit Blick auf die Person finden kann. Dann sollte der Roboter das Zeitfenster vergrößern, so dass der Roboter immer seltener den Zustand der Person überprüft.

Erfrage den Grund für Verhaltensänderungen

Durch Maßnahmen wie eine zeitabhängige Aufenthaltskarte lassen sich gute Positionen vorhersagen. Dies stimmt jedoch nur, solange sich die Person an einen ähnlichen Tagesablauf hält. Änderungen im Verhalten können willkürlich oder durch andere Einflüsse bedingt werden. Wird also eine Suche nach der Person ausgelöst, so kann der Roboter den Grund für das Abweichen vom Verhalten erfragen. Das kann den Eindruck erwecken, dass der Roboter sich um die Person kümmert und sorgt.

Kapitel 4

Stand der Technik

Im folgen Kapitel wird der *State of the Art* in der Thematik präsentiert. Zuerst wird eine Übersicht zu verschiedenen Arbeiten geben. Dann werden bekannte Problemstellungen vorgestellt, die einen Bezug zum Thema haben. Darauf folgt eine Präsentation von ausgewählten Artikeln, welche u. a. auch aus dem Fachgebiet NIKR stammen. Zum Schluss des Kapitels folgt eine Zusammenfassung und ein Ausblick auf die weiteren Schritte.

4.1 Übersicht

Nach [Kruse et al., 2013] handelt es sich bei der Positionssuche (*pose selection*) um die Aufgabe einen geeigneten Ort zu finden, an dem ein Roboter halten und mit der Umwelt interagieren kann. Sei es um mit Menschen zu kommunizieren, ein Glas mit einem Arm aufzunehmen oder einen Ruheplatz zu finden. Dabei werden viele Anforderungen an die Pose gestellt. Solche sind z. B. die Erreichbarkeit, verschiedene Distanzmaße, die Ausrichtung und die Sicht. Welche Kriterien zum Einsatz kommen, hängt im wesentlichen vom gewählten Missionsziel ab.

[Byers et al., 2003] beschäftige sich mit einem autonomen Photographen. In dieser Arbeit wurde der Schwerpunkt auf eine gute Bildkomposition gelegt. Verschiedene Faktoren für eine gute Komposition wurden erarbeitet und getestet. Erkennt der Roboter Hautfarbe im Kamerabild, so wird diese Stelle zu seinem Ziel. Er wird nun

versuchen, eine geeignete Position im Umkreis seines Ziels zu finden um dort Fotos zu schießen. Diese Stelle entspricht dabei einer Posenhypothese, die beobachtet werden soll. Entsprechend seines Ziels berücksichtigt er dann weitere Kriterien wie den richtigen Abstand, Überlagerungen der Gesichter von verschiedenen Personen und weiterer photographischer Merkmale. Auf eine ähnliche Weise betrachtete [Schroeter et al., 2009] in seiner Arbeit die Aspekte für einen autonomen Kameramann. Dabei untersuchte er den Einfluss von Licht bei der Positionssuche. Konkret handelte es sich dabei um geeignete Positionen zur Videotelefonie und zur Beobachtung. Er entwickelte dafür eine Kartierung von Lichtquellen. Dadurch wird die Blendung auf dem Kamerabild und auf dem Bildschirm des Roboters vermieden (Kap. 4.3.1). Das trägt dazu bei, dass das Gesicht für die Videotelefonie und bei der Beobachtung der Mimik gut erkennbar ist. Für die Suche nach der Position setzt er eine *PSO* ein.

[Dautenhahn et al., 2006] und [Koay et al., 2007] untersuchten, wie die Handlungen des Roboters wahrgenommen werden. Die Aufgabe war dabei die Annäherung an eine Person mit der Absicht dieser etwas zu übergeben. Neben der Endposition spielte auch die Anfahrt und die Geschwindigkeit eine wichtige Rolle. Bei der Position untersuchte man vor allem die Auswirkungen der sozialen Distanz auf die Akzeptanz des Nutzers. Dabei wurden feste Positionen und Anfahrverhalten vorgegeben. Die Resultate verwendete [Kessler et al., 2011], um ein Modell für die Distanz abzuleiten.

In [Kessler et al., 2012] und [Kessler et al., 2013] beschäftigt sich der Autor mit der Beobachtungsposition in der Nähe des Nutzers. Dabei wird neben der Posenhypothese auch erstmals eine Aufenthaltswahrscheinlichkeit genutzt. Diese beschreibt einen Bereich, in dem sich die Person aufhalten könnte. Daher sollte der Roboter so stehen, dass er diesen Bereichen ebenfalls beobachten kann (Kap. 4.3.2). Bei der Suche wurde ebenfalls auf die Optimierung durch eine PSO zurückgegriffen. Einen anderen Ansatz zum Finden einer geeigneten Warteposition verfolgte Kitade in [Kitade et al., 2013]. Dort soll ein robotischer Einkaufshelfer eine Position im großen Flur eines Einkaufszentrums finden, bei der er andere Kunden nicht behindert, während er auf die Rückkehr seines Nutzers wartet (Kap. 4.4). Statt einer Optimierung verwendete er eine Klassifizierung durch eine *Support Vector Machine (SVM)*.

Im Artikel von [Ishara et al., 2015] steht die Beobachtung einer Person bei der

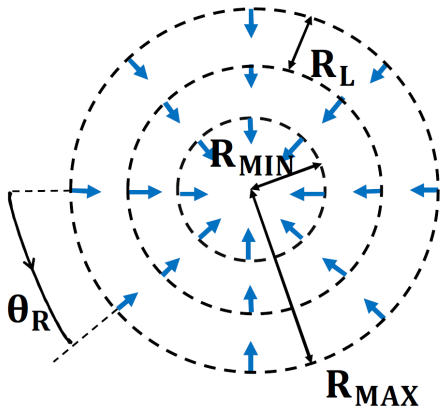


Abbildung 4.1: Beschreibung der Positionsauswahl
*Hier wird die Position nur in einem Kreisbereich um die Person (in der Mitte) gesucht. Dabei entsprechen die blauen Pfeile den Positionskandidaten. Das heißt, das der Abstand immer um R_L erhöht wird, bei R_{MIN} beginnt und nicht über R_{MAX} hinaus geht. Der Winkel wird ebenfalls nur in Schritten der Größe θ_R erhöht. Quelle: [*Ishara *et al., 2015]*

Durchführung mehrerer Übungen im Vordergrund. Der Roboter musste sich hierbei immer neu positionieren, um die beste Sicht zu erhalten. Um das zu ermöglichen wurde im Vorfeld eine Abfolge von Übungen und deren ungefähre Zeitdauer einprogrammiert. Beim Wechsel der Übung suchte sich der Roboter eine neue Position, wobei der Suchbereich auf einen Kreis um die Person beschränkt wurde (Abb. 4.1). Eine ähnliche Idee verfolgten [Ito und Sekiyama, 2015], wobei statt einer Person ein anderer Roboter beobachtet wurde. Das Ziel seiner Arbeit war es, den anderen Roboter besser steuern zu können. Dabei handelte es sich um Systeme im Katastropheneinsatz, die beispielsweise bei einem Reaktorunfall im Atomkraftwerk wie in Fukushima zum Einsatz kommen. Die Roboter wurden von Menschenhand gesteuert und besitzen u. a. nur eine fixierte Kamera mit eingeschränkter Neige- und Schwenkfähigkeit. Ein häufig auftretendes Problem ist die Verdeckung durch Objekte, so dass der Anwender nur auf Vermutung arbeitet und das die mentale Belastung steigert. Durch einen zweiten Roboter kann eine differenzierte Sicht auf die Situation gegeben werden. Auch hier spielt die Bildkomposition die entscheidende Rolle bei einer neu gewählten Position. Die Komposition wird anhand verschiedener Parameter geschätzt und dann mit einer SVM klassifiziert. Neben der Bildkomposition wird auch die Distanz zu den möglichen Positionen verwendet, sowie ein Bereich mit hoher Relevanz (*region of interest (ROI)*). Ähnlich wie in Abbildung 4.1 wählt man Kandidatenpositionen im Umkreis um den Interessenbereich. Dabei werden die genannten Faktoren kombiniert und die Position

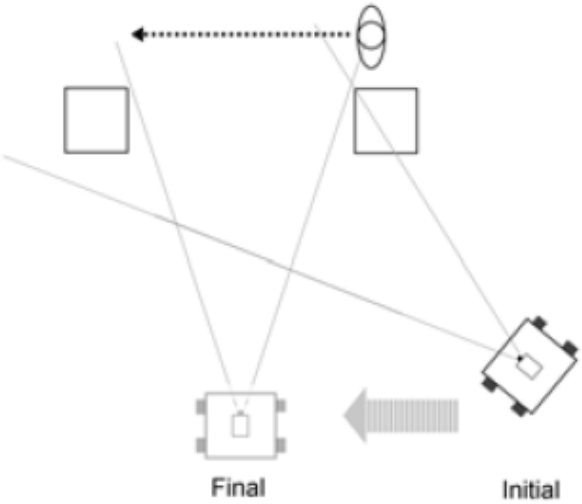
mit der höchsten Übereinstimmung in der SVM wird gewählt.

Abseits der mobilen Roboter finden sich noch weitere Artikel mit einem Bezug zur Thematik. In [Byers et al., 2003] beschäftigte sich der Autor mit dem Finden geeigneter statischer Positionen zur Installation von Kameras. Dabei sollten die Bewegungen und die Aktivitäten von Passanten in einem bestimmten Bereich überwacht werden. Eins der möglichen Anwendungsfelder ist die Überwachungstechnik. Die Personen können sich frei und auf unterschiedlichen Wegen durch das Bild bewegen. Das Ziel war es nun, die Beobachtung für alle möglichen Passanten zu maximieren (*joint observability*), damit eine Aktivitätserkennung erfolgen kann. Wesentlich ist dabei die Verteilung der Laufwege, die von folgenden Kriterien genutzt wird. Es wird die Distanz zu den Laufwegen (Trajektorien) der Passanten berechnet. Diese sollte groß genug sein, um den gesamten Laufweg betrachten zu können und gering genug, so dass die Person in einer möglichst hohen Auflösung im Kamerabild erscheint. Zusätzlich muss beachtet werden, dass alle Pfade innerhalb des Sichtkegels der Kamera liegen müssen. Als nächstes spielte der Blickwinkel zwischen Trajektorie und Kamera eine Rolle. Schaut die Kamera nicht senkrecht auf den Pfad (*foreshortening*), so wird die Person im Bild immer kleiner, da sie sich weiter entfernt. Für die Positionssuche wurde eine Kamera auf einem Roboter montiert und in mehreren Schritten optimiert (Abb. 4.2). Dabei kam der nichtlineare Optimierer *fmincon* aus der MATLAB Optimierungsbibliothek zum Einsatz.

4.2 Verwandte Probleme

Zur Vollständigkeit werden im Folgenden bekannte Problemstellungen im Zusammenhang mit der Positionssuche genannt. Dabei handelt es sich um das *Art Gallery Problem* und das *Next Best View Planning (NBV)*.

Beim Art Gallery Problem [O'Rouke, 1987] geht es darum, eine gegebene Umgebung durch den Einsatz von Kameras oder Wachmännern zu beobachten. Ziel ist es, die Überwachung durch möglichst wenig Mittel zu realisieren. Das heißt konkret, dass die Anzahl an Kameras bzw. Wachmännern sowie die Überschneidung von deren Sichtbereichen minimiert werden soll. Unser Problem lässt sich als Teilproblem darstellen. Wir



(a) Versuchsaufbau

(b) Bild von der initialen Position

(c) Bild nach optimierter Position

Abbildung 4.2: Optimierung der Beobachtungsposition
*Zu sehen ist ein Versuchsaufbau für Tests. In 4.2a ist der Ablauf zu sehen. Abbildung 4.2b zeigt das Bild von der Anfangsposition aus. Man bemerkt, dass der Blickwinkel noch nicht optimal ist. Nach der Optimierung liefert der Roboter das Bild aus Abbildung 4.2c. Man erkennt deutlich, dass der Roboter nun senkrecht auf den Laufweg schaut. Quelle: [*Bodor *et al., 2005].*

versuchen mit einem System die beste Position zu finden. Die Problemstellung wird meist als Graphenproblem formuliert, bei dem eine Kamera möglichst viele Knoten des Graphens sehen soll.

Das NBV [Scott et al., 2003] beschäftigt sich mit der Rekonstruktion und Digitalisierung von dreidimensionalen Objekten. Der Planungsaspekt realisiert dabei die geringste mögliche Anzahl an Beobachtungspositionen, die für die korrekte Abbildung notwendig sind. Ursprünglich zur Inspektion und Überprüfung von hergestellten Bauteilen eingesetzt, lässt sie sich auch im Kontext von Mensch-Maschine-Interaktion benutzen. Dabei lässt sich eine (zeitlich abhängige) Folge von Beobachtungspostionen für die Ausführung von Übungen realisieren, wie bei [Ishara et al., 2015] aus dem vorherigen Abschnitt. Auf unseren Anwendungsfall lässt sich auch hier wieder ein Teilproblem konstruieren: Finde die nächste Position, die unsere Bewertungsfunktion maximiert.

4.3 Positionssuche durch Optimierung

Am Fachgebiet NIKR gab es bereits Arbeiten mit einem ähnlichen Schwerpunkt. Auf ebendiese wird im folgenden Abschnitt eingegangen. In chronologischer Abfolge beschäftigen wir uns erst mit dem *Autonomous Robot Cameraman* ([Schroeter et al., 2009]) und darauffolgend mit *Approaching a person in a socially acceptable manner* ([Kessler et al., 2013])

4.3.1 Autonomous Robot Cameraman

In diesem Artikel beschäftigt sich [Schroeter et al., 2009] mit einer aufgabenspezifischen Positionssuche. Er hebt folgende Aufgaben hervor: Das Führen einer Videokonferenz und das Finden einer passiven Beobachtungspostion (siehe Abb. 4.3) Dabei stellt er ein Beleuchtungsmodell in den Vordergrund, denn bei einer Videokonferenz muss die Person und der Bildschirm gut erkennbar sein. Es muss eine Balance zwischen zu wenig und zu viel Beleuchtung gefunden werden. Damit will er die Schwäche der Kameras umgehen, die Szenen mit hohem Kontrast nicht richtig abbilden.

Es folgen nun weitere Aspekte, die in der Arbeit berücksichtigt wurden. An ers-

(a) Positionen mit und ohne Beleuchtungsmodell

(b) Schlechte Beleuchtungssituation

(c) Gute Beleuchtungssituation

Abbildung 4.3: Ergebnis der passiven Beobachtungsposition

*Zu sehen sind die Ergebnisse für die Suche nach guten passiven Beobachtungspositionen. Es wurden Tests mit (rot) und ohne Beleuchtungsmodell (grün) durchgeführt und in 4.3a präsentiert. Der schlechtere Ausblick aus den grünen Bereichen ist in 4.3b dargestellt und in 4.3c entsprechend der gute Ausblick mit passender Beleuchtung. Quelle: [*Schroeter *et al., 2009].*

ter Stelle folgt eine Modellierung des Freiraums, denn eine ausgewählte Position muss auch erreichbar sein. Dieser wird wiederum durch sogenannte *No-Go-Bereiche* eingeschränkt, die der Roboter nicht betreten darf. Daraufhin folgen Funktionen, die direkt von einer Posenhypothese abhängen: Der Roboter-Personen-Winkel und der gegensätzliche Personen-Roboter-Winkel. Der Erstere beschreibt dabei die Blickrichtung des Roboters unter Berücksichtigung der Position der Person. Bei einem Wert von null schaut der Roboter direkt in die Richtung der Person. Damit lässt sich bewerten, ob der Roboter die Person sehen kann. Der gegensätzliche Winkel beschreibt beschreibt die Sicht in umgekehrter Richtung. Für das Finden einer Beobachtungsposition wurde die Distanz zur nächsten Wand hinzugefügt, um eine Position zu finden, die den Nutzer nicht bedrängt. Weiterhin ist die Größe des Gesichts im Verhältnis zum Rest des Bildes für eine gute Bildkomposition wichtig. Zuletzt werden diverse Kostenfunktionen mit Bezug auf die Beleuchtung eingeführt, wie z. B. die Erkennbarkeit des Gesichts vom Hintergrund.

