

SITUATIONSVERVERSTEHEN  
STEPHAN PULS FÜR DIE  
RISIKOBEURTEILUNG  
BEI DER  
MENSCH-ROBOTER-KOOPERATION



Scientific  
Publishing



Stephan Puls

SITUATIONSVERSTEHEN FÜR DIE RISIKOBEURTEILUNG  
BEI DER MENSCH-ROBOTER-KOOPERATION



# Situationsverstehen für die Risikobeurteilung bei der Mensch-Roboter-Kooperation

von  
Stephan Puls

Dissertation, Karlsruher Institut für Technologie (KIT)  
Fakultät für Informatik, 2014

Tag der mündlichen Prüfung: 16. Juli. 2014

Referenten: Prof. Dr.–Ing. Heinz Wörn, Prof. Dr.–Ing. Tamim Asfour

#### Impressum



Karlsruher Institut für Technologie (KIT)  
KIT Scientific Publishing  
Straße am Forum 2  
D-76131 Karlsruhe

KIT Scientific Publishing is a registered trademark of Karlsruhe  
Institute of Technology. Reprint using the book cover is not allowed.

[www.ksp.kit.edu](http://www.ksp.kit.edu)



*This document – excluding the cover – is licensed under the  
Creative Commons Attribution-Share Alike 3.0 DE License  
(CC BY-SA 3.0 DE): <http://creativecommons.org/licenses/by-sa/3.0/de/>*



*The cover page is licensed under the Creative Commons  
Attribution-No Derivatives 3.0 DE License (CC BY-ND 3.0 DE):  
<http://creativecommons.org/licenses/by-nd/3.0/de/>*

Print on Demand 2015

ISBN 978-3-7315-0299-9  
DOI 10.5445/KSP/1000044584





# Situationsverstehen für die Risikobeurteilung bei der Mensch-Roboter-Kooperation

zur Erlangung des akademischen Grades eines

**Doktors der Ingenieurwissenschaften**

der Fakultät für Informatik

des Karlsruher Instituts für Technologie (KIT)

**genehmigte**

**Dissertation**

von

**Stephan Puls**

aus Schwerin

Tag der mündlichen Prüfung:

16. Juli 2014

Erster Gutachter:

Prof. Dr.-Ing. Heinz Wörn

Zweiter Gutachter:

Prof. Dr.-Ing. Tamim Asfour



## Zusammenfassung

Industrieroboter werden heutzutage sehr verbreitet in der Produktion eingesetzt. Jedoch arbeiten diese Roboter meist isoliert und unter stark kontrollierten Bedingungen, damit eine Gefahr für den Menschen ausgeschlossen werden kann. Damit jedoch die kognitiven Fähigkeiten und Fingerfertigkeit von Menschen mit der Ausdauer, Stärke und Genauigkeit von Robotern in einer Kooperation zusammengeführt werden können, ist es nötig, dass das Robotersystem seine Umgebung wahrnehmen und verstehen kann, um auf sichere Weise darauf zu reagieren. Das Ziel dieser Arbeit besteht darin, durch Verstehen der vorliegenden Situation, in der sich Mensch und Roboter zusammen befinden, eine zielgerichtete, sichere Mensch-Roboter-Kooperation zu ermöglichen. Eingebettet ist diese Arbeit in ein am Institut für Prozessrechentechnik, Robotik und Automation entwickeltes Rahmenwerk genannt MAROCO<sup>1</sup>. Dieses ermöglicht mittels über der Roboterzelle montierten 3D-Kamerasensorik die Erkennung und kinematische Rekonstruktion eines Menschen, der sich im Arbeitsraum des Roboters aufhält.

Der Auswertezyklus zum Verstehen sowie Interpretieren von Situationen und Planung sicherer Roboterverhalten wird im Rahmen dieser Arbeit als ein kognitiver Prozess verstanden. Teile dieses Prozesses stellen die Wahrnehmung, das wissensbasierte Schlussfolgern und die Ausführung von Folgerungsergebnissen dar.

Die bestehende Umgebungswahrnehmung des Robotersystems wurde durch eine Methode zur Objekterkennung erweitert. Für die Erkennung von größeren Objekten im Arbeitsraum, wie bspw. Arbeitstische, Transportwagen und Stühle, wurde ein markenloses Verfahren entwickelt, das auf neuronalen Netzen als assoziative Speicher basiert und bei der Erkennung die Tiefengestalt des Objekts berücksichtigt.

Darüber hinaus wurde in dieser Arbeit eine Aktionserkennung mittels mehrschichtigen Hidden Markov Modellen integriert, da vom Menschen durchgeführte Handlungen und Aktionen eine wesentliche Grundlage darstellen, um Aktivitäten zu beschreiben und Informationen über die vorliegende Situation zu gewinnen. Ebenfalls wurde eine Methode entwickelt, die eine semantische Abbildung der menschlichen Position auf arbeitsrelevante

---

<sup>1</sup>für *Human-Robot-Cooperation*

Orte im Arbeitsraum durchführt. Es ist weiterhin möglich, Positionswechsel zu erkennen und den Zielort prädiktiv zu schätzen.

Algorithmen zur Handlungs- und Objekterkennung anhand verrauschter, realer Sensordaten können keine Fehlerfreiheit erreichen. Um daher einen Teil der Fehler behandeln zu können, werden Handlungen mit Objekten im Verbund betrachtet. Angelehnt an Handlungsplanungsalgorithmen werden erkannte Handlungen mit Vorbedingungen verknüpft, welche im Nachhinein überprüft werden können. Auf diese Weise können Widersprüche aufgedeckt werden.

Das allgemeine Domänenwissen über mögliche Aktivitäten, Szenenobjekte und Situationen ist in einer ontologischen Wissensbasis verankert. Während der Inferenz mittels Beschreibungslogik wird aktuelles Wissen über den Weltzustand mit dem Domänenwissen verknüpft. Entsprechend können Aussagen über die vorliegende Situation gefolgert werden. Darauf aufbauend werden Handlungsvorgaben für den Roboter generiert, so dass abhängig von menschlichem Verhalten zielgerichtete Roboterreaktionen möglich sind. Diese Handlungsvorgaben dienen einem Aufgabenplanungsmodul als Handlungsziele. Das Planungsmodul bestimmt daraus geeignete Handlungsabfolgen für den Roboter.

Die Risikobeurteilung wurde dahingehend grundlegend erweitert, dass das gefolgerte Situationswissen in die Risikobestimmung rückgekoppelt wird. Dazu wurde die bereits modellierte Regelbasis in unscharfer Logik durch eine weitere Regelmenge ergänzt, die situativ eine Hemmung der Risikobeurteilung erwirken kann. Auf diese Weise kann eine räumlich enge Kooperation erreicht werden. Darüber hinaus wirken erkannte Widersprüche bei Erkennungsergebnissen auf die Risikobeurteilung, um eine Hemmung im Falle von Fehlerkennungen zu verhindern.

Bewegungsvorgaben aus der Aufgabenplanung werden als Eingabe für ein Pfadplanungsmodul genutzt. Die Pfadplanung ist reaktiv und sucht unter Berücksichtigung der erweiterten Risikoschätzung und der erkannten Objekte einen kollisionsfreien Bewegungspfad im Konfigurationsraum des Roboters. Bei der Suche und während der Roboterbewegung wird das Geschwindigkeitsprofil des Roboters an die jeweilige Situation angepasst. Auf diese Weise können Gefahren für den Menschen minimiert werden.

Durch die Handlungserkennung und das Situationsverständnis können Rückschlüsse auf das Verhalten des Menschen im Arbeitsraum gezogen werden. Mittels Prozessanalyse des Geschehens ist es möglich, ein Prozessmodell zu entwerfen. In diesem Modell werden temporale Zusammenhänge zwischen Ereignissen dargestellt, aus denen wiederkehrende Handlungsfolgen entdeckt werden.

Die einzelnen Komponenten des Systems wurden evaluiert und die Ergebnisse analysiert. Wie die Evaluation der Komponenten zeigt, wird durch das System ein reaktives, situationsabhängiges und sicheres Roboterverhalten erreicht.



## Danksagung

Diese Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Institut für Prozessrechentechnik, Automation und Robotik (IPR) des Karlsruher Instituts für Technologie (KIT).

Ich danke herzlich meinem Doktorvater, Herrn Professor Dr.–Ing. Heinz Wörn, für die ermöglichten Freiräume, für die Möglichkeit Lehrveranstaltungen zu übernehmen, für seinen unermüdlichen Einsatz im Bereich der Robotik und für seine Unterstützung bei Lehrprojekten, die das Näherbringen von Studenten in das vielfältige Gebiet der Robotik ermöglicht. Außerdem danke ich Herrn Professor Dr.–Ing. Tamim Asfour für die Übernahme des Korreferats.

Herzlich danken möchte ich auch allen Studenten, die die von mir betreuten Aufgaben annahmen und mit viel Energie bearbeiteten. Die interdisziplinäre Teilnahme von Elektrotechnikern, Maschinenbauern und Informatikern an Lehrprojekten war eine motivierende Erfahrung.

Für die kreative und konstruktive Zusammenarbeit am Institut bin ich speziell Daniel Braun, Stefan Escaida, Björn Hein und Stephan Irgenfried dankbar, mit denen neue, interessante Projektideen entworfen und erfolgreich eingereicht wurden. Der unermüdliche Einsatz von Elke Franzke und Nina Maizik am Institut war eine große Unterstützung, für die ich sehr dankbar bin.

Ein ganz besonderer Dank geht an meine Familie – meiner Frau, die mir mit ihrer Liebe und Begeisterung den Rücken stärkt, meiner Tochter, die mit ihrem Lächeln und Frohsein die Welt verzaubert, und meinen Eltern, die mir unentwegt mit Unterstützung und Rat zur Seite stehen.

Karlsruhe im November 2014,

Stephan Puls



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Problemstellung, Zielsetzung und Hypothese . . . . .	2
1.3	Überblick über die eigene Methode . . . . .	4
1.4	Gliederung der Arbeit . . . . .	4
<b>2</b>	<b>Stand der Technik und Forschung</b>	<b>7</b>
2.1	Normative Vorgaben und Stand der Technik . . . . .	7
2.1.1	Normative Rahmenbedingungen . . . . .	8
2.1.2	Stand der technischen Systeme . . . . .	9
2.2	Methoden zur Risikoanalyse und Kollisionsvermeidung . . . . .	10
2.3	Mensch-Roboter-Kooperationssysteme . . . . .	13
2.4	Zusammenfassung . . . . .	16
<b>3</b>	<b>Systemarchitektur</b>	<b>19</b>
3.1	Grundlagen und Vorarbeiten . . . . .	19
3.1.1	Technische Voraussetzungen des MAROCO Systems . . . . .	20
3.1.2	Wesentliche Systemkomponenten . . . . .	21
3.2	Eigene Systemkomponenten . . . . .	23
3.2.1	Erweiterte Komponenten . . . . .	23
3.2.2	Neue Komponenten . . . . .	25
<b>4</b>	<b>Umweltwahrnehmung im Arbeitsraum</b>	<b>29</b>
4.1	Registrierung . . . . .	29
4.1.1	Ablauf und Methodik . . . . .	29
4.1.2	Filterung, Segmentierung und Bewegungsformerkennung . . . . .	32
4.1.3	Evaluation und Diskussion . . . . .	36
4.2	Erkennung von Szeneobjekten . . . . .	39
4.2.1	Problemstellung . . . . .	40

4.2.2	Stand der Forschung . . . . .	40
4.2.3	Objekterkennung . . . . .	42
4.2.3.1	Das neuronale Netz . . . . .	42
4.2.3.2	Objekterkennung . . . . .	45
4.2.4	Evaluation und Diskussion . . . . .	49
4.2.4.1	Erkennung ohne Vorklassifikation . . . . .	50
4.2.4.2	Erkennung mit Vorklassifikation . . . . .	51
4.2.4.3	Laufzeitanalyse . . . . .	54
4.2.4.4	Diskussion . . . . .	56
<b>5</b>	<b>Handlungserkennung und Situationsverstehen</b>	<b>59</b>
5.1	Begriffsbildung . . . . .	59
5.2	Erkennung und Interpretation der menschlichen Position . . . . .	63
5.2.1	Stand der Forschung . . . . .	64
5.2.2	Ortsbestimmung, Lokalisierung und Prädiktion . . . . .	65
5.2.3	Evaluation und Diskussion . . . . .	67
5.2.3.1	Lokalisierung . . . . .	67
5.2.3.2	Prädiktion . . . . .	68
5.2.3.3	Diskussion . . . . .	68
5.3	Erkennung von Handlungen und Aktionen . . . . .	69
5.3.1	Stand der Forschung . . . . .	69
5.3.2	Kontinuierliche Handlungserkennung . . . . .	71
5.3.2.1	Hidden Markov Modelle . . . . .	71
5.3.2.2	Die Erkennungsmethode . . . . .	73
5.3.3	Evaluation und Diskussion . . . . .	74
5.4	Situationsverstehen durch Schlussfolgerung . . . . .	75
5.4.1	Stand der Forschung . . . . .	75
5.4.2	Modellierte Wissensbasis und Beschreibungslogiken . . . . .	77
5.4.2.1	Beschreibungslogiken . . . . .	77
5.4.2.2	Wissensbasis . . . . .	78
5.4.3	Das Schlussfolgerungssystem . . . . .	81
5.4.3.1	Folgerungssysteme und Schnittstellen . . . . .	82
5.4.3.2	Das Folgerungsmodul . . . . .	82
5.4.4	Untersuchung zur Optimierung durch inkrementelle Folgerung . . . . .	86
5.4.5	Evaluation und Diskussion . . . . .	87
5.4.5.1	Ergebnisse . . . . .	88
5.4.5.2	Diskussion . . . . .	89