Alle Kriterien werden gewichtet in einer Summe verrechnet (Gl. (4.1)). Dabei stellt q_i das jeweilige Kriterium dar auf das eine aufgabenspezifische Funktion f_i^{Task} angewendet wird. Dies geschieht, da für verschiedene Aufgaben auch verschiedene Wertebereiche eines Kriteriums interessant sind. Für eine Videokonferenz ist eine Position wichtiger, die das Gesicht im vollen Umfang abbildet, während bei einer Beobachtung ein weiterer Abstand möglich ist. Entsprechend der Aufgabe haben Funktionen auch eine unterschiedliche Wichtigkeit. Steht ein Roboter abseits und beobachtet, so ist in diesem Fall ein gut lesbarer Bildschirm weniger relevant. Um dies mit einzubeziehen wird die Funktion f_i^{Task} durch w_i^{Task} gewichtet, welches wieder vom Kriterium und der Aufgabe abhängt.

$$Q^{Task}(P_{vid}) = \sum_i w_i^{Task} \cdot f_i^{Task}(q_i(P_{vid})) \tag{4.1}$$

$$P_{vid}^{opt} = \underset{P_{vid}}{\operatorname{argmin}}(Q^{Task}(P_{vid})) \tag{4.2}$$

Als Optimierungsverfahren ist die *Particle Swarm Optimization (PSO)* im Einsatz.

Dabei wird ein Set aus Partikeln künstlich bewegt und bezüglich der Gleichung (4.1) ausgewertet. Es wird dabei das Argument P_{vid}^{opt} gesucht, welches die Gleichung (4.2) minimiert. Dabei ist $P_{vid} = (x, y, \varphi, \gamma)$ eine Kombination aus der Pose im zweidimensionalen Raum (x, y, φ) und einem Blendenöffnungswinkel γ.

Im Vergleich zu den Arbeiten von Kessler im folgenden Kapitel ist die Erreichbarkeit bereits als binäres Kriterium definiert, aber geht zu gleichen Teilen in die Score-Funktion mit ein. Damit ist es zumindest theoretisch möglich, dass eine Position gewählt wird, die nicht erreichbar ist.

4.3.2 Finding a good Position to observe a Person

Auch die Arbeiten von [Kessler et al., 2012] und [Kessler et al., 2013] beschäftigen sich mit dem Problem einer optimalen Beobachtungsposition. Diese beschreiben vier allgemeine Kriterien, die für eine gute Position erfüllt sein müssen:

1. Die Person muss sichtbar sein
2. Die Position sollte den Nutzer nicht stören
3. Die Roboter sollte eine geeignete Entfernung zum Nutzer haben (sensorische bzw. soziale Distanz)
4. Der Roboter sollte weitere, alternative Ruhepositionen sehen können (Aufenthaltswahrscheinlichkeit)

Damit ist im Gegensatz zu den Arbeiten aus dem vorherigen Abschnitt die Bildqualität nicht das wesentliche Kriterium. Es wird die geeignete zweidimensionale Position $\mathbf{x} = (x, y)$ gesucht. Bei den Kostenfunktionen kommen die Erreichbarkeit und die Sichtbarkeit als wesentliche Funktionen zum Einsatz. Auch eine sensortechnische Distanz spielt eine Rolle. Außerdem wurde basierend auf den Forschungen von [Hall, 1966] eine soziale Distanz eingeführt. Diese sorgt dafür, dass der Roboter nah genug für einen Interaktionswunsch ist, den Nutzer dabei aber nicht bedrängt. Zuletzt wurde auch eine Kostenfunktion zur Abdeckung einer Aufenthaltswahrscheinlichkeit entworfen. Mit ihr stellt man sicher, dass auch weitere Ruhepositionen sichtbar sind und sich

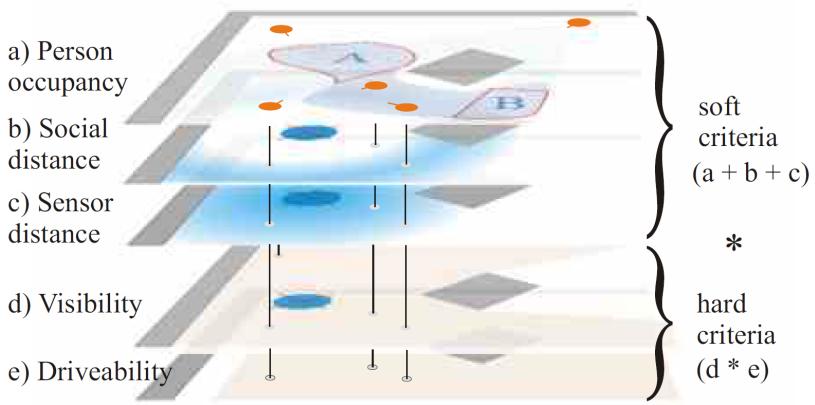


Abbildung 4.4: Zusammenspiel der Kostenfunktionen
*Zu sehen ist hier die Visualisierung der verschiedenen Kostenfunktionen. Dabei wird die Unterteilung in harte und weiche Kriterien verdeutlicht. Die weichen Kriterien umfassen: Aufenthaltswahrscheinlichkeit (a), soziale Distanz (b), sensortechnische Distanz (c). Die harten Kriterien umfassen Sichtbarkeit (e) und Erreichbarkeit (f). Quelle: [*Kessler *et al., 2012].*

der Roboter nicht neu positionieren muss. Der Roboter-Personen-Winkel beträgt immer null, da der Roboter von jeder Position immer genau in die Richtung der Person schauen soll. Der Personen-Roboter-Winkel soll minimiert werden, damit ein möglichst frontaler Blick für die Gestenerkennung genutzt werden kann. Zur Übersicht werden die Kriterien in Abbildung 4.4 dargestellt. In [Kessler et al., 2013] wird das Verfahren um ein Abstraktionsverfahren für die Sichtbarkeit erweitert, in dem eine Höhenkarte (*elevation map*) eingeführt wird. Diese bildet den dreidimensionalen Raum auf einen pseudo-zweidimensionalen Raum ab, in der jede Zelle die Höhe über einer Grundfläche repräsentiert.

Bei der Optimierung setzt man auf die PSO. Die Optimierungsfunktion (Gl. (4.3)) wurde erweitert, so dass sie nun harte Kriterien $h_i(\mathbf{x}, \phi)$ und weiche Kriterien $s_j(\mathbf{x}, \phi)$ unterschiedlich behandelt. Weiterhin ist die Ausrichtung ϕ vorgegeben, indem der Roboter von jeder Position direkt in die Richtung der Person schauen soll. Damit wird nur $\mathbf{x}$ optimiert (Gl. (4.4)). Harte Kriterien müssen stets erfüllt sein und liefern ein binäres Ergebnis. Deshalb werden sie als Produkt dargestellt und evaluieren ob eine Position

gültig ist oder nicht. Die weichen Kriterien werden als Summe von gewichteten Kriterien dargestellt und folgen hier den Vorarbeiten. Es wird jedoch auf aufgabenspezifische Parameter verzichtet.

$$f(\mathbf{x}, \phi) = \prod_i h_i(\mathbf{x}, \phi) \cdot \left[\sum_j \alpha_j s_j(\mathbf{x}, \phi) \right] \tag{4.3}$$

$$\mathbf{x}^{opt} = \underset{\mathbf{x}}{\operatorname{argmax}}(f(\mathbf{x}, \phi)) \tag{4.4}$$

Hierbei spielt es eine bedeutende Rolle, dass der Nutzer ständig beobachtbar ist. Ohne eine Posenhypothese kann das Verfahren nicht arbeiten.

4.4 Positionssuche durch Klassifikation

Aufgrund der thematischen Nähe beschäftigen wir uns im nun folgenden Abschnitt mit der Arbeit von [Kitade et al., 2013]. Dieser setzte den Fokus auf geeignete Wartepositionen in einem Einkaufszentrum. Dort spielten ähnliche Kriterien eine Rolle wie in der zu bewältigen Aufgabe, jedoch kam statt einer Optimierung eine Klassifizierung durch eine *Support Vector Machine (SVM)* zum Einsatz. Sie untersuchten geeignete Wartepositionen in einem Einkaufszentrum, welches aus breiten Fluren und engen Geschäften besteht. Da der Roboter die Person nicht bis ins Geschäft begleiten kann, soll dieser draußen warten.

Um geeignete Positionen zu finden, wurden Labor-Mitarbeitern zu bestimmten Positionen und Situationen in einem Einkaufszentrum befragt. Sie sollten bewerten, welche Stellen mit Sicherheit gut oder schlecht geeignet sind. Daraus wurden Datensatz mit guten, schlechten und unbestimmten Positionen gewonnen. Dabei kristallisierten sich zwei Hauptaspekte heraus: Der Roboter darf einerseits die Passanten und anderseits den Ladenbetrieb nicht stören. Um das erste Kriterium zu erfüllen, wurden aufwendig die Trajektorien der Passanten über einen Zeitraum von fünf Stunden aufgezeichnet. Ein Abgleich mit dem Datensatz ergab, dass schlechte Positionen häufig von Menschen durchquert werden oder es sich z. B. um automatische Türen handelt. Gute Stellen ergeben sich hingegen in der Nähe von Pfeilern und Wänden. Daraus ergibt

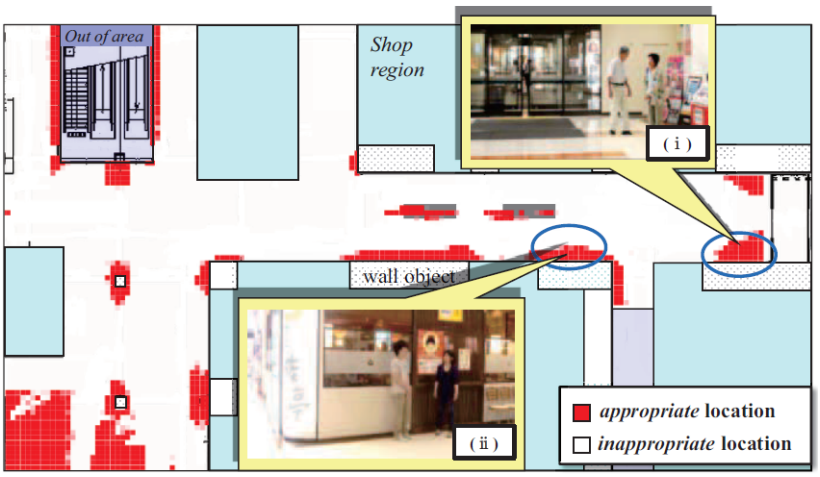


Abbildung 4.5: Gute Warteposition im Einkaufszentrum
*Im Bild zu sehen sind die Ergebnisse nach Klassifikation aller Zellen in der Karte. Gut zu sehen ist, wie gute Wartepositionen stets um Wände oder Stützpfeiler verteilt sind. Die Sichtflächen von Geschäften werden weiträumig gemieden. Quelle: [*Kitade *et al., 2013].*

sich eine Kostenfunktion, die von der durchschnittlichen Anzahl an Passanten an dieser Stelle ausgeht. Um das zweite Kriterium zu erfüllen, definierte man per Hand die Geschäftsbereiche und deren äußere Wandbereiche. Das ergibt sich aus der Tatsache, dass Türen und Schaufenster nicht blockiert werden dürfen. Wie schon zuvor eignen sich Wände und Pfeiler als gute Wartepositionen. Eine zweite Variante einer Kostenfunktion für den Ladenbetrieb ist die Sichtbarkeit des Geschäfts. Diese wurde mit einem Ray-Tracing realisiert, bei dem alle Stellen einer Karte ungeeignet sind, von der man mit weniger als einer festen Maximaldistanz in den Ladenbereich (durch Schaufenster oder Glastüren) sehen kann. Diese Kostenfunktion ergibt sich aus der Summe aller Zellen einer Geschäftsfläche, die von einer Position zu sehen ist.

Die Werte der Kostenfunktionen bilden für jede Position ein Tupel, mit dem die SVM trainiert wurde. Das Ergebnis wird in Abbildung 4.5 dargestellt. Dort ist gut zu erkennen, wie die Geschäftsbereiche gemieden werden. Um nun herauszufinden, ob eine Position geeignet ist, werden die entsprechenden Werte der oben beschriebenen Kostenfunktionen berechnet und als Tupel der SVM zugeführt.

Anhand des Prozesses zur Datengewinnung wird deutlich, dass sich dieses Verfahren vorwiegend für statische Situationen und spezielle Einsatzorte eignet. Dies wird auch in dem Artikel erwähnt. Das erlernte Modell ist an die Verhaltensweisen in der japanischen Kultur angepasst und die Karte des Einkaufszentrums musste manuell gezeichnet werden. Eine weitere Hürde bildet eine temporäre Änderung im Einsatzgebiet. Durch bauliche Maßnahmen können sich die Laufwege der Passanten ändern und so entscheidend auf die Bewertung geeigneter Positionen Einfluss nehmen. Auf diesen Umstand wird jedoch nicht näher eingegangen.

4.5 Zusammenfassung und Ausblick

In der Tabelle 4.1 sind noch einmal alle erwähnten Kostenfunktionen aus den hervorgehobenen Artikeln zusammengefasst. Es wird deutlich, dass es bei den unterschiedlichen Arbeiten viele Überschneidungen gibt. In dieser Arbeit werde ich mich stärker an den Werken des Fachgebietes orientieren, da es einen Zusammenhang zwischen den bisherigen und meiner Arbeit gibt.

Aufgrund des möglichen Umfanges wird sich die Arbeit auf grundlegende Aspekte einschränken. Es werden nur die Sensoren genutzt, die dem Roboter direkt zur Verfügung stehen. Weitere Optionen wie das Zusammenspiel mit Hausautomation oder Fitness-Trackern werden nicht ausgeschöpft. Zur Verfügung stehen in der Arbeit die in Abbildung 4.6 dargestellten und im folgenden aufgezählten Module und Informationen:

1. Eine Umweltrepräsentation durch zwei- und dreidimensionale Karten, dabei unter anderem eine NDT-Karte.

2. Eine zweidimensionale Kostenkarte unter Berücksichtigung der Erreichbarkeit von Positionen bezüglich des Roboterumfangs.

3. Eine zweidimensionale Feedback-Karte und eine Fernbedienung zur Feedbackgenerierung.

4. Eine zweidimensionale Karte der Aufenthaltswahrscheinlichkeit. Sie beschreibt Orte, an denen die Person häufig beobachtet wurde.