5.5	Ermittlung von Handlungszielen für das Robotersystem . . . . .	91
5.5.1	Reaktive Handlungsplanung . . . . .	91
5.5.2	Evaluation und Diskussion . . . . .	93
5.6	Analyse des laufenden Arbeitsprozesses . . . . .	97
5.6.1	Stand der Forschung . . . . .	98
5.6.2	Prozessanalyse . . . . .	100
5.6.2.1	Heuristic Miner . . . . .	100
5.6.2.2	Online Analyse . . . . .	101
5.6.3	Prozessfolgengermittlung . . . . .	102
5.6.4	Evaluation und Diskussion . . . . .	105
5.6.4.1	Qualitätskriterium für Heuristic Miner . . . . .	105
5.6.4.2	Analyse anhand synthetischer Daten . . . . .	106
5.6.4.3	Analyse anhand Sensordaten . . . . .	107
5.6.4.4	Prozessfolgengermittlung . . . . .	111
<b>6</b>	<b>Sicherheit und Risikobeurteilung</b>	<b>113</b>
6.1	Situationsabhängige Risikobeurteilung . . . . .	113
6.1.1	Grundlagen . . . . .	114
6.1.2	Modellierung der Situationsabhängigkeit . . . . .	115
6.1.3	Evaluation und Diskussion . . . . .	118
6.2	Plausibilitätsprüfung . . . . .	121
6.2.1	Stand der Forschung . . . . .	121
6.2.2	Methode zur Plausibilitätsprüfung . . . . .	123
6.2.3	Adaptierte Risikobeurteilung . . . . .	125
6.2.4	Evaluation und Diskussion . . . . .	125
6.2.4.1	Experimentaufbau . . . . .	126
6.2.4.2	Analyse mittels Sensordaten . . . . .	126
6.2.4.3	Analyse mittels synthetischer Inkonsistenzen	128
6.2.4.4	Diskussion . . . . .	129
6.3	Reaktives Roboterverhalten . . . . .	130
6.3.1	Stand der Forschung . . . . .	130
6.3.2	Die Bahnplanungsmethode . . . . .	132
6.3.2.1	Die Offline-Phase . . . . .	132
6.3.2.2	Die Online-Phase . . . . .	133
6.3.3	Reaktivität durch Bewegungsplanung . . . . .	137
6.3.4	Evaluation und Diskussion . . . . .	138

<b>7 Zusammenfassung und abschließende Diskussion</b>	<b>145</b>
7.1 Zusammenfassung der Ergebnisse . . . . .	145
7.2 Abschließende Diskussion . . . . .	147
<b>Abbildungsverzeichnis</b>	<b>151</b>
<b>Tabellenverzeichnis</b>	<b>153</b>
<b>Literaturverzeichnis</b>	<b>155</b>

# 1 Einleitung

## 1.1 Motivation

Technologie unterstützt den Menschen in vielen Bereichen des täglichen Lebens. Speziell einfache, sich wiederholende Arbeitsprozesse werden heutzutage durch Maschinen und Roboter durchgeführt. Seit den siebziger Jahren werden Robotersysteme in der Industrie eingesetzt, um Prozesse teilweise zu automatisieren und entsprechend Kosten zu senken. Auf Grund der Eigenschaften wie Stärke, Verlässlichkeit und Genauigkeit von Robotersystemen hat sich ein breites Spektrum an Einsatzgebieten entwickelt. Folglich sind Roboter ein Teil der täglichen Routine in großen industriellen Betrieben.

Dennoch sind menschliche Arbeitskräfte wegen ihrer Fingerfertigkeit und kognitiven Leistungsvermögen flexibler, dynamischer und anpassungsfähiger. Neue Prozesse und auftretende Fehler können durch Menschen erkannt und entsprechende Reaktionen veranlasst werden. Darüber hinaus stellen Anschaffungskosten und geringe Flexibilität von verfügbaren Automatisierungslösungen hohe Hürden dar, die kleinere und mittelständische Firmen davon abhalten ihre Prozesse zu automatisieren. Das Zusammenführen von menschlichen Leistungen und den Vorteilen der Robotik in der industriellen Mensch-Roboter-Kooperation (MRK) kann das Fehlen von Flexibilität überbrücken und bessere Arbeitsbedingungen für den Menschen schaffen.

## 1.2 Problemstellung, Zielsetzung und Hypothese

Kooperation entsteht durch die Wechselwirkungen von Handlungen verschiedener Partner, wenn die Handlungen auf ein gemeinsames Ziel hinwirken. Entsprechend beeinflussen sich die Aktionen der Partner und müssen aufeinander abgestimmt sein [Axelrod, 1985]. In der Kooperation wird von den Partnern erwartet, dass die Handlungen der jeweils anderen nicht widersprüchlich sind. Folglich ist der Umfang einer Kooperation auf eine Domäne, einen Umstand oder eine Situation beschränkt.

Dieser Erkenntnis von Axelrod folgend, wird im Rahmen dieser Arbeit untersucht, wie Situationsverständnis erreicht werden kann, um Kooperation zwischen Mensch und Roboter zielgerichtet im industriellen Kontext zu erreichen.

Nach der Definition von Eaton kann Kooperation darüber hinaus als direkt oder indirekt eingeteilt werden [Eaton, 1948]. Diese Unterscheidung basiert auf der Art der auftretenden Handlungen der Kooperationspartner. Direkte Kooperation liegt vor, wenn die Partner die gleichen Handlungen durchführen. Entsprechend handelt es sich um indirekte Kooperation, wenn unterschiedliche Aufgaben durch die Kooperationspartner gelöst werden. Da bei der industriellen MRK verschiedene Teilaufgaben durch Mensch und Maschine abgearbeitet werden sollen, um die Stärken beider zu vereinen, handelt es sich folglich um indirekte Kooperation.

Kooperative Situationen lassen sich daher nicht auf Grund der Ähnlichkeit menschlicher Tätigkeiten zur Roboteraufgaben erkennen. Vielmehr ist es nötig, den Kontext einer Situation zu erfassen, um diese bewerten zu können. Der Kontext setzt sich dabei aus menschlichen Handlungen, räumlichen Bezügen zu Arbeitsplätzen im Roboterarbeitsraum und zur Verfügung stehender bzw. genutzter Objekte zusammen.

Darüber hinaus ist es auch wichtig, dass der Roboter keine Gefahr für den Menschen darstellt. Was dabei als gefährlich betrachtet wird, hängt gleichermaßen von der Situation ab. Bei gewünschtem Kontakt zwischen Mensch und Maschine kann von einem geringeren Risiko ausgegangen werden, als wenn der Mensch abgelenkt oder auf seine eigene Aufgabe konzentriert ist.

Das Verstehen und Interpretieren von Situationen im industriellen Kontext wird im Rahmen dieser Arbeit als ein kognitiver Prozess verstanden. Teile dieses Prozesses stellen die Wahrnehmung, das wissensbasierte Schlussfolgern und die Umsetzung von Folgerungsergebnissen dar. Die Zielsetzungen dieser Arbeit, die sich aus diesen Überlegungen ableiten, können wie folgt zusammengefasst werden:

**Wahrnehmung** Zum einen sollen menschliche Handlungen erkannt werden, da diese die wesentliche Basis darstellen, um Aktivitäten zu beschreiben und Informationen über die vorliegende Situation zu gewinnen. Zum zweiten soll eine semantische Abbildung der menschlichen Position auf arbeitsrelevante Orte im Arbeitsraum durchgeführt werden, da möglicherweise gleiche Handlungen an verschiedenen Orten unterschiedliche Bedeutungen aufweisen können. Zum dritten sollen Objekte im Arbeitsraum des Roboters erfasst und identifiziert werden, um kollisionsfreie Bahnplanung zu ermöglichen und um diese beim Schlussfolgern berücksichtigen zu können.

**Schlussfolgerung** Die Informationen der Wahrnehmung sollen als Grundlage dienen, um vorliegende Situationen erkennen zu können. Innerhalb der Domäne der indirekten MRK werden verschiedene Situationen modelliert. Darüber hinaus sollen auf dieser Grundlage Handlungsziele für das Robotersystem gefolgert und deren Umsetzung geplant werden.

**Umsetzung** Die Handlungsvorgaben für das Robotersystem sollen ausgeführt werden, ohne dabei ein Risiko für den Menschen zu erzeugen. Entsprechend ist die Rückkopplung des Situationswissens in eine reaktive Roboterpfadplanung nötig.

**Prozessanalyse** Beobachtete Handlungen des Menschen in ihrem jeweiligen Kontext sollen analysiert werden, um Prozessmodelle während der Kooperation zu ermitteln. Diese Modelle sollen die automatische Prozessformalisierung und die Fehlererkennung in Arbeitsabläufen ermöglichen.

## **Hypothese**

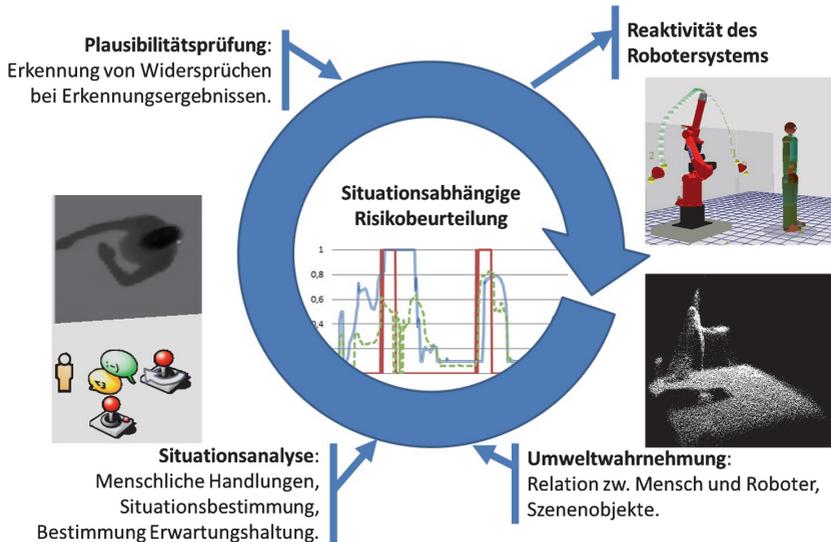
Das Wissen um das Geschehen in der Arbeitszelle lässt Rückschlüsse auf die vorliegende Situation zu, wodurch ein sicheres Miteinander und Interagieren von Mensch und Roboter ermöglicht wird, ohne räumliche oder zeitliche Trennung der Kooperationspartner vorauszusetzen. Die Aufhebung solcher Trennungen führt zu erhöhter Effizienz und ermöglicht eine flexiblere Anpassungsfähigkeit des Produktionssystems.

## **1.3 Überblick über die eigene Methode**

In Abbildung 1.1 ist ein Überblick über die Methode graphisch dargestellt. Im Zentrum der Prozessschleife steht die situationsabhängige Risikobeurteilung. Diese nutzt als Eingabe die Ergebnisse der Umweltwahrnehmung, des Situationsverstehens und der Plausibilitätsprüfung. Die Umweltwahrnehmung dient zur Erfassung der momentanen Roboterumgebung im Arbeitsraum. Die Situationsanalyse ermittelt vorliegende Handlungen und entscheidet mit dem Wissen über vergangene Ereignisse über die vorliegende Situation. Daraus werden ebenfalls Erwartungen an das Robotersystem bestimmt. Die Plausibilitätsprüfung erkennt anhand beobachteter Ereignisse und Szenenobjekte, ob Widersprüche aufgetreten sind, so dass Fehler bei der Erkennung gefolgert werden können. Das Ergebnis ist ein sicheres, reaktives Robotersystem, welches sich adaptiv passend zur Situation verhält.

## **1.4 Gliederung der Arbeit**

Im Folgenden der Arbeit werden die Herangehensweisen und Lösungen zu den genannten Zielsetzungen erörtert und diskutiert. Im Kapitel 2 werden hierzu Arbeiten vorgestellt, die sich mit ähnlichen Problemstellungen im Einzelnen oder Ganzen beschäftigen. Darüber hinaus wird auch der aktuelle Stand der momentan geltenden Normen und Vorschriften im Bereich der MRK vorgestellt. Im Kapitel 3 werden die Grundlagen und Vorarbeiten beschrieben, auf denen diese Arbeit aufbaut. Darauf folgend werden die



**Abbildung 1.1:** Überblick über die Methode. Im Zentrum der Prozessschleife steht die situationsabhängige Risikobeurteilung. Als Eingabe dienen hierzu Ergebnisse der Umweltwahrnehmung, des Situationsverstehens und der Plausibilitätsprüfung. Das Ergebnis ist ein sicheres, reaktives Robotersystem.

eigenen Systemkomponenten und ihre Bezüge untereinander detailliert dargestellt.

Die Methoden zur Wahrnehmung von Objekten und der Registrierung des Kamerasystems zum Robotersystem werden im Kapitel 4 dargelegt. Methoden zur Handlungserkennung und zum Situationsverstehen werden im Kapitel 5 erläutert. Im Kapitel 6 wird vorgestellt, wie das Wissen um die Situation für eine angepasste Risikoermittlung genutzt werden kann. Darauf aufbauend wird ebenfalls auf die entwickelte Methode zur reaktiven Bahnplanung eingegangen.

Im Kapitel 7 werden die Ergebnisse der Arbeit zusammengefasst und diskutiert. Es wird ebenso ein Ausblick gegeben, der mögliche zukünftige Forschungsarbeiten und -richtungen aufzeigt.

## **2 Stand der Technik und Forschung**

Die Vision der intuitiven und einfachen Interaktion zwischen Mensch und Maschine ist seit langem die Triebfeder für die Forschung im Bereich der Mensch-Roboter-Kooperation (MRK). Viele Forschergruppen haben ihre Arbeit dieser Vision verschrieben, wodurch sich neue Forschungsbereiche aufgetan haben. Im Folgenden werden einige themenverwandte Arbeiten speziell aus dem Bereich der industriellen MRK vorgestellt. Zu Anfang wird eine Übersicht gegeben, welche normativen Vorgaben gelten und welche Systeme dem Stand der Technik entsprechen. Anschließend werden Arbeiten vorgestellt, die sich mit Methoden der Kollisionserkennung und -vermeidung beschäftigen, um Sicherheit im Bereich der industriellen MRK zu gewährleisten. Daraufhin wird ein Überblick gegeben, welche Kooperationssysteme bereits Gegenstand der Forschung und Entwicklung sind. Dieser Überblick erhebt jedoch keinen Anspruch auf Vollständigkeit und befasst sich mit für die vorliegende Abhandlung relevanten Arbeiten.