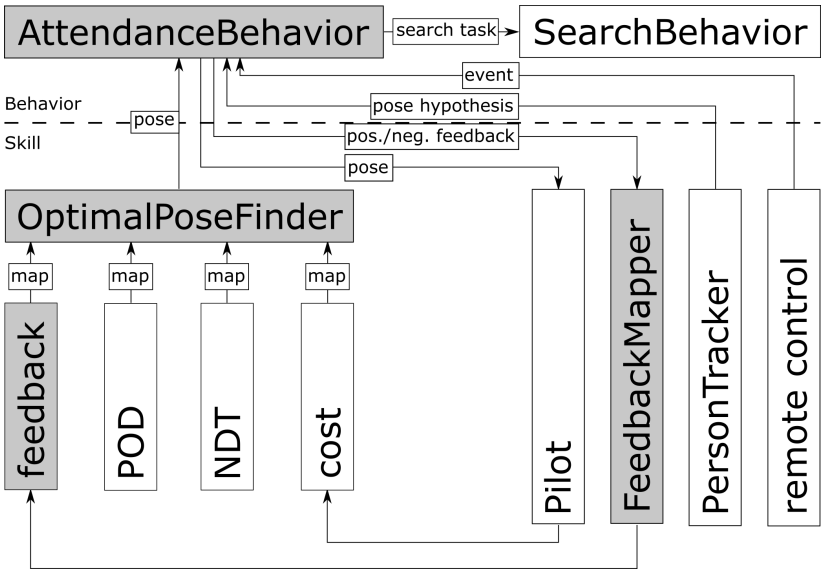


Abbildung 4.6: Blockdiagramm

Zu sehen ist die Ausgangslage bei dieser Aufgabe. Die grau hinterlegen Blöcke werden dabei während dieser Arbeit umgesetzt. Die optimale Positionssuche (OptimalPoseFinder) wird dabei durch verschiedene Arten von Karten versorgt. Das Beobachtungsverhalten (AttendanceBehavior) lässt sich durch Events einer Fernbedienung (remote control) beeinflussen.

5. Ein Navigationspilot für die Anfahrt einer Pose.

6. Ein Personen-Tracker zur Detektion und Verfolgung einer Person. Dabei nutzt er eine Personenhypothese.

7. Eine Personensuche, die verschiedene Aufenthaltsorte abfährt und auf eine Rückmeldung durch den Nutzer wartet.

Bezogen auf die Arbeiten am Fachgebiet NIKR ist ein Wandel zu beobachten. Bei den Arbeiten von [Schroeter et al., 2009] war eine Posenhypothese bei der Positionssuche zwingend erforderlich. Bei [Kessler et al., 2012] wurde zusätzlich eine Aufenthaltswahrscheinlichkeit mit einbezogen, um eine längerfristige Beobachtungsposition zu wählen. In meiner Arbeit soll nun gänzlich auf die Verwendung einer Posenhypothese bei der Positionssuche verzichtet werden. Der Grund dafür ist der erwünschte prädiktive Charakter des Verfahrens, der die zukünftige Position des Nutzers berücksichtigt, um ein simples Hinterherfahren zu vermeiden. Da viele der gezeigten Kostenfunktionen direkt von dieser Hypothese abhängen, müssen diese neu entworfen werden.

Was bisher noch nicht in Kriterien einfließt sind die Vorlieben jedes einzelnen Nutzers. Die Arbeiten von [Dautenhahn et al., 2006], [Walters et al., 2007] und [Koay et al., 2007] legen nahe, dass sich das Nutzungsverhalten eines Roboters durch Personen mit und ohne Erfahrung im Umgang mit Robotern unterscheiden kann. An dieser Stelle eignet sich das Feedback, um das Verhalten während der Laufzeit anzupassen, ohne das dies durch Wartungspersonal des Roboters umgesetzt werden muss. In dieser Arbeit wird daher die Möglichkeit eingeführt, dem Roboter für seine ausgewählte Position zu loben oder zu bestrafen.

	[Schroeter et al., 2009]	[Kessler et al., 2012]	[Kitade et al., 2013]
Erreichbarkeit	X	X	o
Sichtbarkeit	X	X	X
Aufenthaltswahrscheinlichkeit	o	X	o
Bewegungstrajektorien	o	o	X
Soziale Distanz	X	o	X
Sensortechnische Distanz	X	X	X
Roboter-Personen-Winkel	X	o	o
Personen-Roboter-Winkel	X	X	o
Größe des Gesichts	X	o	o
Wanddistanz	o	X	X
Blendung	X	o	o

Tabelle 4.1: Tabelle der eingesetzten Kostenfunktionen

Zu sehen ist ein Überblick über die verschiedenen Kostenfunktionen in den einzelnen Artikeln. Ein X *markiert das Vorhandensein der jeweiligen Kostenfunktion.*

Kapitel 5

Optimierung der Beobachtungsposition

Diese Kapitel beschreibt nun die Suche nach der optimalen Beobachtungsposition (*OptimalPoseFinder*) (Abb. 5.1). Dabei handelt es sich um eine generische Funktion, die an Hand bestimmter Kriterien eine Position findet. Dies geschieht durch geeignete Optimierungsverfahren. Es wird dabei eine zweidimensionale Pose $\mathbf{x} = (x, y, \varphi)$ optimiert. Dabei stehen folgende Informationen zur Verfügung: Die Kostenkarte der Navigation, die Karte der Aufenthaltswahrscheinlichkeit, die Karte des Feedbacks, eine NDT-Karte und die aktuelle Roboterpose. Die verschiedenen Karten wurden bereits im Kapitel 2 ausführlich erklärt, so dass hier nur die Nutzung im Vordergrund steht.

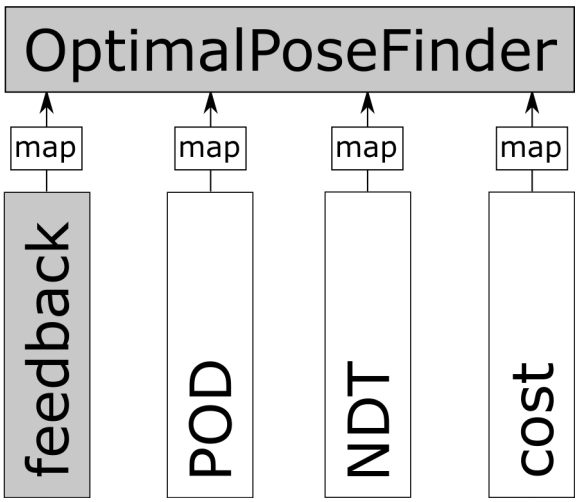


Abbildung 5.1: Informationsquellen für die Positionssuche
Hier ist zu sehen, dass sich die optimale Pose aus den Karten des Feedbacks, der Aufenthaltswahrscheinlichkeit (POD), der NDT-Zellen und der Kosten ergibt.

5.1 Aufbau und Ablauf

Die Positionssuche kennt zwei Phasen. Eine aktive und eine inaktive Phase. In der inaktiven Phase wartet sie darauf, dass sie von einem anderen Modul bzw. Verhalten des Roboters aktiviert wird. Ist sie aktiviert worden, so sucht sie kontinuierlich neue optimale Positionen für eine Beobachtung. Dabei kommt eine universelle Zielfunktion zum Tragen (Gl. 5.1). Sie beruht auf den vorherigen Arbeiten von Kessler, die im Kapitel 4.3.2 gezeigt wurden. Dabei wird die Funktion in die harten Kostenfunktionen $a_i(\mathbf{x})$ und die weichen Kostenfunktionen $v_j(\mathbf{x})$ unterschieden. Mit α_j wird den weichen Kostenfunktionen zusätzlich ein Gewichtungsfaktor zugeordnet. Dieser wird genutzt, um das Zusammenspiel der Funktionen auszubalancieren. Die harten Kostenfunktionen realisieren dabei einen Filter. Dieser akzeptiert eine Position nur dann, wenn sie alle Funktionen erfüllt hat. Ist das der Fall, so wird nun der Wert der Position durch die gewichtete und normalisierte Summe der weichen Kostenfunktionen bestimmt. Bei dieser Zielfunktion wird die Position $\mathbf{x}_{min}$ gesucht, die diese minimiert. Dabei handelt es sich um eine nichtlineare Funktion. Das bedeutet, dass lokale und globale Minima existieren. Für die Lösung benötigt es eine Optimierung mit Stichprobenverfahren. Der triviale Ansatz dafür ist die Lösung durch gewaltsames Durchprobieren (*brute force*) aller möglichen Kombinationen. Am Fachgebiet NIKR hat sich für diese Problemstellung die PSO (Kap. 2.3.3) etabliert, da sie geeignete Eigenschaften zur lokalen und globalen Optimierung besitzt. Dabei wird mit einer Menge an zufälligen Stichproben der Zielfunktion gestartet. Danach wird diese Menge iterative durch eine Mischung aus gelenkter und zufälliger Suche verbessert. Aufgrund dieser Eigenschaften wird sie auch in dieser Arbeit eingesetzt.

$$\mathbf{x}_{min} = \underset{\mathbf{x}}{\operatorname{argmin}} \prod_{i=1}^{n} a_i(\mathbf{x}) \cdot \left[\frac{1}{\sum_{j=1}^{m} \alpha_j} \sum_{j=1}^{m} \alpha_j v_j(\mathbf{x}) \right] \tag{5.1}$$

Die Formel mit harten und weichen Kostenfunktionen kann im Allgemeinen einen Laufzeitvorteil bringen. Dieser tritt dann ein, wenn harte Funktionen weniger Rechenzeit als die weichen Funktionen benötigen. Wird eine Kandidatenpose abgelehnt, so kann die Berechnung der weichen Kostenfunktionen übersprungen werden. Zum

Abschluss wird kurz die Zusammensetzung der universellen Zielfunktion aus den Kostenfunktionen beschrieben. Als hartes Kriterium gilt allein die Erreichbarkeit. Im Gegensatz zu [Kessler et al., 2012] wird die Sichtbarkeit nicht genutzt. Das hängt damit zusammen, dass er die Posenhypothese nutzen konnte. Dadurch war es ihm möglich, direkt in der Nähe der Person nach einer optimalen Position zu suchen. Für diesen Vorteil müsste man in der Aufenthaltswahrscheinlichkeit zuerst den aktuellen Bereich finden. Darauf wird jedoch verzichtet. Bei den anderen Kostenfunktionen handelt es sich somit um weiche Kriterien.

5.2 Kostenfunktionen

In Kapitel 4 wurde sich sehr ausgiebig mit den verschiedenen Arbeiten und deren Kostenfunktionen auseinandergesetzt. Daher folgen nun die Funktionen, die in dieser Arbeit verwendet wurden. Dabei wurde die Aufenthaltswahrscheinlichkeit geschickt genutzt, um die Posenhypothese und weitere Funktionen zu ersetzen. In Abbildung 5.2 sind diese zu sehen. Auf der linken Seite werden die ursprünglichen Kostenfunktionen gezeigt und auf der rechten Seite ihre neuen Gegenstücke. Mit der Karte der Aufenthaltswahrscheinlichkeit lassen sich verschiedenen Kriterien für eine gute Beobachtungsposition erfüllen. Sie enthält die Wahrscheinlichkeit, dass sich die Person zu einer gegebenen Zeit an einem bestimmten Ort aufhalten wird. Damit erhöht sich die Wahrscheinlichkeit, dass die Person gesehen werden kann. Des Weiteren kann auch die soziale Distanz (Kap. 2.3.2) zur möglichen Person mit beachtet werden. Diese beeinflusst, ob sich die Person durch die Nähe des Roboters unwohl fühlt. Das Missachten dieser Distanz könnte zur Folge haben, dass sich der Nutzer nicht auf die erwartete Position begibt. Dadurch würde die Beobachtungsposition nicht ihr Ziel erfüllen. Als Kostenfunktion muss auch die Sichtbarkeit verwendet werden. Der Blick auf die Person darf nicht versperrt sein. Da die Person zum Zeitpunkt der Optimierung noch nicht im Zielgebiet ist, muss eine Ersatzfunktion entwickelt werden. Ebendiese wird in Form einer allgemeinen Sichtbarkeit definiert. Eine weitere wichtige Rolle nimmt die Erreichbarkeit (Kap. 5.2.1) einer Position ein. Eine unerreichbare Beobachtungsposition ist nicht gültig. Die dabei genutzte Kostenkarte lässt sich auch noch in einem

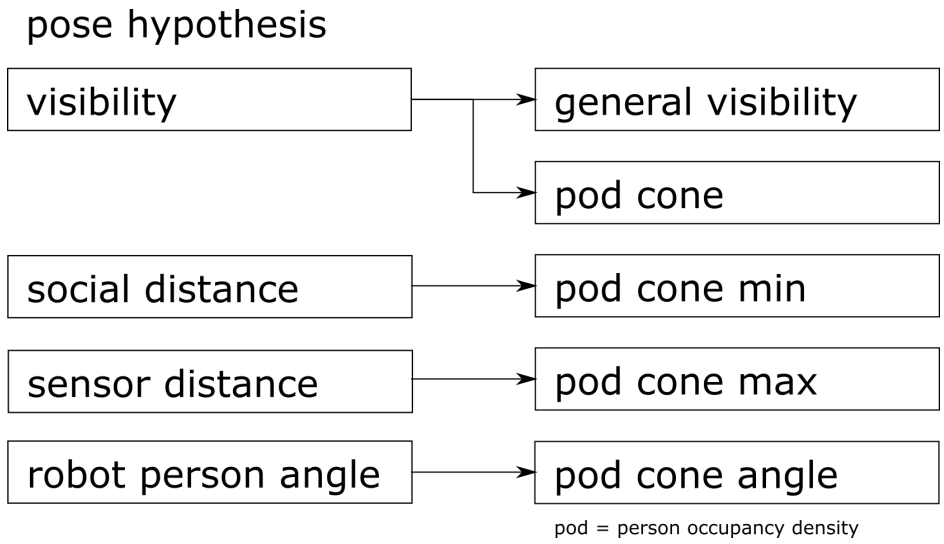


Abbildung 5.2: Abbildung auf neue Kostenfunktionen

Das Module in der linken Bildhälfte bildet aus Sicht der Implementation eine Brücke zum MIRA Framework. Die abstrakten Funktionen liefern die Schnittstelle, auf deren Basis konkrete Funktionen erarbeitet und dann als Plugins eingebunden werden. Mit dem Finder realisiert die Einbindung und Nutzung durch andere Komponenten im Framework.

weichen Kriterium weiterverwenden, welches den Roboter in einem gewissen Rahmen von Wänden fernhalten soll (Kap. 5.2.3). An letzter Stelle steht die Kostenfunktion zur Einbeziehung von Feedback. Das Feedback sorgt dafür, dass sich die Beobachtungsposition immer besser an die Wünsche des Nutzers anpassen kann.

5.2.1 Erreichbarkeit

Die Erreichbarkeit als hartes Kriterium ist zwingend erforderlich, denn eine Pose außerhalb der Karte oder inmitten einer Wand ist nicht zulässig. Dafür steht dem Roboter eine Kostenkarte (Kap. 2.2.1) zur Verfügung. Sie basiert auf einer Hinderniskarte und einer No-Go-Karte. In dieser Repräsentation sind alle als frei markierten Zellen durch den Roboter erreichbar. Dabei können jedoch insbesondere am Rand und inmitten der Karte plötzlich Einschlüsse mit freie Flächen entstehen. Um dies zu verhindern, wurde auf der Karte ein *Flood Fill* Algorithmus ausgeführt. Dieser markiert alle miteinander zusammenhängenden Zellen mit dem gleichen Status (frei), angefangen bei der Roboterposition.