### **2.1 Normative Vorgaben und Stand der Technik**

Bei der Mensch-Roboter-Kooperation muss die Sicherheit für den Menschen gewährleistet sein, da entstehende Prozesskräfte und -geschwindigkeiten hohe Gefahrenpotentiale bergen können. Im Folgenden werden daher grundlegende Techniken und Arbeiten vorgestellt, die sich mit der Sicherheit des Menschen im Arbeitsraum des Roboters beschäftigen.

## 2.1.1 Normative Rahmenbedingungen

Der industrielle Robotereinsatz unterliegt normativen Rahmenbedingungen, wobei die Sicherheitsanforderungen an Industrieroboter in der Norm *DIN EN ISO 10218* festgehalten sind. Diese Norm besteht aus zwei Teilen: ISO 10218-1 definiert Kollaborationsarten und legt technische Anforderungen an die Sicherheitsfunktionen des Roboters fest. In der ISO 10218-2 wird das Robotersystem und der Arbeitsraum mit einbezogen.

Es sind vier Kollaborationsarten durch die Norm festgelegt:

### 1 - Stopp des Roboters mit automatischem Wiederanlauf

„Der Roboter muss anhalten, wenn sich eine Person im Kollaborationsraum befindet.[...] Der Roboter kann den Automatikbetrieb wiederaufnehmen, wenn die Person den Kollaborationsraum verlässt.“

**2 - Handführen** „Falls vorgesehen, muss sich die Ausrüstung für Handführung nahe am Endeffektor befinden [...]. Der Roboter muss bei eingeschalteter Funktion zur sicherheitsbewerteten überwachten Geschwindigkeit betrieben werden. Der obere Grenzwert der [...] Geschwindigkeit muss durch die Risikobeurteilung festgelegt werden.“

**3 - Geschwindigkeits- und Abstandsüberwachung** „Der Roboter muss eine festgelegte Geschwindigkeit und einen festgelegten Abstand zur Bedienperson einhalten. Diese Funktionen können integriert sein oder durch die Kombination externer Eingangssignale realisiert werden.“

**4 - Leistungs- und Kraftbegrenzung** „[...] durch inhärente Konstruktion oder Steuerung [...] Wird ein Grenzwert überschritten, muss ein Sicherheitshalt ausgelöst werden. Der Roboter ist lediglich eine Komponente in einem betriebsfertig kollaborierenden Robotersystem und ist an sich unzureichend für den sicheren kollaborierenden Betrieb. Die Anwendung des kollaborierenden Betriebs sind dynamisch und müssen durch die Risikobeurteilung, die während der Auslegung des Anwendungssystems durchgeführt wird, festgelegt werden. [...] ISO

10218-2 muss bei der Gestaltung des kollaborierenden Betriebs angewendet werden.“

In der Fassung der Norm aus dem Jahr 2006 wurden für die Grenzwerte konkrete Angaben gemacht. Mit der Neuerung der Norm, veröffentlicht im Jahr 2011, werden keine konkrete Grenzwerte genannt. Stattdessen wird auf den zweiten Teil der ISO Norm verwiesen. Dieser betrachtet sich aus dem Prozess ergebende Gefährdungen bspw. durch Werkzeuge, Vorrichtungen und Werkstücke. Ebenfalls werden alle Personen miteinbezogen, die Zutritt zum Roboter haben und deren Positionen zum Roboter.

Des Weiteren befindet sich die ISO/TS<sup>1</sup> 15066 in der Vorbereitung, in der zusätzliche Informationen enthalten sein sollen. Dabei soll die TS 15066 dem Endanwender Orientierung für die Gefährdungsbeurteilung geben und wird unter anderem voraussichtlich Belastungsgrenzwerte des Menschen für den Kontakt von Roboter und Mensch enthalten.

Es ist festzuhalten, dass ausgehend von einer Risikobeurteilung Grenzwerte für die Robotersteuerung abgeleitet werden. Dabei spielen in der Norm wesentliche Gesichtspunkte der gesamten Roboterzelle eine wichtige Rolle. Im Vergleich zu dem Ansatz, der im Rahmen dieser Arbeit verfolgt wird, sind somit Parallelen und Unterschiede festzustellen. Die hauptsächliche Parallele liegt in der Berücksichtigung der möglichen Risiken. Jedoch wird in der vorliegenden Arbeit das Risiko anhand der Situation bestimmt und nicht allgemein für den Prozess a priori ermittelt. Entsprechend kann die Effizienz und Bedienerfreundlichkeit gesteigert werden.

Darüber hinaus sind die Grenzwerte und Roboterhaltensvorgaben hauptsächlich passiv definiert im Gegensatz zum pro-aktiven Roboterverhalten, das bspw. durch Pfadumplanung in dieser Arbeit erreicht wird. Diese Arbeit greift somit Weiterentwicklungen der normativen Rahmenbedingungen auf und zeigt Wege zur Lösung bestehender Probleme auf.

### 2.1.2 Stand der technischen Systeme

Einige Roboterhersteller sind mittlerweile dazu übergegangen Robotersysteme zu entwickeln, die inhärent sicher sein sollen, um MRK Anwendungen

---

<sup>1</sup>TS - Technische Spezifikation

voranzubringen. Zu diesen Robotersystemen zählen unter anderem *Frida* von ABB, woraus sich das Produkt *YuMi*<sup>2</sup> entwickelt hat, *Baxter* von Rethink Robotics<sup>3</sup> und der *LBR* von KUKA<sup>4</sup>. Dem Anspruch der inhärenten Sicherheit folgend, sind diese Systeme im Vergleich zu herkömmlichen Knickarm- und Mehrachsrobotern wesentlich in ihren Leistungsfähigkeiten, wie Geschwindigkeiten und Nutzlasten, eingeschränkt. Folglich ist die Forschung für die MRK mit inhärent unsicheren Roboterkinematiken weiterhin gefragt, sichere und leistungsfähige Konzepte vorzuschlagen und zu untersuchen.

Ein Konzept, wie Mensch und Industrieroboter auf engem Raum arbeiten können, wird von der Firma MRK-Systeme GmbH entwickelt und Vertrieben. Dazu wurde eine physische Hülle für einen Roboter entwickelt, die kapazitiv menschliche Annäherungen wahrnimmt. Die wesentliche Grundlage hierzu wurde von P. Heiligensetzer entwickelt [Heiligensetzer, 2003]. Im Falle einer Annäherung wird der Roboter verlangsamt bis hin zum Stillstand, falls die Distanz zwischen Mensch und Roboter zu gering wird. Für die manuelle, handgeführte Bewegung des Roboters bietet MRK-Systeme ein Modul mit Kraft-Momenten-Sensoren an, um Nullkraft-geregelt den Roboter zu positionieren. Dennoch ist kein Umweltverständnis mit diesen Methoden zu erreichen, weshalb ein aktives Roboterverhalten für die Kooperation nicht möglich ist.

## 2.2 Methoden zur Risikoanalyse und Kollisionsvermeidung

Die folgenden Ansätze und Arbeiten lassen sich in drei Kategorien einteilen: (1) A-priori Systemanalyse, (2) Kollisionsvermeidung und (3) Kollisionsbehandlung. In der Arbeit von Oberer-Treitz et al. werden die Kategorien (2) und (3) auch als aktive und passive Sicherheit bezeichnet, wobei die passive Sicherheit danach strebt, im Falle einer Kollision mögliche Effekte zu reduzieren [Oberer-Treitz et al., 2010].

---

<sup>2</sup><http://www.abb.com/yumi>

<sup>3</sup><http://www.rethinkrobotics.com/baxter/>

<sup>4</sup><http://www.kuka-lbr-iiwa.com/>

Die a-priori Analyse entspricht häufig einer Gefahrenanalyse anhand der oben beschriebenen Normen und der daraus verbundenen Ableitung von Mechanismen zum Schutz des Menschen. Ein modellbasierter Ansatz zur Risikoanalyse und Sicherheitsfallkonstruktion wird durch Do Hoang et al. vorgestellt [Do Hoang et al., 2012]. Hierfür wird die Erweiterung von UML durch die Risikoanalysemethode HAZOP eingeführt, die fallbasiert Gefahren eines Systems ermittelt. Die Methode wird an einem Rollatorassistenten demonstriert und ermöglicht die Identifizierung notwendiger Sicherheitsbetrachtungen und -nachweise für das technische Gesamtsystem bestehend aus Software, Hardware, mechanischem Design und Benutzer.

Zur Kollisionsvermeidung gehört eine Arbeitsraumüberwachung, um mögliche physische Kontakte zu präzisieren und Gegenmaßnahmen einzuleiten. Die fortgeschrittensten Systeme beinhalten eine Art Kamerasystem, um die Position des Menschen und möglicher Weise auch dessen Pose im Arbeitsraum bestimmen zu können. Die Firma Pilz hat ein kommerzielles Sicherheitssystem namens *Safety Eye* entwickelt, das auf Stereokamerasehen und virtuellen Zäunen basiert [Pilz GmbH & Co. KG, 2013]. Diese Lösung teilt den Arbeitsraum in verschiedene 3D-Sicherheitsräume. Jedoch können nur Bildänderungen innerhalb dieser Räume erkannt werden. Entsprechend kann der Mensch nicht von anderen Objekten unterschieden werden. Es ist daher nicht möglich, sinnvolle Informationen über die Szene und Situation zu erfahren.

Ein System, das durch Winkler entwickelt wurde, verwendet ein Time-of-Flight-Kamerasystem zur Überwachung der Roboterzelle. Dynamische Sicherheitszonen werden in einer virtuellen Umgebung definiert [Winkler, 2008]. Um das Risiko für den Werker zu reduzieren, kann die maximale Geschwindigkeit des Roboters anhand der Sensorinformation bzgl. der Sicherheitszonen limitiert werden.

In der Arbeit von Thiernemann wird ein Ansatz vorgestellt, der CCD-Kameras einsetzt [Thiernemann, 2005]. Dabei werden die menschlichen Hände verfolgt und mittels eines Neuro-Fuzzy Systems wird entsprechend ein Gefahrenpotential ermittelt. Daraufhin kann die Robotergeschwindigkeit angepasst werden. Die Bestimmung des Gefahrenpotentials ergibt sich aus Neuro-Fuzzy Regeln über überwachte Geschwindigkeiten und Abstände zwischen Mensch und Roboter.

Die Bestimmung eines Gefahrenindex wird durch Kulic und Croft untersucht [Kulić and Croft, 2006]. Dabei wird ausgehend von Abstand zwischen Mensch und Roboter, relative Geschwindigkeit und Trägheit des Roboters ein Gefahrenindex definiert. Dieser wird mittels Potentialfeldmethode zur Pfadplanung genutzt, um möglichst sichere Pfade zu generieren. Dieser Ansatz wird durch Kulic und Croft wiederum erweitert, indem verschiedene Zeithorizonte bei der Sicherheitsbetrachtung berücksichtigt werden [Kulić and Croft, 2007]. Dabei spiegelt die sichere Planung einen langen Zeithorizont wider. Trajektorienanpassung und reaktive Regelung werden für mittel- bzw. kurzfristige Anpassungen verwendet.

Der Faktor Mensch in der MRK zur Untersuchung von Gefahrenpotentialen wird in der Arbeit von Najmaei et al. untersucht [Najmaei et al., 2010]. Dabei werden neben relativen Geschwindigkeiten und Mensch-Roboter-Abständen die menschliche Blickrichtung und Körperorientierung herangezogen, um einen Risikoindex zu erstellen. Anhand dieses Index wird der Roboterpfad angepasst, wodurch höhere Abstände eingehalten werden, wenn der Mensch nicht zum Roboter schaut.

Ein sogenanntes *kinetostatisches Gefahrenfeld* wird durch [Lacevic and Rocco, 2010] vorgestellt. Auf Grund der Achsbewegungen des Roboters wird ein solches Feld berechnet. Es bezieht die kinetische Energie der Roboterachsen mit ein und definiert ein gerichtetes Potentialfeld. Das Potentialfeld stellt die Gefahrenschätzungen für den menschlichen Werker dar. Jedoch werden nur die Roboterbewegungen betrachtet. Somit werden situationsabhängige Information über Interaktionsprozesse und menschliche Pose nicht berücksichtigt.

Eine ähnliche Idee eines Gefahrenfeldes wird durch [Vogel et al., 2012] verwendet. Dabei kommt ein projektorbasiertes Sensorsystem bei einem Kooperationsarbeitsplatz zum Einsatz. Die notwendige Sicherheitszone um den Roboter wird auf die Werkoberfläche projiziert und kann abhängig von Roboterbewegungsrichtung und Geschwindigkeit angepasst werden. Falls die projizierte Zone unterbrochen wird, stoppt der Roboter. Das System kann jedoch nicht zwischen Objekten und Menschen unterscheiden. Folglich ist eine enge Kooperation zwischen Mensch und Roboter schwer zu erreichen.

In den Arbeiten von Haddadin et al. wird argumentiert, dass Sicherheit für den Menschen durch die Integration von Wissen über Kollisionen und

resultierenden Verletzungen in Regelungssysteme erreicht wird [Haddadin et al., 2008; Haddadin et al., 2010; Haddadin et al., 2012]. Entsprechend ist dieser Ansatz der passiven Sicherheit und somit der Kollisionsbehandlung zuzuordnen. Es wurden Verletzungsuntersuchungen an tierischem Gewebe durchgeführt und Grenzwerte für auftretende Kräfte abgeleitet. Durch die Berücksichtigung und entsprechende Auslegung von Regelungsstrategien kann Sicherheit für den Menschen gewährleistet werden, auch wenn es zu physischen Kontakten kommt. Dieser Ansatz setzt jedoch voraus, dass die verwendeten Roboter den Anforderungen, die durch die Regelung gestellt werden, gerecht werden. Dazu gehört zum einen die notwendige Roboterdynamik und zum anderen leistungsfähige mechatronische Systeme. Solche Anforderungen werden in erster Linie nur durch kleine Leichtbauroboter unterstützt, wie bspw. dem LBR von KUKA.