5.2.2 Sichtbarkeit

Durch das Fehlen einer Posenhypothese ist die *Sichtbarkeit der Person* nicht mehr verfügbar. Da diese jedoch ein notwendiges Kriterium für die Beobachtung ist, wird als Ersatz eine neue *Sichbarkeit im Allgemeinen* entwickelt. Eine Person ist dafür nicht mehr relevant, vielmehr steht das Sichtfeld der Kamera jetzt im Fokus. Es soll nun die Hindernisfreiheit dieses Sichtfelds bestimmt werden. Damit beschreibt man nur noch indirekt die Möglichkeit, eine Person beobachten zu können. Durch ein freies Sichtfeld wird die Wahrscheinlichkeit erhöht, dass eine Person dort zu sehen sein könnte. Ist das Sichtfeld eingeschränkt, z. B. weil der Roboter aus nächster Nähe frontal auf eine Wand zeigt, so kann er dort auch keine Person sehen. Um die Sichtbarkeit zu bestimmen, benötigt man ein dreidimensionales Umweltmodell oder eine geeignete zweidimensionale Abbildung aus diesem. Zu diesen Modellen gehören dreidimensionale Voxel- oder NDT-Karten (Kap. 2.2.2). Die Lösung sollte mit Hilfe einer vorhandenen NDT-Karte umgesetzt werden und dabei im dreidimensionalen Raum arbeiten. Im

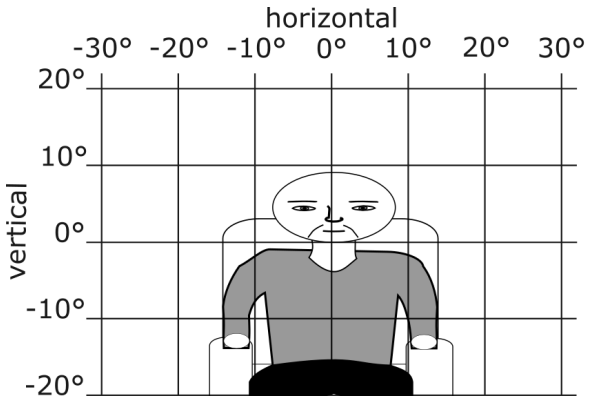


Abbildung 5.3: Dreidimensionale Sichtkegel

Im Bild ist die vereinfachte Unterteilung des Sichtkegels abgebildet. Dabei wird dieser von der Kameraposition in der Mitte gesehen.

Laufe der Arbeit wurde je eine Kostenfunktion für den zwei- bzw. dreidimensionalen Raum erarbeitet. Als erstes folgt nun die dreidimensionale Kostenfunktion.

3D

Wie bereits im Kapitel 2.2.2 erläutert, besteht eine NDT-Karte aus Zellen. Diese wiederum besitzen einen Mittelwertvektor und eine Kovarianzmatrix. Für die Kostenfunktion werden dafür nur die Mittelwerte eingesetzt. Dabei wird ein Sichtkegel aufgespannt, dem eine maximale Sichtweite d_{max} vorgegeben wird. Des Weiteren basiert dieser auf einer gewölbten, nahezu rechteckigen Maske, die durch die horizontalen und vertikalen Öffnungswinkel der Kamera bestimmt werden (Abb. 5.3). Die Fläche wird anhand einer Winkelauflösung *(bin size)* unterteilt, so dass gleichgroße Zellen entstehen. Danach wurde *ray tracing* zwischen der Kameraposition und den NDT-Zellen eingesetzt, wodurch sich eine Sichtgerade ergibt. Mit ihr bestimmt sich die Zugehörigkeit zum Flächensegment, sowie die minimale Distanz zur diesem. Dadurch ergibt sich ein Tiefenprofil, aus dem sich die Kostenfunktion ableitet. Der Sichtkegel wird als optimal bezeichnet, wenn die durchschnittliche Distanz maximal ist (vgl. Abb 5.4). Das bedeutet, dass der Sichtbereich des Roboters frei von Objekten ist. Daraus lässt sich die Gleichung (5.2) ableiten. Der Term $\min d_{ij}(\mathbf{x}, \mathbf{M}_{NDT})$ stellt den minimalen

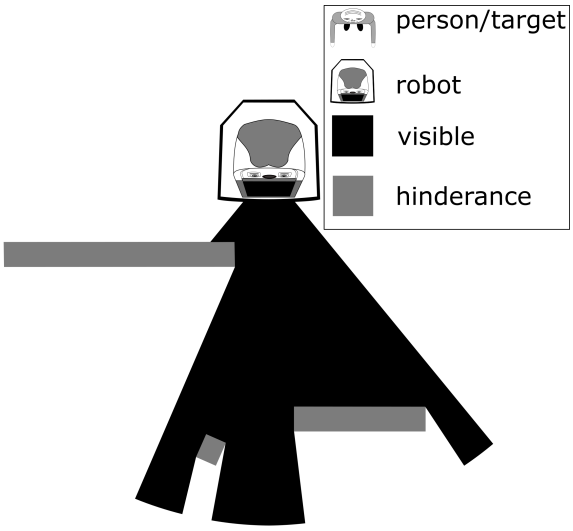


Abbildung 5.4: Sichtbarkeit in 2D

Hier wird der Sichtkegel abgebildet. Er wird dabei durch die maximale Distanz, den Hindernissen und den Öffnungswinkel abgegrenzt. Im optimalen Fall wird im gesamten Winkelbereich die maximale Distanz erreicht.

Abstand von Pose $\mathbf{x}$ in der NDT-Karte $\mathbf{M}_{NDT}$ zur Position i, j in der Maske dar. Alle Minima werden aufaddiert und durch den Term $\frac{1}{d_{max} \cdot (n+m)}$ normalisiert. Dabei beschreibt n und m die Anzahl der Flächen in horizontaler bzw. vertikaler Richtung.

$$f(\mathbf{x}, \mathbf{M}_{NDT}) = \frac{1}{d_{max} \cdot (n+m)} \sum_{i,j} \min d_{ij}(\mathbf{x}, \mathbf{M}_{NDT}) \tag{5.2}$$

Durch die vielen Triangulationen bei jeder überprüften Pose entsteht ein hoher Rechenaufwand. Für die Effizienzsteigerung kam daher die *Fast Library for Approximate Nearest Neighbors (FLANN)* zum Einsatz. Sie beschleunigt die Suche nach allen relevanten NDT-Zellen im Umkreis der zu prüfenden Position.

2D

Aufgrund des Rechenaufwands im dreidimensionalen Fall wurde eine Lösung im zweidimensionalen Raum gesucht. Diese ergibt sich als Abstraktion aus den vorher genannten Aspekten. Für diese Zwecke wird eine zweidimensionale Karte berechnet. Um dreidi-

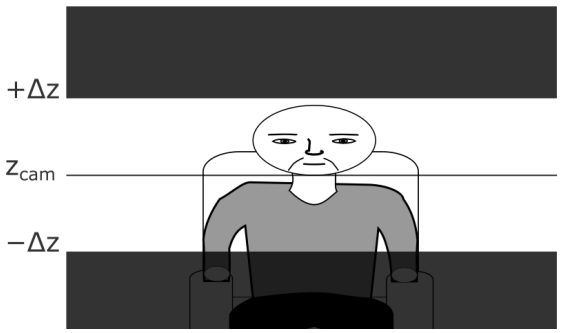


Abbildung 5.5: Eingeschränkter Sichtraum
Bei der Abstraktion werden alle NDT-Zellen in der schwarzen Schicht ignoriert und nur die aus der weißen Schicht in die zweidimensionale Sichthinderniskarte eingetragen. Bestimmt wird die sichtbare Schicht durch die Höhe der Kamera z_{cam} und einen Offsetwert Δz.

mensionale Informationen zu berücksichtigen, wurde ein Sichtraum definiert (Abb. 5.5). Dieser berechnet sich aus einem horizontalen Offset Δz zur Kameraposition z_{cam}, so dass die Höhe des Raums in einen sichtbaren und unsichtbaren Bereich unterteilt wird. Damit gelten nur die NDT-Zellen als sichtbar, wenn für deren Höhenkomponente z zutrifft, dass $|z - z_{cam}| < \Delta z$ ist. Alle nun sichtbaren NDT-Zellen werden als Hindernis in die zweidimensionale Karte eingetragen. Ein Update der Karte findet nur noch statt, wenn eine neue NDT-Karte verfügbar ist. Die Kostenfunktion ergibt sich wie im dreidimensionalen Fall aus dem Einsatz von *ray tracing* (Abb. 5.4). Jedoch lassen sich jetzt effiziente Grid-Algorithmen wie der *Bresenham Line Algorithm* ([Bresenham, 1965]) einsetzen, um das Ergebnis schnell zu berechnen. In Gleichung (5.3) wird die nun vereinfachte Funktion gezeigt, bei der die zweidimensionale Karte $\mathbf{M}_A$ zum Einsatz kommt.

$$f(\mathbf{x}, \mathbf{M}_A) = \frac{1}{d_{max} n} \sum_i \min d_i(\mathbf{x}, \mathbf{M}_A), \text{mit} \quad \mathbf{M}_A : \mathbb{R} \times \mathbb{R} \to \mathbb{R} \tag{5.3}$$

Wie aus den Abbildungen 5.5 und 5.4 abzuleiten ist, handelt sich um eine grobe Approximation für die Sichtbarkeit. Sie setzt voraus, dass die Kamera annähernd eben positioniert und somit nicht auf der Querachse verdreht ist.

5.2.3 Kumulative Kartenkosten

Im Wesentlichen beschränken sich die folgenden Kriterien auf die Nutzung von kumulativen Kartenkosten. Zur Einschränkung des Kartenbereichs werden dabei zwei Masken verwendet. Die erste Maske hat die Form eines Kegels und wird durch drei Parameter bestimmt. Dabei handelt es sich um die minimale bzw. maximale Distanz und um den Öffnungswinkel (Abb. 5.6). Der Winkel und die Maximaldistanz sind dabei notwendige Parameter, um den Bereich einzuschränken. Diese Maske eignet sich bevorzugt für das Emulieren eines Sichtkegels in einer Karte. Damit kann eine Aussage über den (zukünftigen) Sichtbereich getroffen werden. Die zweite Maske hat die Form eines Kreises und wird durch seinen Radius beschrieben (Abb. 5.7). Sie eignet sich besonders gut für den Roboter mitsamt seines Umfanges. Damit wird etwas über die (zukünftige) Standfläche des Roboters ausgesagt. Beide Masken umschließen dabei ein lokales Kostengebirge dessen Werte aufsummiert werden. Das Ziel ist die Minimierung oder Maximierung des Gebirges. Die Funktion der Masken ist in der Gleichung (5.4) dargestellt. Dabei werden die Werte aller Elemente der Maske m an der Pose $\mathbf{x}$ in der Karte $\mathbf{M}$ akkumuliert und normalisiert. Der Wert m_{max} stellt dabei den maximalen Wert dar, den eine Zelle in $\mathbf{M}$ erreichen kann.

$$f(\mathbf{x}, \mathbf{M}) = \frac{1}{m_{max} n} \sum_{i=1}^{n} m_i(\mathbf{x}, \mathbf{M}), \text{mit} \quad \mathbf{M} : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R} \tag{5.4}$$

In den folgenden Unterkapiteln werden nun die verschiedenen Nutzungen der Masken präsentiert.

Aufenthaltswahrscheinlichkeit

Die Schätzung der Aufenthaltswahrscheinlichkeit wird derzeit im Rahmen einer anderen Arbeit am Fachgebiet erstellt. Nach Rücksprache wird angenommen, dass die Verteilung in Form einer gridbasierten Karte vorliegt (Kap. 2.2.1), ähnlich einer Hinderniskarte. Die wichtigste Aufgabe mit dieser Karte ist das Finden des nächsten Aufenthaltsortes. Für diesen Zweck wird die erste Variante der Maske genutzt (Abb. 5.6). Die Summe der Wahrscheinlichkeitswerte innerhalb der Maske soll maximiert werden. Denn das Maximum repräsentiert den häufigen Aufenthalt der Person an diesem Ort

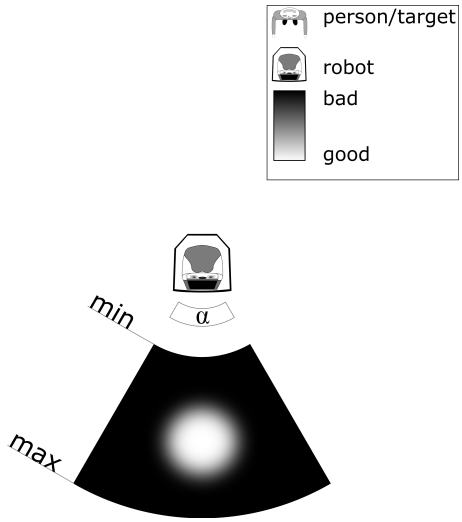


Abbildung 5.6: Kegelförmige Maske
Der Kreisausschnitt wird mittels minimaler Distanz, maximaler Distanz und dem Öffnungswinkel α bestimmt. Die davon eingegrenzte Fläche wird akkumuliert und normalisiert.

zu dieser Zeit. Aber auch die anderen Parameter des Kegels erfüllen eine Rolle. Die minimale Distanz sorgt dafür, dass der Roboter einen Abstand zum Aufenthaltsort einhält. Diese geschieht, in dem der Bereich zwischen der Position der Kamera und dem minimalen Abstand nicht gewertet wird. Damit wird die soziale Distanz berücksichtigt, die in diesem Fall dafür sorgt, dass der Roboter nicht zu nah am Aufenthaltsort parkt und sich der Nutzer dadurch bedroht fühlt. Im Falle einer fehlenden Minimaldistanz wird der Roboter versuchen, den ganzen Sichtbereich von seiner Position aus zu maximieren. Das hätte zur Folge, dass der Roboter z. B. direkt an der Seite eines Sofas zum Halten käme, welches er dann auf der ganzen Länge beobachten kann. Die maximale Distanz wird hier genutzt, um den Sensorabstand zu berücksichtigen. Ist die Person außerhalb des Sensorabstands, so kann man sie nicht beobachten. Der Öffnungswinkel kann letztendlich noch genutzt werden, um die Position des Roboters zu Person zu beeinflussen. Bei einem kleinerem Winkel fördert man die Ausrichtung hin zu einem lokalen Maximum in der Aufenthaltswahrscheinlichkeit. Der Winkel sollte jedoch gleich oder geringfügig kleiner als der tatsächliche Blickwinkel der Kamera sein. Damit wird der sichtbare Bereich der Aufenthaltswahrscheinlichkeit maximiert. Das der Winkel etwas kleiner sein sollte, hängt mit Personenerkennung zusammen. Ist der Nutzer zu nah am Bildrand, dann kann er ggf. nicht mehr erkannt werden. Neben Aufenthaltsorten können auch die Laufwege der Person in Form von einer dynamischen Aufenthaltswahrscheinlichkeit vorliegen. Diese kann mit einer kreisförmigen Maske ein-

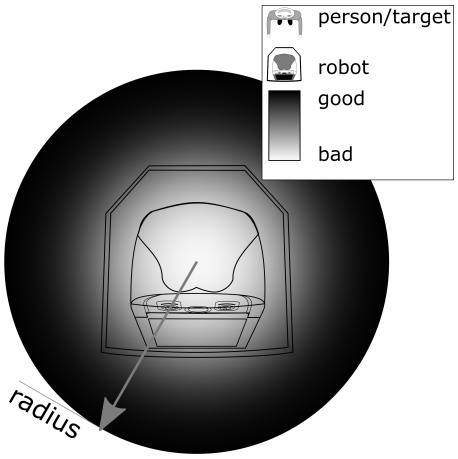


Abbildung 5.7: Kreisförmige Maske
Die Maske wird durch den Radius r bestimmt. Die Werte der eingegrenzten Fläche werden akkumuliert und normalisiert.

bezogen werden. In diesem Zusammenhang soll die Wahrscheinlichkeit am zukünftigen Standort minimiert werden. Damit wird vermieden, dass der Roboter an häufig genutzten Laufwegen zum Halten kommt. Da sich zum Ende der Arbeit herausstellte, dass keine dynamischen Aufenthaltswahrscheinlichkeiten in den Daten aufgezeichnet wurden, muss die Funktionalität auf das Feedback ausgelagert werden, was dessen Wert steigert.

Feedback

Für die Verwendung von Feedback wird die Kreismaske eingesetzt (Abb. 5.7). Dabei wird eine vorhandenen Feedback-Karte (Kap. 2.2.1) verwendet, in welcher ein geringer Wert für gutes Feedback steht. Deswegen wird hier das Kostengebirge minimiert. Ein Nutzen ist dabei die im vorherigen Unterkapitel angesprochene Freihaltung von häufig genutzten Wegen. Ein weiterer Nutzen ergibt sich aus der Vermeidung von Positionen, die den direkten Blick oder den Zugriff auf etwas versperren. Dies gilt insbesondere für Positionen zwischen Person und Fernseher. Für solche Situationen muss auf eine Kooperation mit dem Nutzer gesetzt werden. Dieser kann sofort schlechte Positionen bestrafen und den Roboter zur Neupositionierung zwingen.

Wandabstand

Beim Wandabstand wird auf die Kostenkarte (Kap. 2.2.1) zurückgegriffen, in der die Kreismaske eingesetzt wird. Mit der Erreichbarkeit (Kap. 5.2.1) als eigene Kostenfunktion wird bereits abgesichert, dass der Roboter eine Position erreichen kann, ohne mit einem Hindernis zu kollidieren. Bei der Fahrt müssen wir jedoch beachten, dass der Roboter am Ziel noch rangieren muss. Konkret geht es dabei um die Ausrichtung, für die sich der Roboter dreht. Bei Integrationstests wurde festgestellt, dass der Roboter zu nah an der Wand stand. Auch wenn augenscheinlich genügend Platz für eine Drehung war, konnte der Pilot den Vorgang nicht durchführen. Der Roboter blieb stehen. In manchen Fällen hat er jedoch ohne Probleme eine Drehung an der Position geschafft. Um dem Verhalten jedoch die nötige Stabilität zu verleihen, wurde diese Kostenfunktion geschaffen. Die Kreismetrik stellt dabei eine Dilatation des Roboters da und beschreibt somit dessen Standfläche. Das Ziel ist nun, die Erreichbarkeit dieser Fläche zu erhöhen, was einer Minimierung des Belegtheitswertes im Kreis zur Folge hat. Dadurch wird eine Position, die zu nah an Hindernissen ist, schlechter bewertet.

Kapitel 6

Beobachtungsverhalten

In diesem Kapitel kommen wir nun zum Beobachtungsverhalten (*AttendanceBehavior*). Wie in Abbildung 6.1 zu sehen, bedeutet dies die Nutzung verschiedener Verhalten und Fähigkeiten. Dabei werden im folgenden drei realisierte Aspekte beschreiben, bevor dann der Ablauf im Detail dargelegt wird. Bei den drei Aspekten handelt es sich um die Aktivierung, das Beobachten und die Suche nach der Person. Die Aktivierung des Verhaltens geschieht durch den Nutzer, der dem Roboter seinen Wunsch nach Ruhe mitteilt. Das könnte z. B. durch eine Eingabe auf dem Touch-Monitor erfolgen wie in Abbildung 6.2a dargestellt. Der Roboter fährt dann zu einer geeigneten Position und beobachtet dabei den Nutzer. Dabei bleibt er in Sichtweite der Person (Abb. 6.2b). Wurde die Person für einen längeren Zeitraum nicht gesehen, so wird die Personensuche aktiviert. Hat der Roboter die vermutete Position einer Person erreicht, so fordert er sie zu einer Bestätigung auf, z. B. durch eine Fernbedienung (Abb. 6.2c).

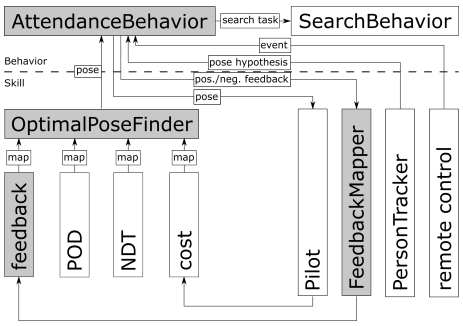


Abbildung 6.1: Informationsfluss des Beobachtungsverhaltens
Hier wird das Verhalten dargestellt, welches den gesamte Prozess mit der Positionssuche steuert. Dabei stellt es u. a. die Verbindung zu einem Feedback-Modul her.

(a) Aktivierung

(b) Beobachtung

(c) Bestätigung mit Fernbedienung

Abbildung 6.2: Drei Aspekte des Beobachtungsverhaltens

In (a) wird die Aktivierung des Beobachtungsverhaltens gezeigt, dass durch eine Eingabe am Touch-Bildschirm geschieht. Danach wartet der Roboter in (b) an einer geeigneten Position und beobachtet. Wurde eine Personensuche ausgelöst, fährt der Roboter mögliche Personen an und fragt nach einer Bestätigung mit einer Fernbedienung (c).

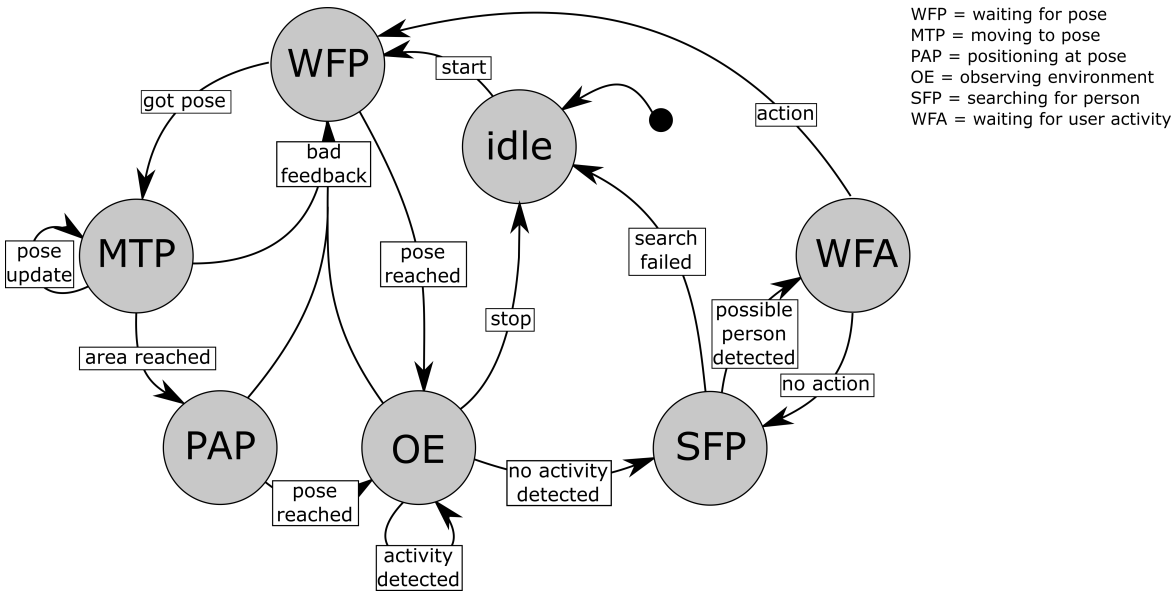


Abbildung 6.3: Zustandsmaschine

Im Bild ist der Ablauf des Beobachtungsverhalten dargestellt. Dabei gibt es einen Hauptablauf von idle über OE wieder zu idle und einen Nebenablauf (SFP, WFA) zur Personensuche.

6.1 Zustandsmaschine

Beim Beobachtungsverhalten wird intern eine Zustandsmaschine genutzt (Abb. 6.3). Sie regelt den Ablauf und die Kommunikation mit allen Modulen, die genutzt werden. Im Ruhezustand (*idle*) wartet das Verhalten. Es startet, wenn es eine Aktivierung bekommt und wechselt dann in den nächsten Zustand. Im WPF Zustand aktiviert es die optimale Positionssuche und wartet dann darauf, dass es eine Position von ihr erhält. Ist diese nun verfügbar ist, so wird geprüft, ob der Roboter schon am Ziel angekommen ist. Steht er bereits an richtigen Position, so wird gleich zur Beobachtung (OE) übergegangen. Falls nicht, so startet das Verhalten den Navigationspilot mit der Position und wechselt in den nächsten Zustand. Im MTP Zustand wird ständig die Entfernung zur Zielpose berechnet, während er sich der Pose annähert. Währenddessen findet weiterhin die Positionssuche statt, um eine noch bessere Position zu finden. Diese wird immer dem Piloten gemeldet, woraufhin er sie als Zielposition setzt. Wenn der Roboter sich in einem bestimmten Radius um die Position befindet, so schaltet er

in die Phase zwei der Navigation (PAP) um. In diesem Zustand (PAP) wird nun die Positionssuche gestoppt. Der Roboter befindet sich nah genug an der ausgewählten Position, so dass wir nur noch die Ausrichtung korrigieren wollen. Der Roboter ist angekommen und beobachtet nun seinen Sichtbereich (OE). Dabei schaut er, ob er die Posenhypothese der Person sehen kann. Dafür wird ein Sichtbereich durch einen Kameraöffnungswinkel und eine maximale Distanz bestimmt. Die Hypothese wird dabei durch einen vorhandenen Personen-Tracker zur Verfügung gestellt, der die Person im Kamerabild erkennt und dessen Position schätzt. Ist die Hypothese nicht mehr zu sehen, startet ein Timer. Läuft dieser ab, so gilt die Person als vermisst. Daraufhin wird das Beobachtungsverhalten eine Personensuche mit Hilfe eines Suchverhaltens (*SearchBehavior*) einleiten. Im SFP Zustand tritt das Beobachtungsverhalten in den Hintergrund und wartet auf die Rückmeldung der Personensuche. Wird keine Person erkannt und wurden alle Suchmöglichkeiten ausgeschöpft, so wechselt das Programm wieder in den Ruhezustand. Erkennt das Suchverhalten eine potentielle Person, so wird das Beobachtungsverhalten benachrichtigt und es tritt wieder in Aktion. In diesem Zustand (WFA) wartet der Roboter nun auf die Aktion einer potentiellen Person. Diese muss ihre Anwesenheit z. B. mit einer Fernbedienung (*remote control*) bestätigen. Passiert dies nicht, so wird weiter gesucht. Bestätigt die Person die Anwesenheit, so beginnt die Suche nach einer geeigneten Beobachtungspostion erneut.

6.2 Feedback

Auf der Fahrt zur Position und während der Beobachtungsphase kann der Nutzer dem Roboter, z.B. mit einer Fernbedienung (*remote control*), Feedback geben. Dabei kann sowohl positives wie auch negatives gegeben werden. Bei negativen Feedback wird die Positionssuche neu gestartet.

Kapitel 7

Experimentelle Untersuchung

Diese Kapitel beschäftigt sich mit dem Thema Testen. Im Folgenden wird zuerst die allgemeine Testumgebung vorgestellt. Danach werden Laufzeittests der Kostenfunktionen präsentiert. Darauf folgt ein Integrationstest, der die Ausführung auf dem Roboter zeigt. Zum Abschluss wird ein kurzer Akzeptanztest präsentiert. Aufgrund von Zeitmangel wurde dieser nur mit einer kleinen Testgruppe ausgeführt, so dass die Ergebnisse nicht repräsentativ sind. Für die Tests wurde als Optimierer die PSO eingesetzt.

7.1 Testumgebung

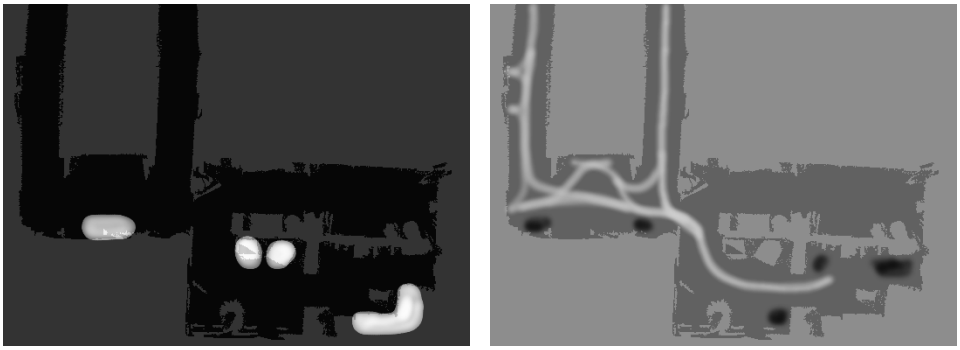
Für die Funktions- und Akzeptanztests diente das *Living Lab* des Fachgebiets NIKR, welches in Abbildung 7.1 gezeigt wird. Es unterteilt sich in zwei Wohnbereiche und einen im Flur zugänglichen Küchenbereich. Um sich die Umgebung besser vorstellen zu können, werden in Anhang A.1 einige Ausschnitte gezeigt. *Living 1* besteht aus zwei Sesseln, einem Fernseher und einigen Schränken (Anh. A.1d). Große Freiflächen ermöglichen hier eine einfache Navigation. *Living 2* ist mit einem Fernseher, einer Schrankwand, einem Sofa und einem Wohnzimmertisch ausgestattet (Anh. A.1b). Der zentral verlegte Teppich erschwert die Fahrt des Roboters. Zusätzlich befindet sich hier auch eine Ladestation für den Roboter. Im Flur befindet sich eine verschließbare Teeküche, ein kleiner Tisch und zwei Sitzgelegenheiten (Abb. A.1a). Bei den Tests werden vor allem die Bereiche *Living 1* und *Living 2* Verwendung finden.

Abbildung 7.1: Schematische Ansicht des Labors
Abgebildet ist die Draufsicht auf das Living Lab mit Zugang zum Flur und der Küche. Weiterhin sind die Positionen eingezeichnet, an denen die Fotos von Anhang A.1 geschossen wurden.

Neben der Karte für die Navigation spielen zwei weitere Karten eine besondere Rolle: Die Karte der Aufenthaltswahrscheinlichkeit (Abb. 7.2a) und die Feedback-Karte (Abb. 7.2b). Diese Karten üben den entscheidenden Einfluss bei der Positionsfindung aus. Im Laufe der Funktionstest wird an passenden Stellen immer wieder auf sie verwiesen, um Zusammenhänge darzustellen.