Durch Oberer-Treitz et al. wurden ebenfalls Verletzungsrisiken anhand von experimentellen Crash-Tests zwischen Dummies und Robotern bestimmt [Oberer-Treitz et al., 2010]. Dabei wurden speziell die Auswirkungen von Kollisionen mit Kopf- und Nackenbereich des Menschen untersucht. Daraus wurden Verletzungskriterien ermittelt, die es erlauben das Gefahrenpotential eines Robotersystems offline zu bestimmen. Entsprechend können Sicherheitsstrategien im Prozess eingesetzt werden, die zu Entscheidungen über Roboter- oder Prozessausführungen im Sinne der Sicherheit führen. Demzufolge zielt dieser Ansatz auf eine a-priori Systemanalyse ab, um während des Prozesses Kollisionen und Gefahren zu vermeiden.

## 2.3 Mensch-Roboter-Kooperationssysteme

Im häuslichen Bereich wird an Haushalts- und Servicerobotern geforscht, die dem Menschen tägliche Aufgaben und Pflichten abnehmen sollen. Zu den bekanntesten Forschungsplattformen in diesem Bereich zählen mitunter der Care-O-Bot [Reiser et al., 2009], der humanoide Roboter Armar [Asfour et al., 2006] und die PR2 Plattform von Willow Garage<sup>5</sup>.

Im industriellen Bereich, in dem Anwendungsbedingungen definierter, kontrollierter und besser vorhersagbar sind, sind keine kommerziell erhältlichen Systeme für die autonome Kooperation verfügbar. Dieser Bereich ist

---

<sup>5</sup><https://www.willowgarage.com/pages/pr2/overview>

noch immer Gegenstand aktiver Forschung, um für den Menschen sichere Systeme zu entwickeln.

Am Anwendungsbeispiel der Automobilproduktion werden durch Shi et al. die möglichen Kooperationsmöglichkeiten zwischen Mensch und Roboter untersucht [Shi et al., 2012]. Dabei werden die aktuellen Zustände beleuchtet und zukünftige Möglichkeiten aufgezeigt. Im Wesentlichen werden drei verschiedene Ebenen der Kooperation unterschieden: *low Level*, *medium Level* und *high Level*, wobei sich die Ebenen durch den Grad der Kooperation voneinander abgrenzen. Somit sind Mensch und Roboter bei der *low Level*-Kooperation integrale Bestandteile der Werkzelle. Mit steigender Ebene entfernen sich Mensch und Maschine voneinander.

In der Arbeit von Thiemermann wird ein System zur Kleinteilemontage mit einem SCARA-Roboter entwickelt und untersucht [Thiemermann, 2005]. Ausgehend von zulässigen Grenzwerten für auftretende Kräfte und Geschwindigkeiten werden ergonomische Einflussgrößen und Potentiale der direkten MRK ausgearbeitet. Mittels kamerabasierter Arbeitsraumüberwachung wird ein sicheres Verfahren des Roboters durch Geschwindigkeitsanpassung erreicht.

Die Kleinteilemontage war ebenfalls Gegenstand der Untersuchungen im EU-Projekt *ROSETTA* [Patel et al., 2012]. Bei den Systembetrachtungen wurden Aspekte der Roboterinstallation, Programmwartung während Ausführung und Einlernen, Sensorintegration in die Programmausführung sowie Sicherheitsverhalten berücksichtigt. Zielsetzung war unter anderem die Ausarbeitung einer Grundlage für die Standardisierung von Sicherheitsvoraussetzungen für kollaborierende Roboter. Dabei wurde das Roboterkonzept *Frida* von ABB entwickelt, wobei auch Verletzungsgefahren im Hinblick auf direkte, kontaktbasierte Kooperation analysiert wurden [Matthias et al., 2011].

Im *rorarob*-Projekt wurde hingegen die Anwendung der MRK bei Schweißaufgaben von Rohr- und Rahmenkonstruktionen untersucht [Busch et al., 2012]. Demzufolge geht es um die Handhabung von häufig sperrigen und schwergewichtigen Bauteilen. Im Zentrum der Betrachtungen standen unter anderem Aspekte zur Arbeitsplanung, ergonomische Arbeitsplatzgestaltung und Prozesssicherheit, wodurch körperliche Beanspruchungen des Werkers reduziert und Belastungen minimiert werden.

In der Arbeit von Graf wird ein System vorgestellt, das sich in der Hauptsache der Arbeitsraumüberwachung in der industriellen MRK widmet [Graf et al., 2010]. Dazu wurde eine Methode entwickelt, um die menschliche Kinematik anhand von Tiefenbildern zu schätzen. Das eingesetzte Kamerasystem, das mittels Lichtlaufzeitmessung Bildtiefen ermittelt, befindet sich an der Decke der Roboterzelle und hat einen Teil des freien Arbeitsraums des Roboters im Blick. Mittels aufwändiger Bildverarbeitungsmethoden, wie der Bestimmung von optischen Flussfeldern und anisotroper Filterung, werden die Tiefendaten verarbeitet. Für die Rekonstruktion kinematischer Parameter werden Methoden vorgestellt, die ausgehend von der Kopfposition die Blickrichtung und die Oberkörperorientierung bestimmen. Auf Grund der Kamerapositionierung senkrecht über dem Menschen können keine Parameter über Rumpf und Beinstellungen ermittelt werden. Sobald der Mensch seine Arme benutzt, werden diese im Bildbereich sichtbar, wodurch ebenfalls die Bestimmung kinematischer Parameter der Arme möglich ist. Die Information über die menschliche Kinematik wird verwendet, um eine Risikomodellierung und -schätzung durchzuführen. Auf deren Grundlage wird ein erster Ansatz für eine sichere Roboterbahnplanung vorgestellt. Folglich sind die Grundlagen für eine überwachte, autonome MRK gelegt, jedoch sind keine Anwendungen, die über das einfache Ausweichen des Roboters hinaus gehen, möglich.

Die industrielle MRK auf kleinem Raum wird in der Arbeit von Lenz untersucht, wobei auch hier die Kleinteilmontage als Anwendungsbeispiel herangezogen wird [Lenz, 2011]. In dem System werden Handpositionen und Blickrichtung des Menschen visuell und modellbasiert verfolgt. Anhand dieser Daten werden menschliche Aktionen modelliert, die den Kontext eines Arbeitsablaufes aufspannen. Anhand des Arbeitsablaufes werden Roboteraktionen geplant und ausgeführt, so dass bspw. weitere Werkstücke dem Menschen zur Verfügung gestellt werden können, sobald dieser einen Montageabschnitt beendet hat. Das sichere Verfahren des Roboters beschränkt sich auf die Einhaltung von Sicherheitsabständen und geringen Geschwindigkeiten.

In den Arbeiten von Tan et al. wird die kooperative Produktion in zellenbasierten Arbeitsräumen untersucht. In [Tan et al., 2010a] wird hierzu die gewünschte Produktionsaufgabe in Teilschritte zerlegt, um kooperative Möglichkeiten zu bestimmen und den kooperativen Prozess zu planen. Die Sicherheitsanalyse anhand des Aufbaus der Werkzelle definiert ent-

sprechende Bereiche, in denen der Roboter Bewegungen ausführen darf [Tan et al., 2010b]. Ebenfalls werden maximale Geschwindigkeiten und minimale Abstände bestimmt. Dieses Vorgehen entspricht im Wesentlichen den normativen Vorgaben. Ähnlich wie bei Lenz werden darüber hinaus menschliche Handlungen analysiert, wodurch Rückschlüsse auf den Zustand des Arbeitsprozesses gewonnen werden. Anhand dessen werden Roboteraktionen ausgelöst [Tan and Inamura, 2013].

Das System, welches durch Alami et al. vorgestellt wird, umfasst ein Gesamtkonzept mit Berücksichtigung von Situations- und Kontextwissen, Handlungsplanung sowie Aktionsausführung [Alami et al., 2014]. Situationen und Kontext stellen hierbei reine geometrische Repräsentationen der Szene dar, wobei Schlussfolgerungen über geometrische Beziehungen gezogen werden. Zur Aufgabenbewältigung wird eine Handlungsplanung basierend auf hierarchischen Aufgabennetzen genutzt, die sogenannte *geteilte* Pläne erzeugt. Diese enthalten unter anderem Aufgaben, die durch Kooperationspartner ausgeführt werden müssen. Somit werden die Aufgaben für den Menschen vom Roboter vorgegeben, wodurch der Mensch zum Assistenten des Roboters wird. Es ist jedoch zu erwarten, dass dadurch die Akzeptanz des Systems bei Benutzern gemindert wird.

In der Arbeit von Jafar et al. wird der emotionale Aspekt der menschlichen Wahrnehmung robotischer Handlungen betrachtet [Jafar et al., 2014], wodurch sich Rückschlüsse auf Akzeptanzverhalten ziehen lassen. Es wurden Probanden befragt, die dem Robotersystem ausgesetzt wurden. Die Experimente sind in Komplexität gestaffelt, wobei die Spannweite von passiver Präsenz des Menschen bis hin zur Roboterprogrammierung reicht. Die resultierenden Bewertungen tendieren zu positiven Haltungen gegenüber dem Robotersystem. Jedoch werden kaum Aussagen zu Einflüssen der Probandenauswahl getroffen.

## 2.4 Zusammenfassung

Im Bereich der industriellen MRK wird aktiv und intensiv geforscht. Die normativen Vorgaben entwickeln sich weiter, um der Zusammenführung von Mensch und Roboter Rechnung zu tragen. Dabei werden jedoch auch

die noch vorhandenen Lücken aufgedeckt, die durch Forschungsarbeiten geschlossen werden müssen.

Die Arbeiten verschiedenster Forschungsgruppen zeigen diverse neue interdisziplinäre Bereiche auf, die untersucht werden. Trotz vielseitiger und langjähriger Forschungsarbeit sind noch immer Herausforderungen zu überwinden, bevor die direkte, kontaktbasierte MRK in der Industrie Einzug hält. Dazu reicht es nicht aus, allein kleine und leistungsgeminderte Robotersysteme zu berücksichtigen. Vielmehr ist es notwendig, möglichst allgemeine Ansätze anzugehen und zu untersuchen. Die aufgeführten Arbeiten zeigen, dass sichere Kooperation nur mit der umfassenden Auswertung und Interpretation der Umweltwahrnehmung zu erreichen ist. In den folgenden Kapiteln soll die vorliegende Arbeit ihren Beitrag dazu leisten.



# 3 Systemarchitektur

Um Verständnis für eine Situation zu entwickeln und Mensch-Roboter-Kooperation zu ermöglichen, bedarf es der ganzheitlichen Betrachtung des Roboterarbeitsraumes. Die Grundlage für diese Betrachtung wurde durch die Entwicklung eines Software-Rahmenwerkes namens MAROCO<sup>1</sup> gelegt. Der Stand von MAROCO zu Beginn der vorliegenden Arbeit wird in der Dissertation von Jürgen Graf dokumentiert [Graf, 2010]. Im Folgenden sollen die wesentlichen Aspekte des Rahmenwerks eingeführt und auf die relevanten Vorarbeiten eingegangen werden. Im Anschluss wird ein Überblick über die Systemkomponenten gegeben, die im Rahmen dieser Arbeit entstanden bzw. ausgebaut wurden.

## 3.1 Grundlagen und Vorarbeiten

Das MAROCO System ist ein Forschungsrahmenwerk und verknüpft zur Umweltwahrnehmung und Auswertung verschiedene Komponenten. Auf die wesentlichen relevanten Komponenten soll im Folgenden eingegangen werden. Detailliert sind die Vorarbeiten in [Graf, 2010] ausgeführt. Darüber hinaus wurde ein Überblick über das System in [Puls et al., 2012c] gegeben, wobei bereits weitere, eigene Komponenten integriert wurden.

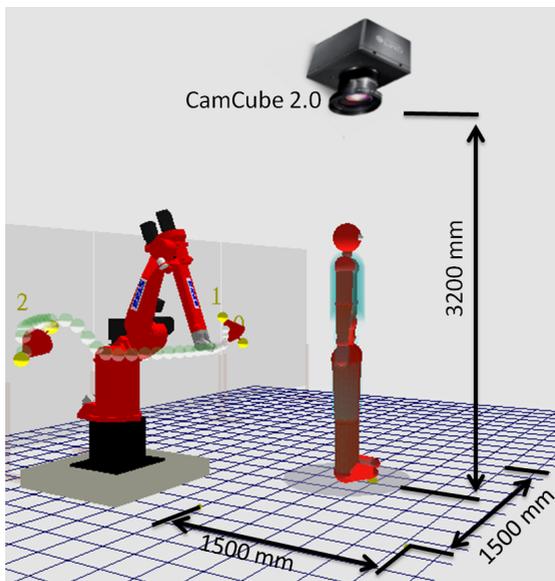
Als Einstieg werden die technischen Voraussetzungen des Systems erläutert, da diese ebenfalls für die vorliegende Arbeit die Grundlage darstellen.

---

<sup>1</sup>für Human-Robot-Cooperation

### 3.1.1 Technische Voraussetzungen des MAROCO Systems

Im Rahmen der Vorarbeiten durch [Graf, 2010] wurde ein Layout der Roboterzelle entworfen. Als zentraler Sensor kommt eine CamCube 2.0 von PMD Technologies [PMD Technologies, 2009] zum Einsatz, die an der Decke senkrecht über dem Arbeitsraum des Roboters angebracht ist. In Abb. 3.1 wird das Layout dargestellt. Der Sensor nutzt die Lichtlaufzeitmessung, um Pixeln Tiefenwerte zuordnen zu können. Die dabei entstehende Tiefenkarte des Arbeitsraumes wird als Grauwertbild kodiert und im MAROCO System weiterverarbeitet und ausgewertet.



**Abbildung 3.1:** Layout der Roboterzelle, wie sie im MAROCO System verwendet wird.

Der Vorteil dieser 3D Sensortechnologie ist, dass neben den Tiefeninformationen ebenfalls Amplitudenbilder zur Verfügung stehen. Amplitudenwerte sind ein Mittel, um die Remission der aktiven Beleuchtungseinheit

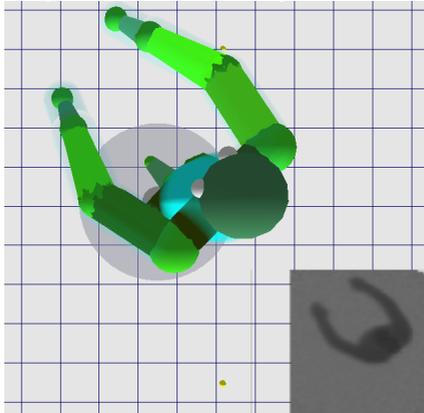
auszuwerten. Dabei wird die Remission durch Objekte und deren Oberflächeneigenschaften beeinflusst. Das erlaubt wiederum die Anpassung von Auswertelgorithmen, um bessere Robustheit zu erreichen. Zudem ist die gleichzeitige Verwendung mehrerer gleicher Sensoren bei gleichem Arbeitsraum möglich, da die modulierten Lichtimpulse andere Sensoren nur in sehr geringem Maße beeinträchtigen. Nachteilig ist jedoch, dass das verwendete Kamerasystem eine vergleichsweise geringe Bildauflösung von  $200 \times 200$  Pixeln aufweist.

Die Verwendung günstigerer Sensoren, wie bspw. Microsoft Kinect oder Asus Xtion, ist prinzipiell möglich, jedoch im MAROCO System nicht vorgesehen. Darüber hinaus könnten etwaige Kameraleistungen wie die Menschverfolgung nicht genutzt werden, da Kinect und Xtion eine Kameraausrichtung vor dem Benutzer voraussetzt. Dann können jedoch leichter Verdeckungen des Menschen auftreten, da zum einen der Roboter und zum anderen Werkgegenstände den Blick verdecken können. Die Befestigung der Kamera an der Decke hat ebenfalls den Nutzen, dass gewollte und ungewollte Manipulationen am Sensorsystem vermieden werden, wodurch ein konsistenter Sensoraufbau erreicht wird und Sicherheitsaspekte berücksichtigt werden.

### 3.1.2 Wesentliche Systemkomponenten

Die Tiefenbilder, die das Kamerasystem bereitstellt, werden in der Hauptsache genutzt, um einen Menschen im Roboterarbeitsraum wahrzunehmen. Die Tiefeninformationen werden mittels verschiedener Verfahren ausgewertet, so dass ein geometrisches Menschmodell rekonstruiert werden kann. Ausgehend von der Bestimmung der Position des Menschen werden die Orientierung des Kopfes und der Schultern bestimmt, in dem diese durch Ellipsen angepasst und die entsprechenden Hauptachsrichtungen verwendet werden.

Falls der Mensch die Arme hebt, sind diese ebenfalls für die Kamera erkennbar. Folglich werden die Schulter- und Ellenbogenwinkel zur vollständigen Rekonstruktion bestimmt. Aussagen über den Rumpf oder die Beine werden nicht getroffen. In Abb. 3.2 ist die Rekonstruktion der menschlichen Kinematik anhand des Tiefenbildes dargestellt.



**Abbildung 3.2:** Rekonstruktion der menschlichen Kinematik anhand von Tiefenbildern.

Die rekonstruierten Daten werden in ein geometrisches Umgebungsmodell aufgenommen. Dieses Modell enthält zudem a-priori modellierte statische Objekte und kinematische Modelle von bis zu drei Robotern, wobei hauptsächlich ein Roboter, der Reis RV6L, bei späteren Untersuchungen Verwendung findet. Auch für die vorliegende Arbeit gilt diese Feststellung, da dieser der einzige Roboter in der Zelle ist, der auch physisch eingesetzt und für Evaluationen verwendet werden kann.

Auf dem Umweltmodell basierend wird mittels eines Fuzzy Logic Systems eine Risikoschätzung über die Gefährdung des Menschen vorgenommen. Dazu werden Informationen über Blickwinkel, Distanz zwischen Mensch und Roboter und Bewegungsrichtung mit Regeln einer Wissensbasis verarbeitet. Dabei wird ein hohes Risiko angenommen, wenn der Mensch dem Roboter den Rücken zudreht, da somit der Mensch den Roboter nicht wahrnehmen und auf dessen Bewegungen reagieren kann. Ist der Mensch dem Roboter zugewandt, steht diesem jedoch zu nahe, wird auch ein hoher Risikowert ermittelt, um mögliche Kollisionen zu vermeiden. Diese Risikoschätzung ist demzufolge abhängig von der augenblicklichen menschlichen Pose, jedoch werden keine Informationen über mögliche Interaktion oder Kooperation einbezogen.

Das Umweltmodell wird darüber hinaus genutzt, um kürzeste Abstände zwischen Mensch und Roboter bestimmen zu können. Zusammen mit der Risikoschätzung kann eine kollisionsfreie Bahnplanung durchgeführt werden. Hierfür wurde in [Graf et al., 2009] ein erster Ansatz vorgestellt. Dieser Ansatz wird in der vorliegenden Arbeit vertieft und ausgebaut.

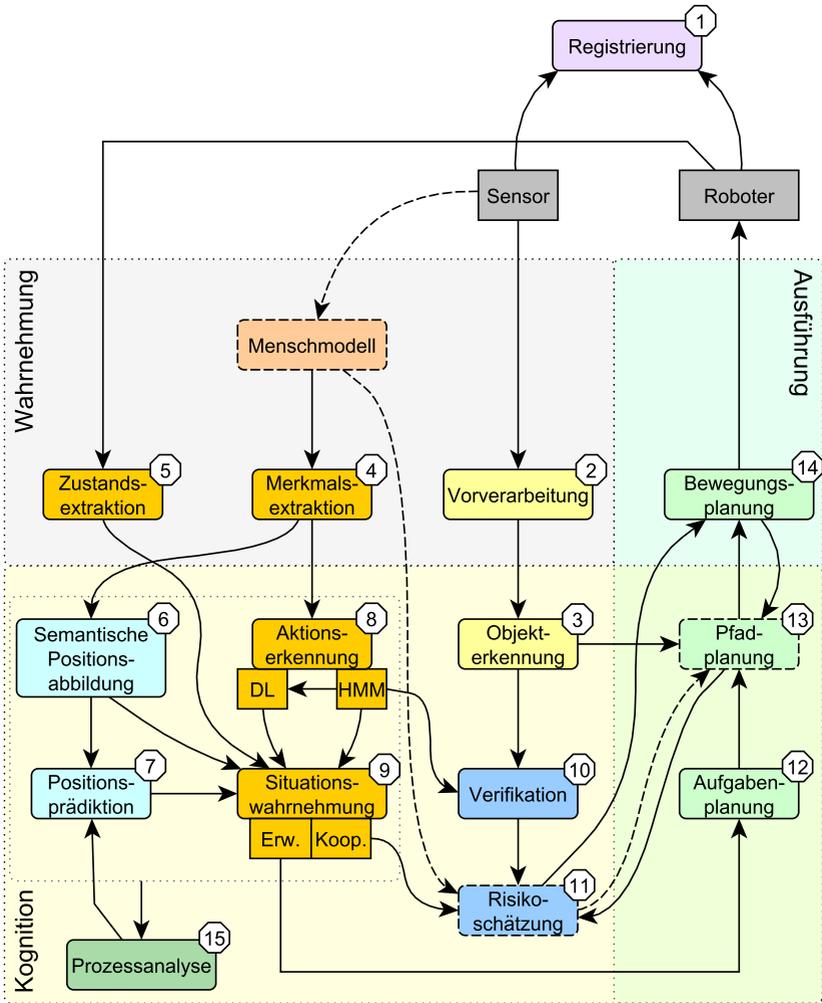
## 3.2 Eigene Systemkomponenten

Aufbauend auf den Vorarbeiten wurden im Rahmen dieser Arbeit Komponenten erweitert und vor allem neue Systemkomponenten entwickelt. Ein Überblick über die entsprechenden Systemkomponenten wird in diesem Abschnitt gegeben und ist in Abbildung 3.3 veranschaulicht. Zur Übersicht wird bei den Erläuterungen auf die jeweilige Komponente mit dem zugehörigen Index in der Abbildung 3.3 referenziert.

### 3.2.1 Erweiterte Komponenten

Zu den Weiterentwicklungen gehören die Risikoschätzung und die Pfadplanung. Die Risikoschätzung (11) wurde dahingehend erweitert, dass Eingaben aus der Situationswahrnehmung und der Verifikation verarbeitet werden können. Das führt zur Berücksichtigung der im Arbeitsraum vorliegenden Situation und betrachtet nicht ausschließlich die menschliche Pose. Um das untersuchte und robuste Systemverhalten beibehalten zu können, wurde ein mehrschichtiger Regelansatz entwickelt, der es erlaubt, das Risiko zu bestimmen und entsprechend dann zu beeinflussen, wenn es die Situation zulässt.

Die Pfadplanung (13) wurde ebenfalls weiterentwickelt, um enge Kooperation zwischen Mensch und Roboter zu ermöglichen. Da normativ Geschwindigkeitsanpassungen nötig sind, sobald sich ein Roboter in der Nähe eines Menschen bewegt, wurde gleichermaßen in der Planung eine risikoabhängige Geschwindigkeitsanpassung umgesetzt. Dadurch wird erreicht, dass zuvor unpassierbare Bereiche für die Durchfahrt geöffnet werden, wenn der Roboter sich ausreichend langsam bewegt. Speziell in der Annäherung des Roboters an den Menschen, bspw. bei Objektübergaben, ist diese Anpassung eine Notwendigkeit, da sonst der Roboter in sicherer



**Abbildung 3.3:** Übersicht der entwickelten Systemkomponenten. Gestrichelt markierte Komponenten und Übergänge enthalten übernommene Vorarbeiten. Die Generierung des Menschmodell ist ohne Änderungen enthalten. Die Risikoschätzung wurde grundlegend erweitert und angepasst. Gleiches gilt für die Pfadplanung, um enge Kooperation zwischen Mensch und Roboter bereits während der Planung zu ermöglichen.

Entfernung anhalten und auf die Abstandsvergrößerung zum Menschen warten würde. Eine ergonomische Zusammenarbeit mit dem Menschen würde entsprechend unterbunden werden. Darüber hinaus können auch Kollisionsmodelle von Szenenobjekten während der Planung berücksichtigt werden.

### 3.2.2 Neue Komponenten

Um Beobachtungen des Sensorsystems räumlich zum Roboter in Bezug setzen zu können, ist eine Registrierung (1) der Ortskoordinatensysteme notwendig. Da der Roboter fest verankert ist, ist die Registrierung des Kamerasystems bezüglich des Roboters sinnvoll. Entsprechend wurde im Rahmen dieser Arbeit eine automatische Registrierungsmethode entwickelt, die ein frei positioniertes Kamerasystem erlaubt, solange sich der Roboter-TCP im Sichtfeld befindet. Als Resultat liegt eine Transformationsmatrix zwischen den Koordinatensystemen vor.

Für die Erkennung von Szenenobjekten werden die Sensordaten zunächst aufbereitet und vorverarbeitet (2). Dabei werden Bildbereiche isoliert, die in der folgenden Objekterkennung (3) analysiert werden. Es wurde hierfür ein Verfahren entwickelt, das Neuronale Netze als assoziative Speicher einsetzt, um die geschichteten Tiefeninformationen der Bildbereiche zu klassifizieren.

Aus dem vorhandenen Menschmodell werden mittels Merkmalsextraktion (4) diejenigen Parameter ermittelt, die für folgende Komponenten als Eingabe dienen. Dazu zählen unter anderem die kartesische Position im Arbeitsraum, Armgelenkwinkel und Blickrichtung des Menschen. Die Positionsinformation wird auf relevante Orte im Arbeitsbereich semantisch abgebildet (6). Entsprechend liegt somit die Information vor, ob, und falls, an welchem Arbeitsbereich sich der Mensch aufhält. Die Bewegungstrajektorie wird weiterhin untersucht, um prädiktiv Positionsänderungen und deren Ziel zu schätzen (7).

Die ermittelten Menschparameter werden ebenfalls für die Aktions- und Handlungserkennung (8) verwendet. Dabei werden verschiedene Verfahren eingesetzt. Zum einen werden menschliche Aktionen mittels Hidden Markov Modellen klassifiziert. Die somit erkannten Aktionen werden zum anderen

in einem Inferenzsystem als logische Fakten genutzt, um Aussagen über komplexere Aktionsfolgen treffen zu können.

Zusätzlich zu diesen ermittelten Resultaten werden Informationen über den Roboter und Manipulator bestimmt (5). Die Menge an Information dient der Situationswahrnehmung (9) als Eingabe. Mittels logischer Inferenz werden basierend auf einer definierten ontologischen Wissensbasis Zusammenhänge erkannt und Aussagen über die vorliegende Situation getroffen. Daraus ergibt sich eine Einschätzung, ob es sich bspw. um eine kooperative Situation handelt. Gleichmaßen wird davon eine Erwartung an das Roboterverhalten abgeleitet.

Die ermittelten menschlichen Aktionen (8) werden mit den Informationen der Objekterkennung (3) kombiniert und erlauben eine Verifikation der Erkennungsergebnisse (10). Dabei werden Widersprüche aufgedeckt, sobald Aktionen Objektinteraktionen voraussetzen, ohne dass diese Objekte in der Szene erkannt werden. Im Falle von Widersprüchen wird die Risikoschätzung (11) entsprechend beeinflusst, so dass eine situationsabhängige Reduktion des Risikos vermieden wird. Andernfalls wird Information über die vorliegende Situation genutzt, um möglicherweise überhöhten Risikowerten entgegenzuwirken.