7.2 Laufzeit der Kostenfunktionen

Im folgenden Unterkapitel werden die Funktionen auf ihre Laufzeit analysiert. In der Tabelle 7.1 wurde die Positionsbewertung für 4453 Positionen durchgeführt, von denen 917 erreichbar waren. Dabei fand der Test im Simulator statt und es wurde zur Positionsoptimierung die PSO eingesetzt. Dabei erfolgt die Betrachtung von zwei besonderen Aspekten. Zuerst wird die Sichtbarkeit im zwei- und dreidimensionalen Raum unter-



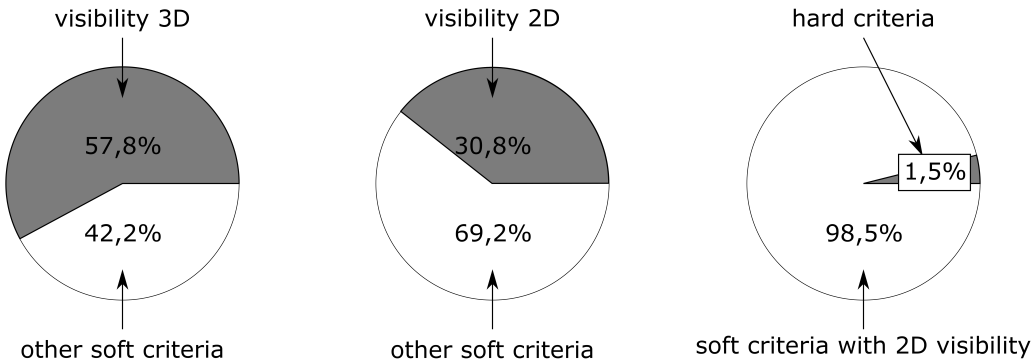
(a) Aufenthaltswahrscheinlichkeit der Person

(b) Statische Feedback-Karte

Abbildung 7.2: Entscheidende Karten für das neue Verhalten
Auf der linken Seite ist die Karte der Aufenthaltswahrscheinlichkeit zu sehen. Ein hoher Wert (weiß) signalisiert eine Stelle, an der eine Person häufig angetroffen wird. Auf der rechten Seite handelt es sich um eine Feedback-Karte. Diese vermitteln den Wunsch des Nutzers bezüglich einer guten Standposition. Hier werden die schwarzen Flächen bevorzugt.

	Kostenfunktion	Aufrufe	Gesamtzeit (ms)
Harte Kritieren	Erreichbarkeit	4453	23,8
Weiche Kriterien	Aufenthaltswahrscheinlichkeit	917	92,5
	Feedback	917	46,25
	Wanddistanz	917	46,25
	Nächstgelegene Pose	917	0,6
	Sichtbarkeit 3D	917	251,4
	Sichtbarkeit 2D	917	82

Tabelle 7.1: Laufzeit der einzelnen Metriken
Für die Messwerte wurde die Positionsoptimierung direkt gestartet. Es wurde kontinuierlich die Optimierung ausgeführt. Dabei wurden 4453 Positionen getestet, von denen 917 erreichbar waren. Es wird deutlich, dass die Sichtbarkeit als Metrik eine hohe Rechenzeit benötigt.



(a) 3D Sichtbarkeit **(b)** 2D Sichtbarkeit **(c)** Gesamtzeit

Abbildung 7.3: Rechenzeit als Kreisdiagramme

Es werden drei Kreisdiagramme gezeigt die aus den Daten der Tabelle 7.1 erstellt wurden. In (a) wird die dreidimensionale Sichtbarkeit mit den anderen weichen Metriken, ohne die 2D Sichtbarkeit, verglichen. In (b) wird das Gegenstück zum ersten Diagramm gezeigt, welches den Vergleich zur zweidimensionalen Sichtbarkeit darstellt. Im letzten Diagramm wird die Rechenzeit von harten und weichen Kriterien bei gleicher Aufrufe gegenüber gestellt.

sucht. Danach folgt die Betrachtung der Rechenzeit von harten und weichen Kriterien.

Im Rahmen der Arbeit wurde die Sichtbarkeit neu definiert. Dabei sollte dies im dreidimensionalen Raum, unter Nutzung einer NDT-Karte, geschehen. Als Alternative wurde eine Sichtbarkeit in 2D erstellt, die mit einer Abstraktion aus der NDT-Karte arbeitet. Die beiden Sichtbarkeitsfunktionen wurden in Abbildung 7.3a und 7.3b mit den anderen weichen Kriterien gegenübergestellt. Wie zu sehen ist, nimmt die dreidimensionale Sichtbarkeit über 50 Prozent der Rechenzeit für die weichen Kriterien ein. Im Gegensatz dazu kann der Einsatz der zweidimensionalen Kostenfunktion den Aufwand für die Berechnung der Sichtbarkeit auf ein Drittel reduzieren. Dadurch wird nur ein Drittel der Rechenzeit für die weichen Kostenfunktionen benötigt. Die Ergebnisse bezüglich des hohen Rechenaufwands aus [Kessler et al., 2012] konnten damit erneut bestätigt werden. In seiner Folgearbeit [Kessler et al., 2013] präsentierte er eine alternative Höhenkarte als pseudo-zweidimensionale Karte.

In Abbildung 7.3c werden die Rechenzeiten der harten und weichen Kriterien gegenübergestellt. Dabei wird deutlich, dass die harten Kriterien bei gleicher Anzahl an

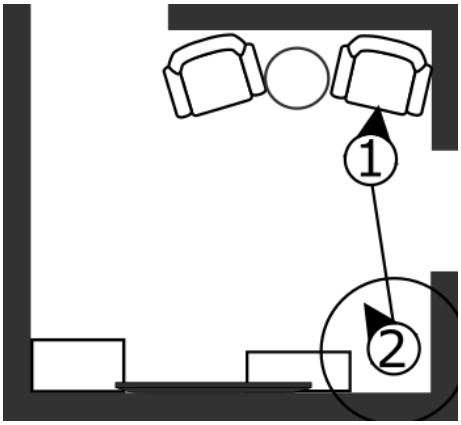


Abbildung 7.4: Eine normale Anfahrt zu einer Position
Zu sehen ist der Living 1 *Teil des Labors. An der Position 1 steht der Roboter zu beginn. Der Positionsfinder findet Position 2 und fährt diese an. Bei überschreiten des Kreises um Position 2 wechselt das Verfahren in die zweite Navigationsphase und stoppt den Positionsfinder. Schließlich richtet sich der Roboter in dieser Phase noch aus.*

Aufrufen weniger als zwei Prozent der Rechenzeit benötigen. Wird eine Position durch die harten Kriterien abgelehnt, so können die rechenintensiven weichen Kriterien übersprungen werden. Die Unterteilung in diese beiden Gruppe ist daher gerechtfertigt.

7.3 Integrationstest

Für die Übergabe der Applikation an das Fachgebiet muss dieses grundlegend getestet werden. Dabei werden im Folgenden verschiedene Tests durchgeführt.

Finden einer Position: Der erste Test handelt von der Funktionsfähigkeit des optimalen Positionfinders. Dieser muss anhand seiner Kriterien eine optimale Position finden. Der Partikelschwarm der PSO startet an der aktuellen Roboterposition und verteilt seine Partikel normalverteilt. Danach wird in mehreren Runden optimiert. Wenn der Schwarm konvergiert ist, wird die optimale Position ausgegeben (Anh. A.2). Vergleicht man dieses Ergebnis nun mit den Karten aus Abbildung 7.2, so sieht man die Zusammenhänge. Die Position befindet sich innerhalb des positiven Bereichs der Feedback-Karte und die Ausrichtung zeigt mittig in die Richtung der hohen Aufenthaltswahrscheinlichkeit bei den Sitzgelegenheiten (Siehe Abb. 7.4). Damit ist die Funktion der Positionsfindung gezeigt.

Anfahren einer Position: Auch während der Anfahrt zu einer Position wird stets von Neuem optimiert. Neben Hindernissen auf dem Weg berücksichtigt es so auch das Entdecken neuer optimaler Positionen, da vormals versperrte Bereiche jetzt wieder zugänglich sind. Dabei unterscheiden wir hier die zwei trivialen Fälle: Es tritt kein Hindernis auf (a) und es tritt eines auf (b).

(a) Eine normale Fahrt: In diesem Fall ist die Startphase die kritische Phase. Der Roboter kann zu diesem Zeitpunkt gleich weit entfernt von mehreren gleich guten Positionen stehen. Im praktischen Einsatz wird der Roboter irgendwann die erste Bewegung machen und sich somit einer Position näheren. Als Konsequenz verlieren die anderen guten Positionen an Wert. In diesem Fall sind die zwei Phasen der Navigation gut zu beobachten (Vgl. Anh. A.3). In der erste Phase wird ständig weiter optimiert und in der zweiten Phase wird nur noch positioniert. In Abbildung 7.4 ist um die Zielposition (Position 2) ein Kreis eingezeichnet, welcher den Übergang der Phasen darstellt. Hat er die Linie erreicht, so wird der Positionsoptimierer gestoppt und der Roboter richtet sich an der Stelle aus.

(b) Der Pfad wird plötzlich blockiert: Der Roboter fährt zur momentan optimalen Pose und ein Hindernis blockiert plötzlich den Weg. Es wird erwartet, dass der Positionsoptimierung eine neue Pose ausgibt. Dieser Vorgang wird in Abbildung 7.5 illustriert. Der Roboter startet an Position 1, aus der er das Hindernis (in grau) nicht sehen kann. Wie im ersten Fall wird die optimale Position 2 gewählt. Der Roboter fährt diese also an, wobei er sich dreht und mit seinem Laser-Scanner die entsprechende Ecke nun einsehen kann. Er erkennt nun, dass die Anfahrt nicht möglich ist. Das spiegelt sich in seiner Kostenkarte wieder. Auf Basis ebendieser sucht die Optimierung nun die nächstbeste Position (4) zum Warten und fährt diese an (Anh. A.4). Ist die Aufenthaltskarte dynamisch und erkennt der Roboter, dass die Person sich nur bei den Sesseln aufhalten wird, so wählt er die nächste beste Position (4) (Abb. 7.6).

Beobachten einer Person: Der Roboter beobachtete eine Posenhypothese, die von einem Personen-Tracker geliefert wird. Wenn diese Hypothese innerhalb des Sichtkegels

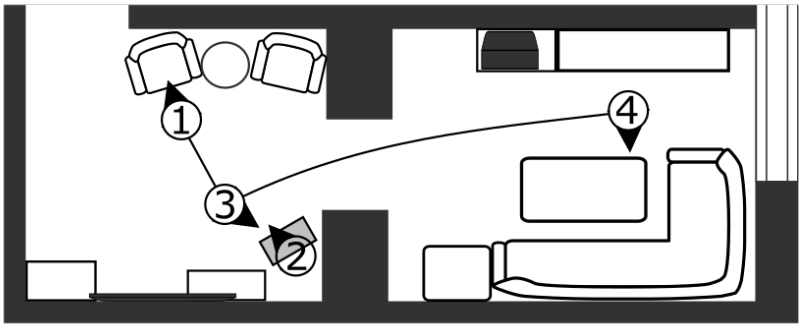
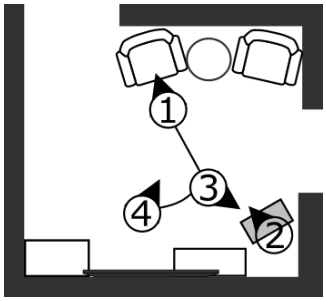


Abbildung 7.5: Eine Anfahrt mit Hindernis und statischer Aufenthaltskarte
An der Position 1 steht der Roboter zu beginn. Der Positionsoptimierer findet Position 2 und fährt diese an. Bei Position 3 erkennt er, dass sich ein Hindernis zwischen ihm und der Pose befindet. Daraufhin findet er Position 4 und fährt diese an.

Abbildung 7.6: Eine Anfahrt mit Hindernis
Das Geschehen ist wie in Abbildung 7.5, wobei hier bekannt ist, dass sich die Person nur bei den Sesseln aufhält. Deswegen muss in deren Nähe die nächst beste Position gewählt werden.



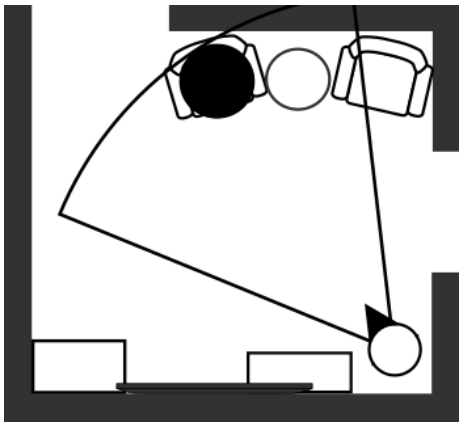


Abbildung 7.7: Beobachtung der Hypothese *Zu sehen ist der* Living 1 *Teil des Labors. Der Roboter beobachtet die Hypothese des Nutzers. Wenn sich die Hypothese innerhalb des Kreisausschnittes befindet, so wird dies als Aktivitätsbeweis geführt.*

liegt, so wird dies als erkannte Aktivität der Person gewertet (Abb. 7.7). Wenn für die Dauer eines gegebene Zeitintervalle keine Aktivität erkannt wird, so nutzt das Beobachtungsverhalten eine Personensuche. Wird eine mögliche Person gefunden, so muss diese ihre Präsenz mittels einer Fernbedienung bestätigen.

7.4 Akzeptanztests

Aufgrund von Zeitmangel konnte kein ausgedehnter Akzeptanztest durchgeführt werden. Ziel wäre die Untersuchung der Akzeptanz bezüglich des Beobachtungsverhaltens und der Möglichkeit, dass es durch Feedback beeinflusst werden kann.

Um einen nicht repräsentativen Ausblick dazu zu geben, wurde eine Umfrage mit drei Mitarbeitern des Fachgebietes durchgeführt. Sie sind den Umgang mit dem Roboter gewohnt und können demnach eine kritische Einschätzung bezüglich der Umsetzung geben. Es wurden drei Ausgangslagen untersucht in der bereits vorgestellten Testumgebung (Abb. 7.1). Dabei saß die Person einmal auf dem Sessel in *Living 1* (Anh. A.5), auf dem Sofa in *Living2* (Anh. A.6) und einmal stand sie in *Kitchen* (Anh. A.7). Die Position der Person war zugleich die einzige Aufenthaltswahrscheinlichkeit, die der Roboter kannte. In der Feedback-Karte waren die Laufwege gegeben (wie Abb. 7.2b ohne positives Feedback).

Das Geben von Feedback wurde gemischt aufgenommen. Teilweise wurde bemängelt, dass man zu oft Feedback geben muss, bis ein gewünschtes Verhalten auftritt. Auch

wurde eine akustische Rückmeldung bei der Gabe von Feedback vermisst. Alle Drei stimmten jedoch überwiegend zu, dass es Spaß macht, die Veränderungen im Verhalten zu beobachten, wenn man Feedback gegeben hatte. Die Anfahrt wurde überwiegend als nicht flüssig bemängelt. Dabei traf dies insbesondere bei der zweiten Phase der Anfahrt zu, weil der Roboter lange brauchte, um sich auszurichten oder das Ziel nicht erreichte. Auch wurde bemängelt, dass der Roboter im Küchenbereich mit Hindernissen kollidierte. Das ist im Nachhinein durch eine fehlerhafte No-Go-Karte zu begründen. Es lässt sich nicht ableiten, ob die Positionswahl schlecht war.

Kapitel 8

Zusammenfassung und Ausblick

In diesem Kapitel wird die Arbeit kurz zusammengefasst und danach wird ein Ausblick gegeben.