Die abgeleitete Erwartungshaltung an das Roboterverhalten wird direkt als Vorgabe bei der Aufgabenplanung (12) eingesetzt. Für diese Planung sind elementare Roboterfunktionen hinterlegt, die zu einem Ablaufplan zusammengestellt werden. Einzelne Abarbeitungsschritte werden dann an die Pfadplanung (13) übergeben, um den Plan schrittweise abzarbeiten. Ergeben sich zwischen den Schritten Änderungen für die Aufgabenplanung, wird eine erneute Planung durchgeführt.

Die Pfadplanung (13) ermittelt einen risikominimierten, kollisionsfreien Pfad für den Roboter und übergibt diesen der Bewegungsplanung (14). Diese ist dafür verantwortlich, dass zum einen mögliche Kollisionen mit dem Menschen frühzeitig erkannt werden, um eine Umplanung zu veranlassen. Zum anderen wird eine Trajektorie bestimmt, die geschwindigkeits- und ruckbegrenzt ist. Somit werden Vorgaben über maximale Geschwindigkeiten der Pfadplanung in diesem Schritt umgesetzt.

Somit ist in sich der Kreislauf des kognitiven Zyklus geschlossen. Der Mensch kann auf Roboterbewegungen reagieren und veranlasst durch sein

Handeln ein angepasstes Roboterverhalten. Um die Vorgänge im Arbeitsraum weiter untersuchen zu können, werden wesentliche Ergebnisse der Positions-, Aktions- und Situationsbestimmung in einem Prozessanalyse-schritt (15) weiterverarbeitet. Die sich daraus ergebenden Resultate bezüglich räumlicher Abläufe sind für die Positionsprädiktion (7) nutzbar. Weitere Ergebnisse können durch den Menschen auf Prozessfehler oder Optimierungspotentiale untersucht werden und geben einen Einblick in die Geschehnisse in der Arbeitszelle.



# 4 Umweltwahrnehmung im Arbeitsraum

Die Wahrnehmung der Umwelt des Roboters stellt eine wesentliche Aufgabe dar, um gezielte Interaktionen des Systems zu ermöglichen. Die Erkennung einzelner Objekte im Arbeitsraum stellt hierbei einen wichtigen Punkt dieser Arbeit dar. Um jedoch räumliche Bezüge zwischen Robotersystem und Gegenständen der Umwelt herstellen zu können, bedarf es einer Registrierung des Sensorsystems zum Robotersystem. Im Folgenden wird die entwickelte Methode zur automatisierten Registrierung vorgestellt, wonach der Lösungsansatz zur markerlosen Objekterkennung im Arbeitsraum erläutert wird.

## 4.1 Registrierung

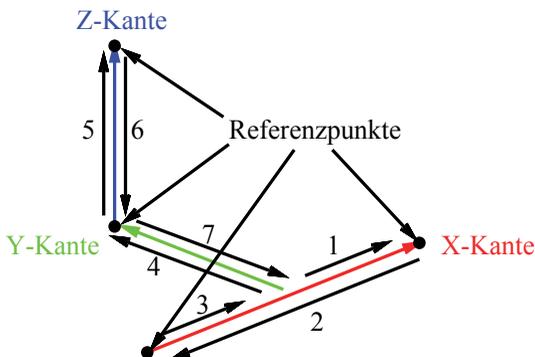
Wie eingangs erwähnt, dient die Registrierung dazu, die Ortskoordinatensysteme der verschiedenen Komponenten, d.h. Roboter und Sensor, in räumlichen Bezug zu setzen. Ziel ist es, eine entsprechende Transformationsvorschrift  $T$  zu ermitteln, die eine Abbildung eines Punktes von einem Quellkoordinatensystem  $K$  in ein Zielkoordinatensystem  $K'$  beschreibt. Die folgenden Abschnitte erläutern das Vorgehen, um dieses Ziel automatisiert zu erreichen.

### 4.1.1 Ablauf und Methodik

Ziel der automatisierten Registrierung ist es, dass keine andauernden manuellen Eingriffe und Eingaben einer Person während der Registrierung nötig sind. Es soll zu Beginn eine Robotertrajektorie festgelegt werden,

die es erlaubt, die benötigten Referenzpunkte aus einer Aufnahmesequenz direkt zu ermitteln. Diese Referenzpunkte werden dann zur Lösung der entsprechenden Gleichungssysteme herangezogen, wodurch die Transformation vom Kamerakoordinatensystem zum Roboterkoordinatensystem bestimmt wird.

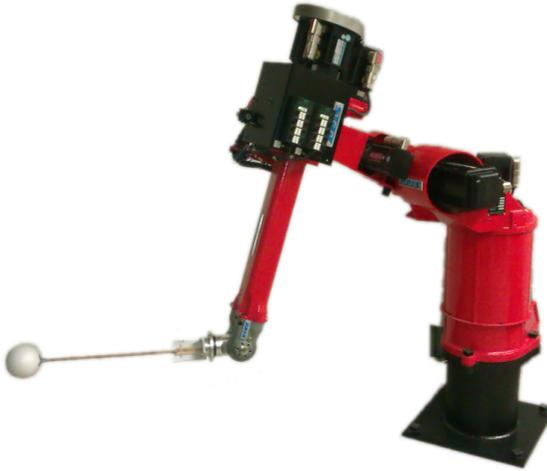
Im Rahmen dieser Arbeit wurde eine Robotertrajektorie gewählt, die sich an den Koordinatenachsen des Roboterkoordinatensystems orientiert. Die Trajektorie und die entsprechenden Referenzpunkte sind in Abb. 4.1 veranschaulicht. Die Referenzpunkte stellen somit charakteristische Punkte des Bewegungspfades dar. Damit die Referenzpunkte auch den entsprechenden Roboterkoordinatenachsen zugeordnet werden können, sind die Pfadabschnitte speziell zueinander positioniert. Daraus ergibt sich auch die positive Richtung der jeweiligen Achse. Die Y-Kante steht mittig auf der X-Kante. Die Z-Kante hat ihr Ende am Beginn der Y-Kante.



**Abbildung 4.1:** Robotertrajektorie und Anordnung der 4 Referenzpunkte zur Registrierung. Die Nummerierung der Pfeile beschreibt die Abfolge beim Abfahren der Trajektorie.

Damit die Registrierungsmethode mit verschiedenen Robotersystemen anwendbar sein soll, ist keine direkte Ansteuerung des Roboters durch die Registrierungssoftware vorgesehen. Entsprechend ist es notwendig, dass diese Trajektorie beim ersten Registrierungsvorgang dem Roboter per Programmierhandgerät geteacht wird. Später kann das Roboterprogramm für mögliche weitere Registrierungen eingesetzt werden. Folglich ist jedoch

keine direkte Information über Roboterzustand und -position während der Registrierung vorhanden. Um diesem Umstand zu begegnen, wird ein Registrierkörper am Roboter-TCP angebracht, der sich aus den Sensordaten extrahieren lässt. Über die Position des Registrierkörpers kann somit auf die Position des Roboter-TCPs gefolgert werden. In Abb. 4.2 ist der verwendete Roboter *Reis RV6L* mit einem Registrierkörper dargestellt.



**Abbildung 4.2:** Roboter Reis RV6L mit Registrierkörper.

Der Aufbau der Software ist modular gestaltet, damit einzelne Komponenten für andere Einsatzzwecke und Sensoren adaptiert werden können. Die einzelnen Module haben die folgenden Aufgaben:

**Dateneingabe** Die Eingabedaten liest dieses Modul z.B. aus einer Kamera oder Dateien. Als Ausgabe wird eine Punktwolke erzeugt. Jeder Punkt der Punktwolke kann neben seiner Raumposition noch weitere Attribute haben.

**Datenaufbereitungsfilter** Die Punktwolke kann durch verschiedene Filter aufbereitet werden, um etwa Rauschpunkte zu entfernen. Ziel in diesem Schritt ist es, die Datenqualität zu verbessern.

**Segmentierung** Als Eingabe erhält dieses Modul die gefilterte Punktwolke. Dieser Filter unterteilt die Punktwolke in charakteristische Segmente. Die dabei entstehenden Segmente sind die Ausgabe dieses Moduls.

**Objektidentifizierung** Aus den Segmenten versucht die Objektidentifizierung anhand spezifischer Merkmale ein Referenzobjekt zu identifizieren. Als Ausgabe liefert dieses Modul alle erkannten Möglichkeiten für den gesuchten Registrierkörper.

**Formerkennung** Dieses Modul erhält als Eingabe mehrere Bewegungsbahnen von Objekten, unter denen der Registrierkörper gesucht werden soll. Die Bewegungsbahn oder Form der Eingaben werden analysiert, um das finale Referenzobjekt zu bestimmen. Dieses wird als Ausgabe an das nächste Modul geliefert.

**Registrierung** In diesem Modul wird die konkrete Registrierung vom Kamerakoordinatensystem zum Roboterkoordinatensystem durchgeführt. Bei Eingabe des Referenzobjektes werden die notwendigen Merkmale extrahiert, um die Transformationsmatrix zu bestimmen.

### 4.1.2 Filterung, Segmentierung und Bewegungsformerkennung

Wie bereits in Abschnitt 3.1.1 erwähnt, liefert das verwendete Kamerasystem mehr als reine Tiefeninformationen zu den Bildpunkten. Diese zusätzlichen Informationen können in einem ersten Schritt verwendet werden, um die erhaltene Punktwolke zu filtern und verrauschte Daten zu entfernen. Anhand der gefilterten Daten wird die Punktwolke segmentiert, wonach die Bewegungen der in Frage kommenden Segmente analysiert und mit der definierten Referenzbewegung verglichen werden.

#### Filterung

Die Amplitudeninformation kann als Wert für die Signalstärke interpretiert werden. Eine höhere Amplitude entspricht dabei einem besseren Signal und damit einem genaueren Pixel. Ein Amplitudenfilter sortiert entsprechend Punkte aus, deren Amplitude außerhalb minimaler und maximaler

Schwellwerte liegt. In Abbildung 4.3 kann man links ungefilterte mit rechts gefilterten Daten vergleichen.



**Abbildung 4.3:** Sensordaten, links ungefiltert, rechts mit dem Amplitudenfilter gefiltert.

Die Kamera prüft ebenfalls bereits intern die Qualität der einzelnen Pixel. Dabei markiert sie Pixel mit einem Status. Es stehen vier Möglichkeiten zur Auswahl: *ungültig*, *gesättigt*, *geringes Signal* und *inkonsistent*. Ungültige Pixel sollten generell nicht verwendet werden, weil sie keine verlässlichen Distanzwerte bieten. Gesättigte Pixel sind überbelichtet und bieten ebenfalls keine verlässliche Datenqualität. Pixel mit geringem Signal sind gleichermaßen über die Amplitudeninformation erkennbar. Inkonsistente Pixel treten dann auf, wenn zu schnelle Bewegungen in der Szene die Daten der Pixel intern inkonsistent werden lassen. Der Status-Filter sortiert alle Pixel aus, die eine dieser Ausprägungen aufzeigen.

Um darüber hinaus weitere Ausreißer zu eliminieren, wurde ein Filter entworfen, der anhand einer Nachbarschaftsbetrachtung der Tiefeninformation Ausreißer erkennt. Dazu wird zu jedem Punkt die gemittelte Abweichung

der Tiefe zu seinen  $n$  Nachbarpunkten bestimmt. Über diese mittleren Abweichungen wird die Standardabweichung berechnet. Jeder Punkt, der eine größere mittlere Abweichung als die Standardabweichung besitzt, wird aussortiert. Der Filter hat den Vorteil, dass kein numerischer Grenzwert angegeben werden muss. Es können sehr verrauschte Daten gefiltert werden, ohne alle Punkte zu eliminieren.

Falls die Abmessungen des Arbeitsraumes und der ungefähre Abstand der Kamera zum Roboter bekannt sind, kann darüber hinaus die Punktwolke anhand minimaler und maximaler Tiefenwerte gefiltert werden. Dadurch lässt sich die Anzahl der zu verarbeitenden Punkte verringern und weitere Berechnungen beschleunigen.

### **Segmentierung**

Der DBSCAN-Algorithmus [Ester et al., 1996] wird verwendet, um die Punktwolke zu clustern und somit Ballungen in der Punktwolke zu extrahieren. Zur Datenverwaltung und Nachbarschaftssuche wird mit einem Binary Space Partitioning Tree (BSPT) [de Berg et al., 2008] gearbeitet, da diese Datenstruktur eine effiziente Verarbeitung ermöglicht.

### **Bewegungsformerkennung**

Die segmentierten Cluster werden anhand der Größe ihrer jeweiligen *Bounding-Box* gefiltert, indem zu kleine bzw. zu große Cluster nicht weiter betrachtet werden. Das zu identifizierende Objekt, im Testaufbau ein Ball an der Spitze des Roboterarms (siehe Abb. 4.2), hat eine Bounding-Box, die aufgrund von Rauschen nur zwischen gewissen Werten variiert. Anhand dieser Grenzwerte werden alle interessanten Cluster extrahiert. Die Filtrierung anhand der Bounding-Box wird jedoch nicht in der Tiefenrichtung beschränkt, weil in dieser Richtung mehr Rauschen zu beobachten ist.

Dennoch kann auf diese Weise der Registrierkörper nicht eindeutig identifiziert werden, da weitere, ähnlich große Objekte im Arbeitsraum zu finden sein können. Es gilt daher, in dieser Menge an möglichen Aufenthaltsorten des Objektes genau den richtigen zu finden.

Die Menge der möglichen Aufenthaltsorte des gesuchten Objektes ergibt sich aus einer fehlerbehafteten und dynamischen Beobachtung. Für die gleichzeitige Betrachtung verschiedener Möglichkeiten sind Partikelfilter [Doucet and Johansen, 2008] besonders geeignet. Auf das Problem angewendet wird der Bewegungspfad gesucht, so dass die Partikel die beobachteten Objektpfade beschreiben. Für die Bewertung eines Partikels wird der Abstand aufeinanderfolgender Markierungen beachtet. Je weiter sie voneinander entfernt sind, desto unwahrscheinlicher wird der Partikel. Entsprechend werden Partikel mit zu geringer Wahrscheinlichkeit verworfen. Über den Verlauf der Bildfolgenauswertung reduzieren sich somit die möglichen Partikel, so dass nur wenige Pfade auf Ähnlichkeit mit der definierten Trajektorie hin untersucht werden müssen.