8.1 Zusammenfassung

In dieser Arbeit wurde eine optimale Positionssuche und ein Beobachtungsverhalten erarbeitet. Die Positionssuche (*OptimalPoseFinder*) arbeitet dabei erstmalig ohne eine Posenhypothese der Person. Die Berücksichtigung dieser wird allein durch eine Aufenthaltswahrscheinlichkeit realisiert. In diesem Rahmen wurden bekannte Kostenfunktionen auf diese Wahrscheinlichkeit abgebildet. Der Verzicht auf die Posenhypothese ermöglicht dem Optimierer Positionen in anderen Räumen zu wählen, die die Person erst in naher Zukunft betreten wird. Ebenfalls zum ersten Mal wurde Feedback bei der Optimierung miteinbezogen, so dass der Nutzer die Position beeinflussen kann. Dadurch findet eine bessere Anpassung des Roboters an den Nutzer statt. Bei der Sichtbarkeit wurde ein neuer Ansatz entwickelt, der nicht mehr die Sichtbarkeit der Person, sondern den sichtbaren Freiraum vor dem Roboter beschreibt. Dabei hat sich der Ansatz im zweidimensionalen Raum durchgesetzt, da die Variante in 3D einen wesentlichen Mehraufwand in der Laufzeit bedeutete. Welcher der beiden Ansätze die bessere Qualität erzeugt, konnte noch nicht getestet werden. Die Optimierung wurde generisch implementiert, so dass Positionen und Posen mit unterschiedlicher Anzahl

an Dimensionen optimiert werden können. Dabei wurde der Ansatz einer universellen Zielfunktion aus den Arbeiten von Kessler ([Kessler et al., 2012]) weiterverwendet. Für die Nutzung der Positionssuche wurde ein Beobachtungsverhalten (*AttendanceBehavior*) entwickelt. Dieses nutzt die optimierte Position, um den Nutzer von dort zu beobachten. Um die Position zu erreichen wurde eine zweiphasige Anfahrt realisiert. Die Beobachtung der Person besteht daran, dass die vorhandene Posenhypothese im Sichtbereich liegen muss. Ist dies nicht der Fall, so wird nach einem Timeout eine ebenfalls vorhandene Personensuche aktiviert. Bei einer erfolgreichen Suche startet der Roboter das Beobachtungsverhalten erneut. Bei Misserfolg wird noch kein spezielles Verhalten aktiviert. Im Beobachtungsverhalten ist auch die Rückmeldung durch Feedback integriert wurden. Wenn der Roboter die Position anfährt oder von dort beobachtet, kann der Nutzer dem Roboter über eine Fernbedienung Feedback geben. Ist dieses negativ, so leitet das Verhalten sofort eine neue Positionssuche ein.

8.2 Ausblick

In dieser Arbeiten wurden vor allem Grundlagen geschaffen. Es kamen vorwiegend kumulative Kostenfunktionen zum Einsatz. Die Zellen in den eingeschlossenen Gebirgen werden mit den gleichen Gewicht verwertet. Dabei ist vorstellbar, das auch andere Gewichtungen denkbar sind, z. B. normalverteilt. Ähnliche Techniken kamen schon bei [Kessler et al., 2012] zum Einsatz. Bei der Sichtbarkeit wurden die beiden Ansätze bis jetzt nur nach ihrer Laufzeit bewertet. Ein Überprüfung bezüglich der Qualität des Ergebnisses steht noch aus. Die Aktivitätserkennung wird nur durch die Anwesenheit der Posenhypothese realisiert. Jedoch ist auch eine bewusstlose Person anwesend und der Roboter sollte auf eine solche Situation reagieren können. An dieser Stelle ist ein Spielraum für weitere Verfahren. Es wird nur eine rudimentäres Personensuchverhalten genutzt, wenn die Person abwesend ist. Bei Erreichen einer möglichen Person fordert es eine Eingabe durch den Nutzer. Wenn diese nicht mehr dazu in der Lage ist, sucht der Roboter weiter und wird irgendwann die Suche erfolglos abbrechen. Gleiches würde passieren, wenn überhaupt keine Person anwesend ist. Wenn der Nutzer nur ein Bad nimmt, ist es nicht weiter tragisch, wenn der Roboter ihn nicht findet und

in einen Ruhemodus wechselt. Hat die Person z. B. unerlaubt die Wohnung verlassen und bräuchte dafür eine Begleitung in Form von Pflegepersonal, so muss dieser Fall entsprechend behandelt werden. Schlussendlich müssen auch noch ausgedehnte Akzeptanztests durchgeführt werden. Aufgrund von Zeitmangel konnten dies nicht innerhalb dieser Masterarbeit geschehen. Dabei ist die Akzeptanz des Beobachtungsverhaltens im Allgemeinen und die Akzeptanz von Feedback im Speziellen zu untersuchen.

Anhang A

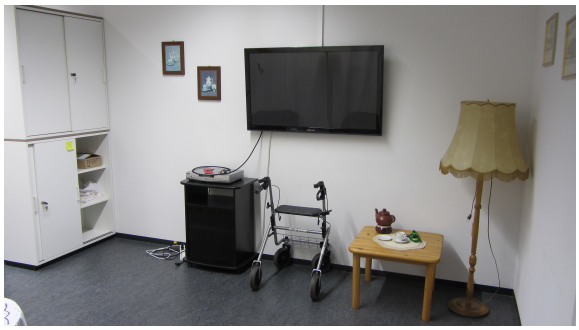
Weitere Abbildungen

(a) Teeküche mit Sitzgelegenheit

(b) Schrankwand in *living 2*

(c) Sitzgarniatur in *living 2*

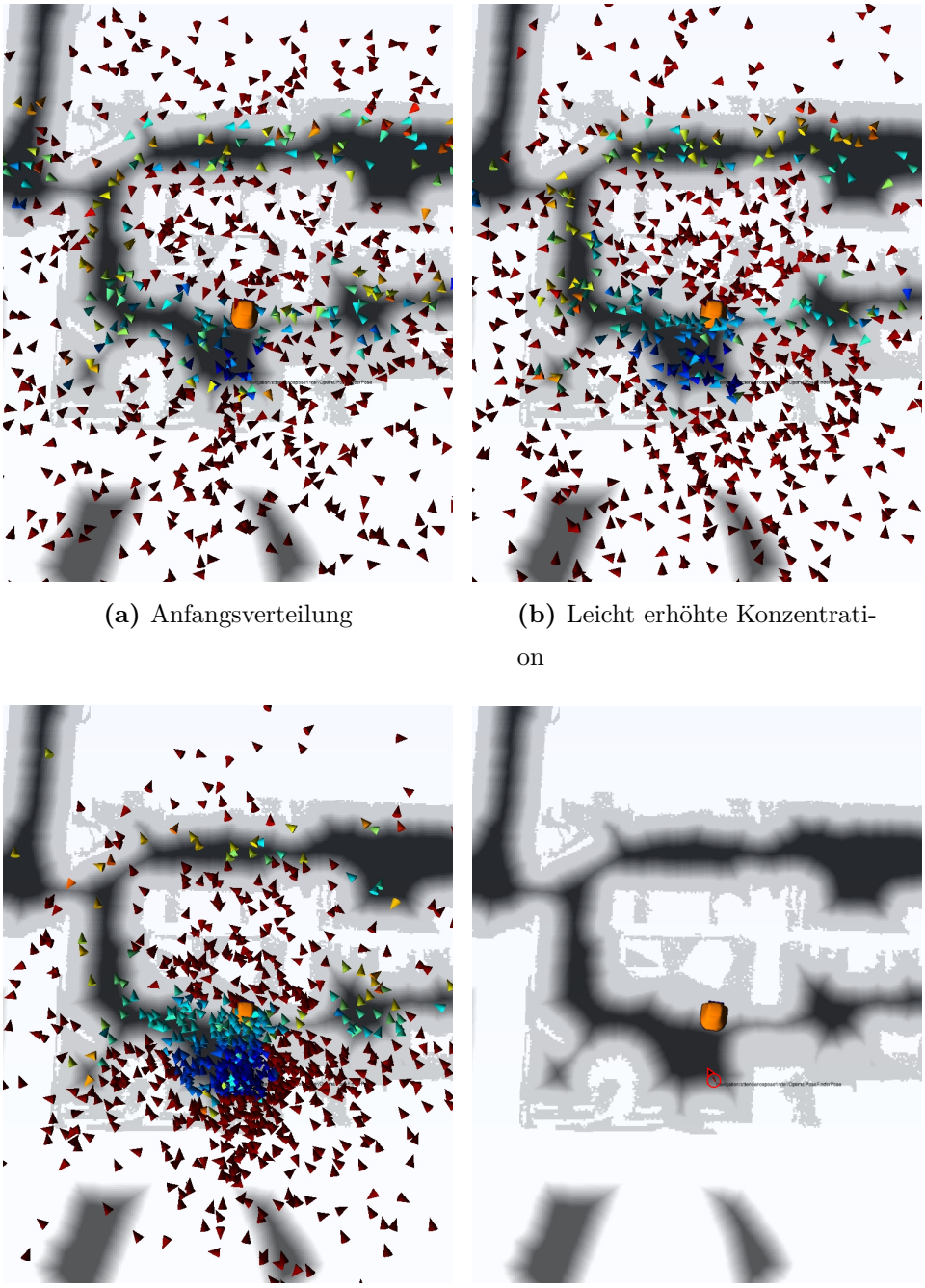
(d) Sessel in *living 1*



(e) Fernseher in *living 1*

Abbildung A.1: Fotos aus dem *Living Lab*

Um die Gegebenheiten darzustellen werden verschiedene Ausschnitte aus dem Labor gezeigt.



(a) Anfangsverteilung

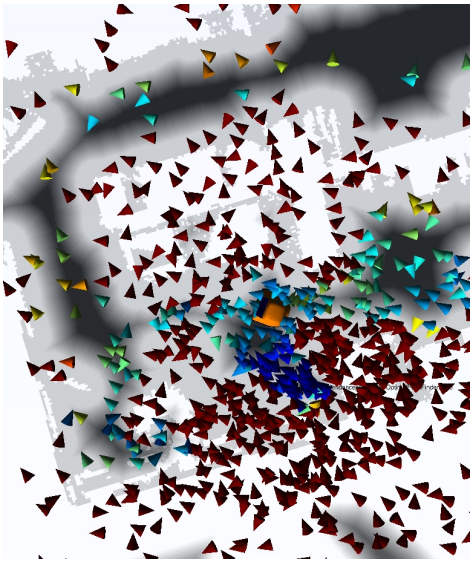
(b) Leicht erhöhte Konzentration

(c) Hohe Konzentration

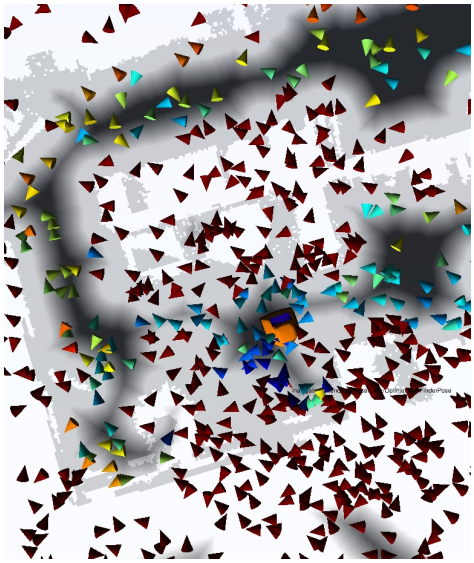
(d) Optimale Pose

Abbildung A.2: Ablauf einer Positionsfindung

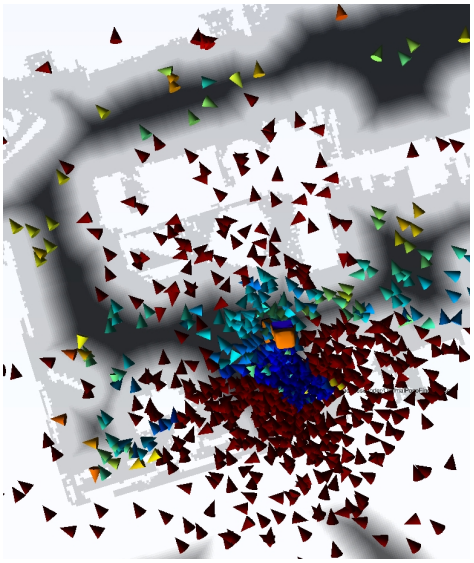
Zu sehen sind Auszüge aus dem Ablauf des Partikelfilters. Blaue Kegel symbolisieren gute Werte die über das Farbspektrum bis zum schlechtesten Wert (rot) vordringen. Der Roboter findet eine gute Position in living 1*, was mit dem entsprechenden Feedback-Wert zusammenhängt.*



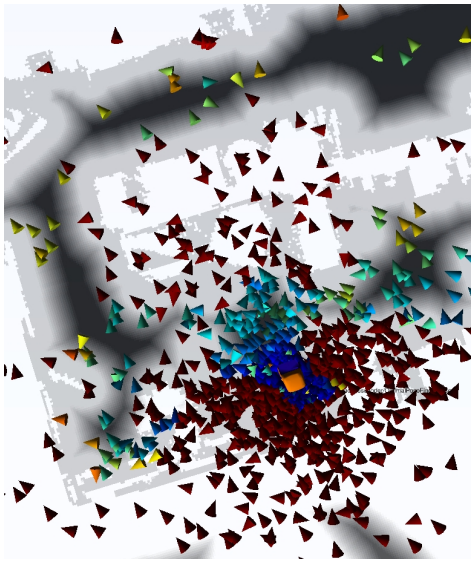
(a) PoseFinder konvergiert



(b) PoseFinder neu initialisiert



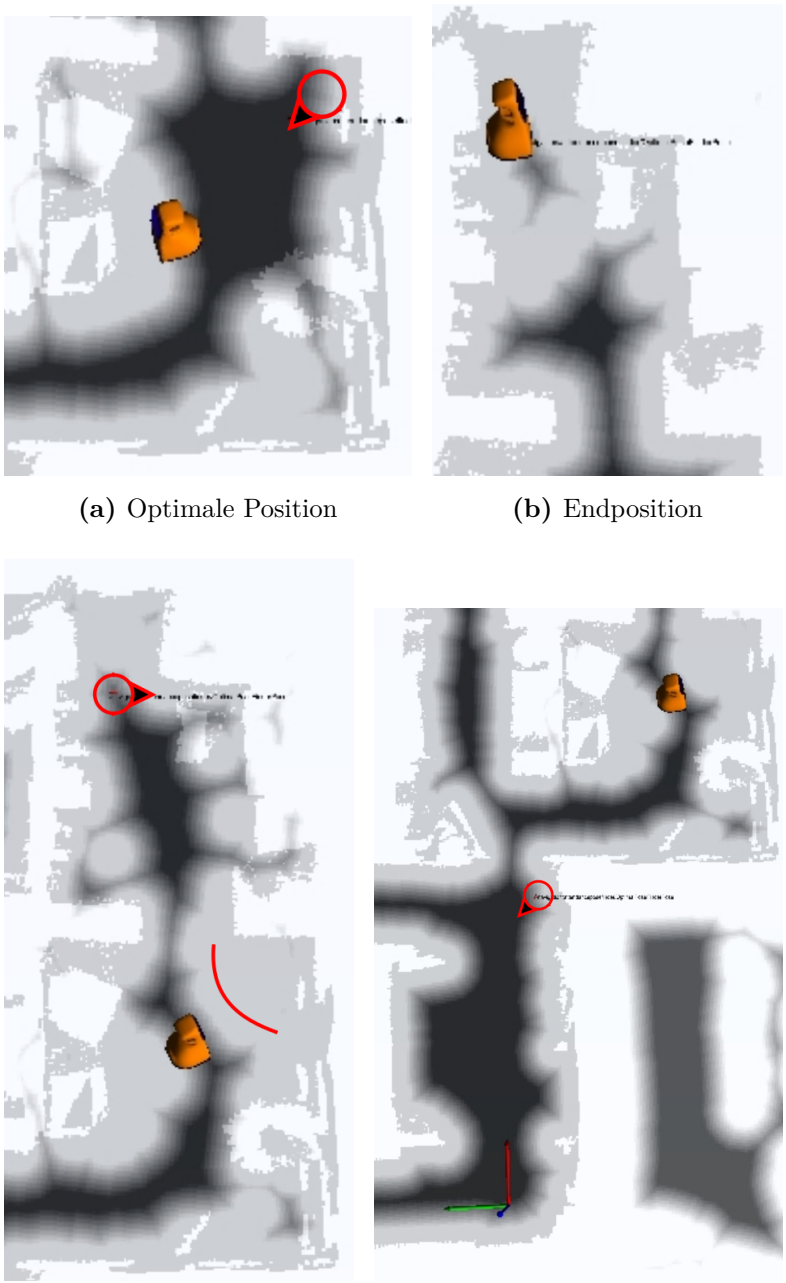
(c) PoseFinder gestoppt



(d) Positionierung des Roboters

Abbildung A.3: Optimierung während der Fahrt

Zu sehen sind Auszüge aus dem Ablauf der Partikel Schwarm Optimierung. Blaue Kegel symbolisieren gute Werte die über das Farbspektrum bis zum schlechtesten Wert (rot) vordringen. Während der ersten Phase der Anfahrt zur optimalen Position wird kontinuierlich neu optimiert (a und b). Wenn der Roboter innerhalb eines gewissen Radius ist, wird sich nur noch mit Hilfe der letzten Position ausgerichtet (c und d).