Für den Abgleich der Partikel mit der Trajektorie wird der Objektpfad je Partikel in Kanten segmentiert. Dazu sind die vorgegebenen Kantenlängen der Trajektorie bekannt, wodurch die Kanten direkt aus den Punkten, die den jeweiligen Objektpfad beschreiben, bestimmt werden können. Daraufhin werden die Beziehungen der Kanten eines Pfades zueinander untersucht, d.h. es wird deren Orthogonalität und Positionierung überprüft.

Entspricht ein Partikel der definierten Trajektorie, werden die Referenzpunkte direkt ermittelt und für die Bestimmung der Transformationsmatrix  $T$  (siehe Glg. 4.1 und 4.2) genutzt.

## Transformationsbestimmung

Seien vier korrespondierende Punkte  $\vec{p}_1, \vec{p}_2, \vec{p}_3, \vec{p}_4$  aus Koordinatensystem  $K$  und  $\vec{p}'_1, \vec{p}'_2, \vec{p}'_3, \vec{p}'_4$  aus  $K'$ . So lässt sich ein lineares Gleichungssystem aufstellen. Die Variablen, die es dabei zu bestimmen gilt, sind die Werte der Rotationsmatrix  $R$  und des Translationsvektors  $\vec{t}$ , die zusammen die Transformation  $T$  von Koordinatensystem  $K$  zu  $K'$  beschreiben:

$$T = \begin{pmatrix} a_1 & a_2 & a_3 & t_1 \\ b_1 & b_2 & b_3 & t_2 \\ c_1 & c_2 & c_3 & t_3 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \quad (4.1)$$

Für  $a_1, a_2, a_3$  und  $t_1$  ergibt sich folgendes Gleichungssystem:

$$\begin{pmatrix} p_{1_1} & p_{1_2} & p_{1_3} & 1 \\ p_{2_1} & p_{2_2} & p_{2_3} & 1 \\ p_{3_1} & p_{3_2} & p_{3_3} & 1 \\ p_{4_1} & p_{4_2} & p_{4_3} & 1 \end{pmatrix} \cdot \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ t_1 \end{pmatrix} = \begin{pmatrix} p'_{1_1} \\ p'_{2_1} \\ p'_{3_1} \\ p'_{4_1} \end{pmatrix}. \quad (4.2)$$

Die Gleichungen für die Parameter  $b_i, c_j$  und  $t_i$  ergeben sich analog. Durch die Bestimmung der Referenzpunkte und der Definition der Trajektorie sind die Punkte  $\vec{p}_i$  und  $\vec{p}'_j$  bekannt und können zur Lösung des Gleichungssystems eingesetzt werden.

### 4.1.3 Evaluation und Diskussion

Die Registrierungsmethode wurde in der Roboterzelle durchgeführt, die Ergebnisse aufgezeichnet und zur Evaluation analysiert. Im Folgenden wird auf die Experimentierumgebung eingegangen und die Ergebnisse vorgestellt und diskutiert.

#### Experimentierumgebung

Als Registrierkörper wurde eine Styroporkugel an der Spitze eines Stabes befestigt und direkt am Werkzeugflansch montiert (siehe Abb. 4.2). Der Durchmesser des Registrierkörpers betrug  $10\text{cm}$ . Der Stab wurde mit gering reflektierendem Material beschichtet, so dass dieser durch Anpassung der Amplitudenfilterung für die Datenweiterverarbeitung nicht berücksichtigt werden musste.

Für die Robotertrajektorie wurden Kantenlängen von  $40\text{ cm}$  gewählt. Beim Verfahren des Roboters wurden an den Referenzpunkten kurze Bewegungspausen eingefügt, damit diese bei möglicherweise geringer Bildrate nicht übersehen werden. Die Positionierung der Trajektorie im Arbeitsraum kann frei gewählt werden und wurde zu Evaluationszwecken parametrisiert.

## Ergebnisse

Die Bestimmung der Transformationsmatrix ist von der Bestimmungsqualität der Referenzpunkte abhängig. Entsprechend sind Aussagen über die Güte dieser Punkte von wesentlichem Interesse.

Bei der Rekonstruktion der Trajektorie anhand der Sensordaten ist es ideal, wenn die einzelnen Pfadkanten keine Lücken an ihren Verbindungsstellen aufweisen. Es wurde angenommen, dass, falls Lücken auftreten, es zur fehlerhaften Rekonstruktion der Kantenlängen und entsprechend zu einer fehlerhaften Translationsschätzung kommt. Des Weiteren müssen die Kanten jeweils orthogonal zueinander sein. Jede Winkelabweichung von der Orthogonalität kann sich in Fehlern bei der Translation und Rotation auswirken. Die Ergebnisse zur Bestimmung dieser Einflussfaktoren sind in Tabelle 4.1 aufgeführt.

Das verwendete Kamerasystem kann ebenfalls an einem Stativ befestigt werden, wodurch die freie Positionierung um den Roboterarbeitsraum herum möglich ist. Der Aufbau ist in Abb. 4.4 dargestellt. Dadurch können verschiedene sukzessiv aufgenommene und registrierte Punktwolken fusioniert werden. Das Ergebnis ist auf der rechten Seite in Abb. 4.4 visualisiert.

## Diskussion

Es ist anhand der Ergebnisse aus Tabelle 4.1 zu erkennen, dass mit zunehmender Entfernung zwischen Registriertrajektorie und Kamera die Abweichung von der Orthogonalität anwächst. Jedoch ist auch erkennbar, dass dieser Fehler nicht direkt mit der Offsetbestimmung der Registrierung korreliert, da der Offset bei Verschiebung entlang der  $Z$ -Achse stets geringer ausfällt, als es durch den Roboteroffset vorgegeben ist. Bezüglich der Verschiebung längs der  $X$ -Achse kann eine Korrelation mit der Orthogonalitätsabweichung nicht festgestellt werden.

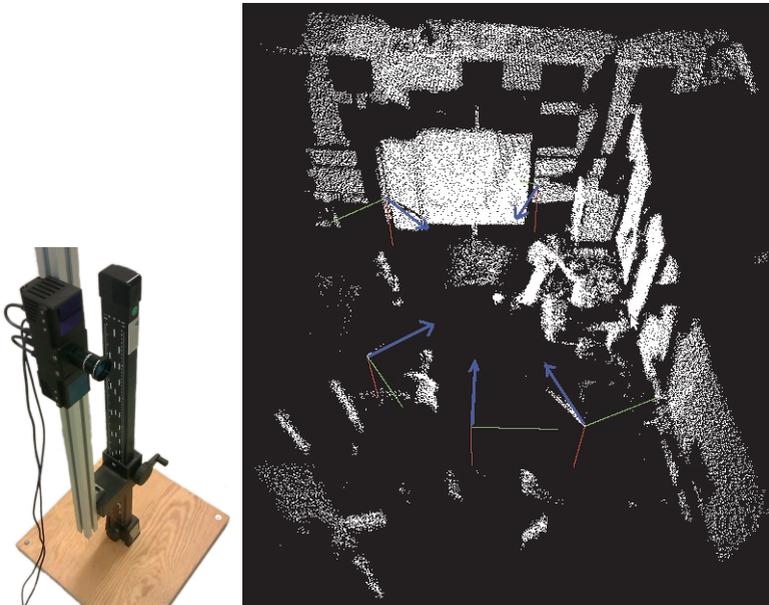
Bei der Betrachtung der Lückengröße fällt auf, dass die Lücke zwischen  $X$  und  $Y$ -Kante deutlicher zu Tage treten als zwischen  $Y$  und  $Z$ -Kante. Diese Beobachtung kann mit dem Aufbau der Trajektorie zusammenhängen, da sich die ermittelte  $X$ -Kante anhand von Rauschen vom Startpunkt

**Tabelle 4.1:** Registrierung mit 10 cm durchmessenden Registrierkörper und unterschiedlichen Startpositionen der Roboterbewegungen.

Lückengröße (X-Y/Y-Z)	Abweichung von Orthogonalität	Offset der Registrierung
<i>Roboteroffset -40 cm in Z-Achse</i>		
(0,36 cm / 2,03 cm)	1,60°	-39,51 cm
<i>Roboteroffset -20 cm in Z-Achse</i>		
(2,73 cm / 0,61 cm)	2,87°	-19,99 cm
<i>Roboteroffset 0 cm</i>		
(1,83 cm / 0,15 cm)	2,12°	0 cm (Referenzposition)
<i>Roboteroffset +20 cm in Z-Achse</i>		
(2,44 cm / 0,52 cm)	2,88°	+19,99 cm
<i>Roboteroffset +40 cm in Z-Achse</i>		
(1,10 cm / 1,48 cm)	2,91°	+39,39 cm
<i>Roboteroffset -20 cm in X-Achse</i>		
(0,91 cm / 0,44 cm)	3,09°	-20,51 cm
<i>Roboteroffset 0 cm</i>		
(0,87 cm / 0,52 cm)	2,53°	0 cm (Referenzposition)
<i>Roboteroffset +20 cm in X-Achse</i>		
(1,70 cm / 0,39 cm)	1,17°	+20,38 cm

der Y-Kante entfernen kann. Die End- und Startpunkte der Y-Kante bzw. Z-Kante sind durch Rauschen entlang der jeweiligen Kante weniger beeinflusst. Eine Korrelation zwischen Lückengröße und Offsetfehler kann jedoch nicht erkannt werden, da bei größeren Lücken teilweise kleinere Fehler auftreten.

Insgesamt liegt der Fehler der Offsetbestimmung der Registrierung im Vergleich zur Referenzposition bei deutlich unter einem Zentimeter. Dieser Fehler ist ausreichend gering, um zuverlässig Abstandsbestimmungen zwischen Roboter, rekonstruiertem Menschmodell und Szeneobjekten zu gewährleisten.



**Abbildung 4.4:** Links: Anbringung der Kamera CamCube 2 von PMD-Technologies an einem Stativ. Rechts: Fusionsergebnis verschiedener, registrierter Punktwolken. Die Kamerakoordinatensysteme sind eingezeichnet, wobei die blauen Pfeile die jeweilige optische Achse markieren.

## 4.2 Erkennung von Szeneobjekten

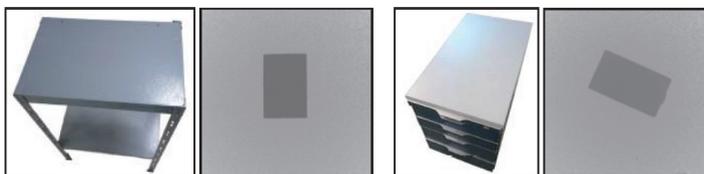
Das Wissen über Gegenstände im Arbeitsraum des Roboters birgt im Wesentlichen zwei Potentiale. Zum einen können Kollisionen bei der Roboterbewegung vermieden werden, da die Objekte bei der Pfadplanung berücksichtigt werden können. Zum anderen können Gegenstände menschliches Handeln beeinflussen, ermöglichen oder örtlich festlegen. Im Folgenden soll die entwickelte Methode zur markenlosen Objekterkennung im Arbeitsraum vorgestellt und erörtert werden. Die wesentlichen Aspekte sind auch in [Puls et al., 2012d] beschrieben.

### 4.2.1 Problemstellung

Der Arbeitsraum des Roboters wird durch ein 3D Kamerasystem, der CamCube2 von PMDTechnologies, überwacht. Die Kamera bietet eine Auflösung von  $204 \times 204$  Pixeln. Wie bereits beim Abschnitt zur Registrierung erwähnt, sind die Tiefeninformationen rauschbehaftet (siehe Abb. 4.3).

Auf Grund der Sensorpositionierung an der Zellendecke, senkrecht über dem freien Arbeitsraum (siehe Abschnitt 3.1.1), sind darüber hinaus für die weitere Auswertung teilweise nur wenige Objektmerkmale ersichtlich. Tische werden bspw. nur als flache Rechtecke in der Szene wahrgenommen, wodurch die Unterscheidbarkeit stark beeinflusst wird (siehe Abb. 4.5).

Ziel ist es entsprechend, eine Methode zu entwickeln, die mit rauschbehafteten Daten umgehen kann und mit wenigen Objektmerkmalen auskommt. Dabei soll auch untersucht werden, wie sehr ähnliche Objekte unterschieden werden können.



**Abbildung 4.5:** Links: Beistelltisch. Rechts: Schubladentisch.

### 4.2.2 Stand der Forschung

Bildverarbeitung und speziell die Objekterkennung beschäftigt Forscher und Entwickler seit vielen Jahrzehnten. Viele Methoden und Verfahren wurden entwickelt, die sich in ihren Anwendungsfällen bewährt haben. Beispielsweise wurde durch Hu die Verwendung von Bildmomenten für die Objekterkennung eingeführt [Hu, 1962]. Moderne Verfahren wie SIFT [Lowe, 1999] oder SURF [Bay et al., 2006] arbeiten auf Objektmerkmalen, wie Ecken oder Kanten, und ermöglichen Objekterkennung, die invariant

gegenüber Veränderungen der Objektlage und Skalierung ist. Es werden jedoch Grauwert- oder Farbbilder der Objekte benötigt. Auf Grund der in dieser Arbeit verwendeten Time-of-Flight-Tiefenbildsensoren sind solche Objektbilder nicht verfügbar. Darüber hinaus können die Daten ein beträchtliches Rauschverhalten aufweisen – speziell am Bildrand und bei schlechter Reflektanz der Objektoberfläche.

Künstliche neuronale Netze (KNN) werden ebenfalls seit Jahrzehnten für die Mustererkennung eingesetzt [Carpenter, 1989]. Seit der Einführung von KNNs als assoziative Speicher durch Hopfield [Hopfield, 1982] werden diese vielfach für Objekterkennung und Bildwiederherstellung genutzt, da sie besonders robust gegenüber verrauschter Daten sind.