(a) Optimale Position **(b)** Endposition

(c) Hindernis erkannt **(d)** Eine Alternative

Abbildung A.4: Optimierung während der Fahrt mit Hindernis

Der Verlauf ist wie in A.3. Hier zeigen wir nur die gewählten Postionen. Am Anfang wird eine sehr ähnliche Position gewählt und angefahren (a). Nach einem Drehvorgang (c) sieht der Roboter, dass die Position blockiert ist. Im aktuellen Raum kann er nur noch auf Positionen stehen, die als Laufwege ausgewiesen sind. Daher wählt er entferntere Positionen (c und d). Während mehrerer Durchläufe der PSO setzt sich Position in living 2 *durch (b).*

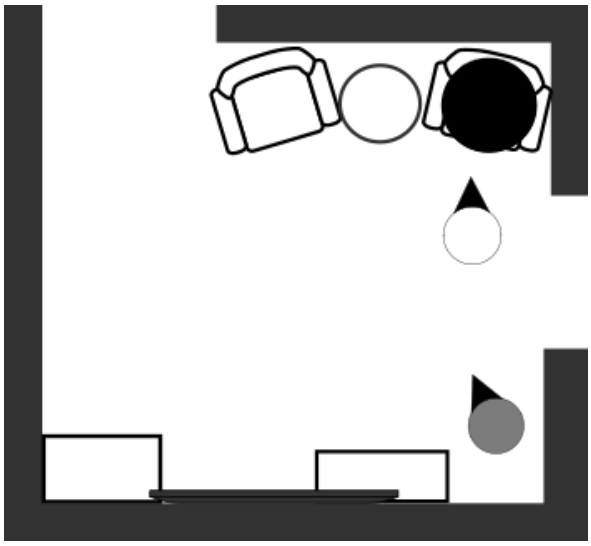


Abbildung A.5: Test Szenario in living 1

Der Ausgangspunkt war eine sitzende Position im Sessel (schwarzer Punkt). Dabei fungierte diese Position gleichzeitig als Aufentahltswahrscheinlichkeit. Von dort sollte der Roboter eine geeignete Position finden. Er platzierte sich vorwiegend unten rechts in der Ecke (grau). Jedoch ist rechts von ihm ein Schrank mit Schiebetüren. Die Nutzer wünschte sich deshalb, dass der Roboter weiter nach links rückt, da er den Schrank dort nicht blockieren würde

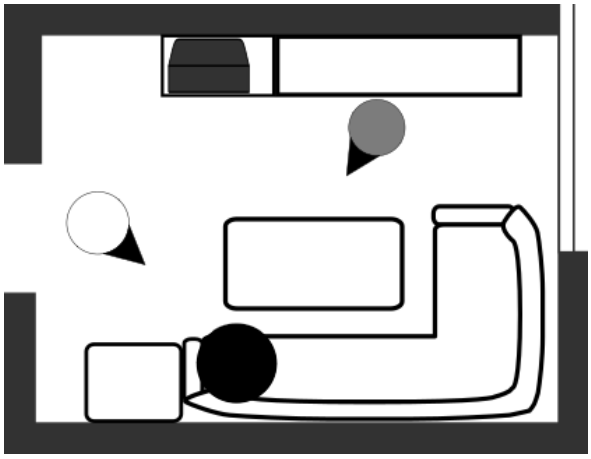


Abbildung A.6: Test Szenario in living 2

Der Ausgangspunkt war eine sitzende Position auf dem Sofa. Der Roboter stellte sich vor eine Schrankwand (grau). Auf diese konnte so nicht zugegriffen werden. Die Nutzer wünschten sich, dass der Roboter sich weiter links, neben dem Fernseher aufstellten.

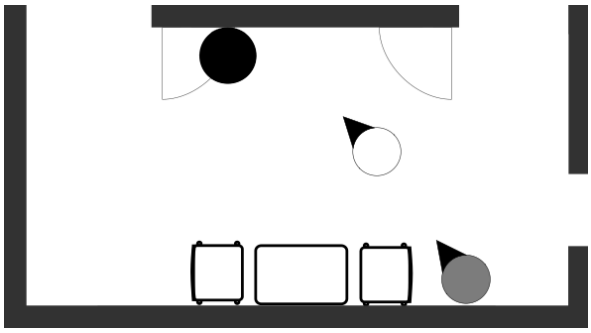


Abbildung A.7: Test Szenario in kitchen

Der letzte Test fand in der Küche statt. Hier platzierte sich der Roboter nah am Fenster. Jedoch kam es dazu, dass er die Hindernisse (Stuhl und Tisch) nicht richtig wahr nahm und gegen sie fuhr.

Abbildungsverzeichnis

1.1 SYMPARTNER Projekt 3

2.1 Eingesetzter Roboter 8
2.2 Erreichbarkeit 10
2.3 Aufenthaltswahrscheinlichkeit 12
2.4 Varianten des Feedback 14
2.5 Entstehen einer NDT-Zelle 15
2.6 Technische Distanz 17
2.7 Soziale Distanz 17

3.1 Spielfreudige Person 21
3.2 Gestürzte Person 22
3.3 Funktionale Aspekte eines Beobachtungsverhaltens 23
3.4 Beobachtung durch Hausautomation 25
3.5 Suche nach der Person 27

4.1 Beschreibung der Positionsauswahl 33
4.2 Optimierung der Beobachtungsposition 35
(a) Versuchsaufbau 35
(b) Bild von der initialen Position 35
(c) Bild nach optimierter Position 35
4.3 Ergebnis der passiven Beobachtungsposition 37
(a) Positionen mit und ohne Beleuchtungsmodell 37
(b) Schlechte Beleuchtungssituation 37

(c) Gute Beleuchtungssituation 37
4.4 Zusammenspiel der Kostenfunktionen 40
4.5 Gute Warteposition im Einkaufszentrum 42
4.6 Blockdiagramm 44

5.1 Informationsquellen für die Positionssuche 47
5.2 Abbildung auf neue Kostenfunktionen 50
5.3 Dreidimensionale Sichtkegel 52
5.4 Sichtbarkeit in 2D 53
5.5 Eingeschränkter Sichtraum 54
5.6 Kegelförmige Maske 56
5.7 Kreisförmige Maske 57

6.1 Informationsfluss des Beobachtungsverhaltens 59
6.2 Drei Aspekte des Beobachtungsverhaltens 60
(a) Aktivierung 60
(b) Beobachtung 60
(c) Bestätigung mit Fernbedienung 60
6.3 Zustandsmaschine 61

7.1 Schematische Ansicht des Labors 64
7.2 Entscheidende Karten für das neue Verhalten 65
(a) Aufenthaltswahrscheinlichkeit der Person 65
(b) Statische Feedback-Karte 65
7.3 Rechenzeit als Kreisdiagramme 66
(a) 3D Sichtbarkeit 66
(b) 2D Sichtbarkeit 66
(c) Gesamtzeit 66
7.4 Eine normale Anfahrt zu einer Position 67
7.5 Eine Anfahrt mit Hindernis und statischer Aufenthaltskarte 69
7.6 Eine Anfahrt mit Hindernis 69
7.7 Beobachtung der Hypothese 70

A.1 Fotos aus dem *Living Lab* 78
(a) Teeküche mit Sitzgelegenheit 78
(b) Schrankwand in living 2 78
(c) Sitzgarniatur in living 2 78
(d) Sessel in living 1 78
(e) Fernseher in living 1 78
A.2 Ablauf einer Positionsfindung 79
(a) Anfangsverteilung 79
(b) Leicht erhöhte Konzentration 79
(c) Hohe Konzentration 79
(d) Optimale Pose 79
A.3 Optimierung während der Fahrt 80
(a) PoseFinder konvergiert 80
(b) PoseFinder neu initialisiert 80
(c) PoseFinder gestoppt 80
(d) Positionierung des Roboters 80
A.4 Optimierung während der Fahrt mit Hindernis 81
(a) Optimale Position 81
(b) Endposition 81
(c) Hindernis erkannt 81
(d) Eine Alternative 81
A.5 Test Szenario in living 1 82
A.6 Test Szenario in living 2 83
A.7 Test Szenario in kitchen 83

Literaturverzeichnis

[Max] *Tweety and Max homepage.* [online]: https://www.tu-ilmenau.de/neurob/research/robots/tweety-max/ (2017-01-13).

[Biber und Strasser, 2003] Biber, P. und W. Strasser (2003). *The normal distributions transform: a new approach to laser scan matching.* In: *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, Bd. 3, S. 2743–2748 vol.3.

[Bodor et al., 2005] Bodor, R., A. Drenner, M. Janssen, P. Schrater und N. Papanikolopoulos (2005). *Mobile camera positioning to optimize the observability of human activity recognition tasks.* In: *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, S. 1564–1569.

[Bresenham, 1965] Bresenham, J. E. (1965). *Algorithm for Computer Control of a Digital Plotter*. IBM Syst. J., 4(1):25–30.

[Byers et al., 2003] Byers, Z., M. Dixon, K. Goodier, C. M. Grimm und W. D. Smart (2003). *An autonomous robot photographer*. In: *Intelligent Robots and Systems, 2003. (IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, Bd. 3, S. 2636–2641 vol.3.

[Dautenhahn et al., 2006] Dautenhahn, K., M. Walters, S. Woods, K. L. Koay, C. L. Nehaniv, A. Sisbot, R. Alami und T. Siméon (2006). *How May I Serve You? A Robot Companion Approaching a Seated Person in a Helping Context.* In: *Proceedings of the 1st ACM SIGCHI/SIGART Conference on Human-robot Interaction*, HRI '06, S. 172–179, New York, NY, USA. ACM.

[Gellert und Vintan, 2006] Gellert, Arpad und L. Vintan (2006). *Person Movement Prediction Using Hidden Markov Models.* Studies in Informatics and Control 15.

[Gross et al.] Gross, Horst-Michael, A. Bley, B. Klein, S. Meyer, M. Hassenzahl und S. Spittel. *SYMPARTNER – SYMbiose von PAUL und RoboTer CompaNion für eine Emotionssensitive UnteRstützung.* [online]: http://www.sympartner.de/ (last visited: 2016-12-12).

[Hall, 1966] Hall, Edward Twichell (1966). *The hidden dimension.* Doubleday.

[Ishara et al., 2015] Ishara, K., I. Lee und R. Brinkworth (2015). *Mobile robotic active view planning for physiotherapy and physical exercise guidance.* In: *2015 IEEE 7th International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, S. 130–136.

[Ito und Sekiyama, 2015] Ito, M. und K. Sekiyama (2015). *Optimal viewpoint selection for cooperative visual assistance in multi-robot systems.* In: *2015 IEEE/SICE International Symposium on System Integration (SII)*, S. 605–610.

[Kennedy und Eberhart, 1995] Kennedy, James und R. Eberhart (1995). *Particle swarm optimization.* In: *Neural Networks, 1995. Proceedings., IEEE International Conference on*, Bd. 4, S. 1942–1948 vol.4.

[Kessler et al., 2012] Kessler, Jens, D. Iser und H.-M. Gross (2012). *I'll Keep You in Sight: Finding a Good Position to Observe a Person.* In: *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, S. 4392–4398. IEEE.

[Kessler et al., 2013] Kessler, Jens, M. Schmidt, S. Helsper und H.-M. Gross (2013). *I'm Still Watching You: Update on Observing a Person in a Home Environment.* In: *Europ. Conf. on Mobile Robots (ECMR).*

[Kessler et al., 2011] Kessler, Jens, Ch. Schroeter und H.-M. Gross (2011). *Approaching a Person in a Socially Acceptable Manner Using a Fast Marching Plan-*

ner. In: *Int. Conf. on Intelligent Robotics and Applications (ICIRA)*, Bd. 7102 d. Reihe *LNCS*, S. 368–377. Springer.

[Kitade et al., 2013] Kitade, T., S. Satake, T. Kanda und M. Imai (2013). *Understanding suitable locations for waiting*. In: *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, S. 57–64.

[Koay et al., 2007] Koay, Kheng Lee, E. A. Sisbot, D. S. Syrdal, M. L. Walters, K. Dautenhahn und R. Alami (2007). *Exploratory Study of a Robot Approaching a Person in the Context of Handing Over an Object.*. In: *AAAI spring symposium: multidisciplinary collaboration for socially assistive robotics*, S. 18–24.

[Kruse et al., 2013] Kruse, Thibault, A. K. Pandey, R. Alami und A. Kirsch (2013). *Human-aware robot navigation: A survey*. Robotics and Autonomous Systems, 61(12):1726 – 1743.

[O'Rouke, 1987] O'Rouke, Joseph (1987). *Art Gallery Theorems and Algorithms*. Oxford University Press.

[Saarinen et al., 2013] Saarinen, Jari P., H. Andreasson, T. Stoyanov und A. J. Lilienthal (2013). *3D normal distributions transform occupancy maps: An efficient representation for mapping in dynamic environments*. The International Journal of Robotics Research, 32(14):1627–1644.

[Schroeter et al., 2009] Schroeter, Christof, M. Hoechemer, St. Mueller und H.-M. Gross (2009). *Autonomous Robot Cameraman – Observation Pose Optimization for a Mobile Service Robot in Indoor Living Space*. In: *IEEE Int. Conf. on Robotics and Automation (ICRA)*, S. 424–429. IEEE.

[Scott et al., 2003] Scott, William R., G. Roth und J.-F. Rivest (2003). *View Planning for Automated Three-dimensional Object Reconstruction and Inspection*. ACM Comput. Surv., 35(1):64–96.

[Volkhardt und Gross, 2013] Volkhardt, Michael und H.-M. Gross (2013). *Finding People in Home Environments with a Mobile Robot*. In: *Europ. Conf. on Mobile Robots (ECMR)*.

[Walters et al., 2007] Walters, M. L., K. Dautenhahn, S. N. Woods und K. L. Koay (2007). *Robotic etiquette: Results from user studies involving a fetch and carry task*. In: *Human-Robot Interaction (HRI), 2007 2nd ACM/IEEE International Conference on*, S. 317–324.