In einer Arbeit von Nasrabadi und Li werden Hopfield Netze für die Erkennung von sich teilweise überdeckenden Objekten eingesetzt [Nasrabadi and Li, 1991]. Dafür werden zu Beginn anhand der Eingabebilder Modellgraphen durch Polygonannäherungen generiert. Als Knoten im Graphen werden die Eckpunkte der Polygone genutzt. Jeder Knoten wird darüber hinaus mit Informationen über Winkel und Distanzen zu den anderen Knoten ergänzt. Während des Lernens wird für jedes Objekt ein Modellgraph erstellt und zu einem globalen Modellgraphen hinzugefügt. Die Wiedererkennung von Objekten geht analog vor, wobei aus den Modellgraphen ein Szenengraph erstellt wird. Um den Szenengraphen auf den globalen Modellgraphen abzubilden und entsprechend die Objektzuordnung zu rekonstruieren, werden Hopfield Netze für die Minimierung einer definierten Energiefunktion genutzt.

Die Arbeit von Nasrabadi und Li wird durch Young et al. weiterentwickelt [Young et al., 1997]. Es wird dabei für die Objekterkennung ein mehrschichtiges Hopfield Netz eingesetzt. Jede Schicht kodiert Objektmerkmale spezifisch für eine Auflösung des Eingabebildes. Die Verbindungen der Schichten werden gewichtet, wodurch eine Glättung der Energiefunktion erreicht wurde. Lokale Minima konnten auf diese Weise robuster vermieden werden.

Die Verwendung mehrschichtiger neuronaler Netze für das Speichern und Wiederherstellen von Grauwertbildern wird in der Arbeit von [Constantini et al., 2006] vorgestellt. Das neuronale Netz basiert auf Hopfield Netzen und besitzt einen Verbindungsaufbau zwischen den Neuronen angelehnt an zelluläre neuronale Netze [Chua and Roska, 2002]. Entsprechend sind nicht

alle Neuronen einer Schicht miteinander verbunden. Stattdessen wird für jedes Neuron eine Nachbarschaft definiert, wobei sich die Nachbarschaft auch auf Neuronen anderer Schichten bezieht. Neue Bilder können mit Hilfe einer iterativen Regel gelernt werden. Analog zu Hopfield Netzen wird für die Wiederherstellung der Bilder die Energieminimierung des Netzes genutzt. Um Grauwertbilder verarbeiten zu können, wird jeder Grauwert durch die Gray-Kodierung dargestellt, wobei jede Schicht einem Bit der Kodierung zugeordnet wird. Das System ist in der Lage, Bilder der Größe  $200 \times 200$  pixel und mit 256 Grauwerten zu lernen.

In der Arbeit von [Shima et al., 2011] wird ein mehrschichtiges Hopfield Netz verwendet, um Bilder wiederherzustellen. Dazu wird das Bild in Zellen aufgeteilt, welche aus Quadraten von Pixeln bestehen. Jede Schicht des Netzes korrespondiert zu einer Zelle. Entsprechend wird die Wiederherstellung lokal berechnet. Die Ausgabe des Hopfield Netzes wird genutzt, um die Distanz zum gelernten Bild zu bestimmen. Das Ergebnis entspricht somit dem Bild, das die minimal Distanz zur Netzausgabe hat.

### 4.2.3 Objekterkennung

Wie bereits in Abb. 4.5 dargestellt, können die Sensordaten als grauwertkodierte Tiefenkarten repräsentiert werden. Dunkle Grautöne bilden Bildbereiche nahe der Kamera ab, hellere Grautöne zeigen Bildbereiche, die weiter vom Sensor entfernt sind. Anhand der Untersuchungen durch [Constantini et al., 2006] können Grauwertbilder mittels mehrschichtiger neuronaler Netze als assoziative Speicher gelernt und wiederhergestellt werden. Entsprechend dieser Möglichkeiten wurde im Rahmen dieser Arbeit eine Methode zur Objekterkennung entwickelt, die auf der Verwendung neuronaler Netze als assoziative Speicher basiert.

#### 4.2.3.1 Das neuronale Netz

Neuronale Netze können verschiedene Topologien aufweisen. Viele Netze besitzen Eingabeneuronen, die die Eingabeschicht bilden und über die Daten in das Netz eingegeben werden. Analog lassen sich Ausgabeneuronen bestimmen, die die Ausgabeschicht bilden. Zwischen diesen Schichten

können sich beliebig viele weitere Schichten befinden, wobei die Struktur der Neuronenverknüpfungen unterschiedlich aufgebaut sein können. Beispielsweise gibt es vorwärtsgerichtete Netze, auch *feedforward Netze* genannt, bei denen Neuronen einer Schicht ausschließlich mit Neuronen der nächsten Schicht verbunden sind.

**Netzwerkarchitektur** Hopfield entwickelte und untersuchte Netze bei denen die Eingabeneuronen ebenfalls die Ausgabeneuronen darstellen [Hopfield, 1982]. Dazwischen gibt es keine weiteren Schichten, sondern alle Neuronen sind miteinander direkt verbunden. Die Leistung des Netzes wird somit durch die verstärkenden und abschwächenden Verbindungen zwischen den einzelnen Neuronen erzielt. Das Ergebnis wird durch wiederholte Aktivierung des Netzes ermittelt bis ein Fixpunkt im Netz erreicht wird.

In der Arbeit von [Constantini et al., 2006] wurden mehrere Hopfield Netze in verschiedenen Schichten zusammen verknüpft, wobei die Verknüpfungsstruktur zwischen den Neuronen abgeändert wurde. Es wurde dabei auf die vollständige Verknüpfung der Netze verzichtet. Stattdessen wurde eine Nachbarschaftsbeziehung der Neuronen definiert, die sich an der Betrachtung von zellulären neuronalen Netzen (siehe [Chua and Roska, 2002]) orientiert.

Das neuronal Netz besteht aus  $L$  Schichten, welche jeweils  $M \times N$  Neuronen haben. Für jedes Neuron  $(i, j, p)$  wird die Nachbarschaft  $N_{r,s}$  folgendermaßen definiert:

$$\begin{aligned}
 N_{r,s}(i, j, p) = \{ & (k, l, q) : |k - i| \leq r \vee |k - i| \geq M - r \\
 & |l - j| \leq r \vee |l - j| \geq N - r \\
 & |q - p| \leq s \vee |q - p| \geq L - s \},
 \end{aligned} \tag{4.3}$$

wobei  $r$  und  $s$  natürliche Zahlen sind, mit  $r \geq s$ . Es gilt weiterhin  $i = 1, \dots, M, j = 1, \dots, N$  und  $p = 1, \dots, L$ . Folglich stellen  $r$  und  $s$  jeweils den Nachbarschaftsradius für Interschicht- und respektive Intraschichtverbindungen dar. Jede Verbindung hat ein spezifisches Gewicht, welches während des Lernens angepasst wird und welches eine verstärkende bzw. abschwächende Wirkung einer Verbindung erzielt.

Angelehnt an Hopfield-Netze gilt für die Gewichte reflexiver Kanten  $w_{ijp,ijp} = 0$ . Entsprechend ergibt sich die effektive Nachbarschaft  $\underline{N}_{r,s}$  aus  $N_{r,s}$  ohne das korrespondierende Neuron selbst.

**Musterlernen** Das Training des Netzes wird durch die Anpassung der Kantengewichte anhand von  $Q$  Mustern erreicht. Somit wird ein Attraktor im Netz erzeugt. Die Aktualisierung der Gewichte hängt von den gegebenen Mustern ab und basiert auf der Arbeit von [Constantini et al., 2006]. Ein Muster hat die gleichen Dimensionen wie das neuronale Netz und jedes Element des Musters muss den binomialen Werten  $\{-1, 1\}$  entsprechen.

Für ein Kantengewicht  $w_{ijp,klq}$  zum Zeitpunkt  $(t + 1)$  werden die Zustände  $y_{ijp}$  und  $y_{klq}$  aller Muster  $m = 1, \dots, Q$  multipliziert und aufsummiert. Zusätzlich wird die Summe mit der *Penalty-Funktion*  $P$  multipliziert und das alte Kantengewicht  $w_{ijp,klq}(t)$  addiert. Zu Anfang wird  $w_{ijp,klq}(0) = 0$  angenommen. Die Berechnung ist gegeben durch:

$$w_{ijp,klq}(t + 1) = w_{ijp,klq}(t) + \eta \sum_{m=1}^Q y_{ijp}^{(m)} y_{klq}^{(m)} \times P(\Delta_{ijp}^{(m)}(t)) \quad (4.4)$$

wobei  $(k, l, q) \in \underline{N}_{r,s}$  und  $\eta > 0$  die Lernrate ist. Weiterhin gilt:

$$\Delta_{ijp}^{(m)}(t) = \sum_{k,l,q \in \underline{N}_{r,s}(i,j,p)} w_{ijp,klq}(t) y_{ijp}^{(m)} y_{klq}^{(m)} - \delta, \quad (4.5)$$

und

$$P(x) = \begin{cases} 0, & x \geq 0, \\ 1, & x < 0. \end{cases} \quad (4.6)$$

Dabei ist  $\delta$  eine Stabilitätsschranke und  $m$  liegt im Intervall  $[1, \dots, Q]$ .

Das Einlernen des Netzes erfolgt überwacht. Dies bedeutet, die Anzahl der Klassen und die Klassenzugehörigkeit der zu lernenden Muster sind zum Zeitpunkt des Lernens bekannt. Es können keine einzelnen Muster im Nachhinein zu einer Klasse hinzugelernt werden, jedoch können die einzelnen Klassen jederzeit unabhängig voneinander gelernt werden.

**Mustererkennung** Bei der Erkennung eines Musters wird dieses als Eingabe an das Netz angelegt. Dann erfolgt eine Aktivierung des Netzes nach dem BSB<sup>1</sup>-Modell (siehe [Anderson et al., 1977]):

$$x_{ijp}(t+1) = f \left[ x_{ijp}(t) + \sum_{(k,l,q) \in N_{r,s(i,j,p)}} w_{ijp,klq} \cdot x_{lq}(t) \right], \quad (4.7)$$

wobei  $f$  die Schwellwertfunktion

$$f(y) = \begin{cases} -1, & y < -1, \\ y, & -1 \leq y \leq 1, \\ 1, & y > 1. \end{cases} \quad (4.8)$$

ist. Die Aktivierung wird so lange durchgeführt, bis der Wert der Energie des Netzes unter einem fest gewähltem Schwellwert liegt. Der finale Zustand des Netzes stellt das erkannte Muster dar.

#### 4.2.3.2 Objekterkennung

Objekterkennung ist die Abbildung eines Objektes auf eine bekannte Klasse. Das Lernen verschiedener Klassen durch ein einzelnes neuronales Netz erzeugt multiple Attraktoren in diesem Netz. Somit ist, auf Grund der ungerichteten Konvergenz des Netzes, eine solche direkte Abbildung nicht möglich.

Anhand der Erfahrungen bei der Aktionserkennung mittels Hidden Markov Modellen [Puls and Wörn, 2012] wurde ein analoges Vorgehen für den neuronalen Klassifikator entworfen. Bei der Aktionserkennung wird jede Aktion durch ein separates Modell beschrieben. Während der Klassifikation wird das Modell gewählt, das die höchste Wahrscheinlichkeit zur Beobachtungssequenz generiert. Dieses Vorgehen, separate Modelle für jede Aktion bzw. für jedes Objekt zu nutzen, wurde somit für den neuronalen Klassifikator umgesetzt.

Damit Objekte erkannt werden können, die frei im Arbeitsraum positioniert sind, muss die Methode positions- und orientierungsinvariant sein. Darüber

---

<sup>1</sup>BSB: Brain-State-in-a-Box

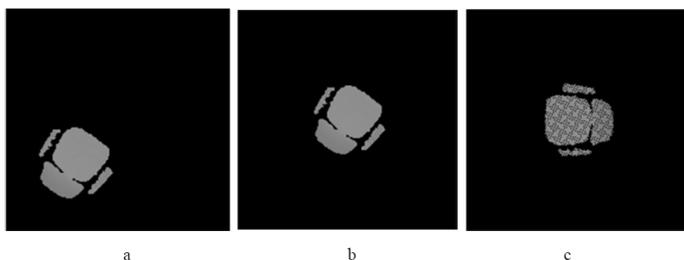
hinaus müssen Objekte isoliert und in ein entsprechendes Muster überführt werden, damit sie als Eingabe in den Klassifikator dienen können. Im Folgenden ist die Abfolge der Verarbeitungsschritte aufgeführt.

**Segmentierung** Der erste Schritt beinhaltet die Isolation eines Objekts vom Hintergrund durch einen Schwellwertfilter. Da die Grauwerte als Tiefeninformationen interpretiert werden können, bildet der Schwellwert die Entfernung des Bodens zur Kamera ab. Als Resultat liegt ein Binärbild vor, das als Maske dient, um die zum Objekt gehörigen Bildpunkte zu extrahieren (siehe Abb. 4.6a).

Die Positionsinvarianz wird erreicht, indem das Objekt anhand des Objektschwerpunkts zur Bildmitte verschoben wird (Abb. 4.6b).

Die Unabhängigkeit der Klassifikation von der Rotation wird mittels Tiefengradientenbetrachtung ermöglicht. Das Objekt wird so rotiert, dass der Gradient mit der  $X$ -Achse übereinstimmt (Abb. 4.6c). Falls kein nennenswerter Gradient vorliegt, etwa bei einem Tisch, wird die kürzeste Seite zur  $X$ -Achse ausgerichtet. Somit ist dieses Vorgehen objektunabhängig, wobei jedoch generelle Objektmerkmale ausgewertet werden.

Die ermittelten Informationen über Translation und Rotation des Objekts werden genutzt, um anhand der Kameraregistrierung die globale Pose des Objekts zu bestimmen.



**Abbildung 4.6:** Objektsegmentierung. Objektmaskierung (a), Translation zur Bildmitte (b) und Orientierungseinstellung basierend auf Tiefengradient des Objekts (c).



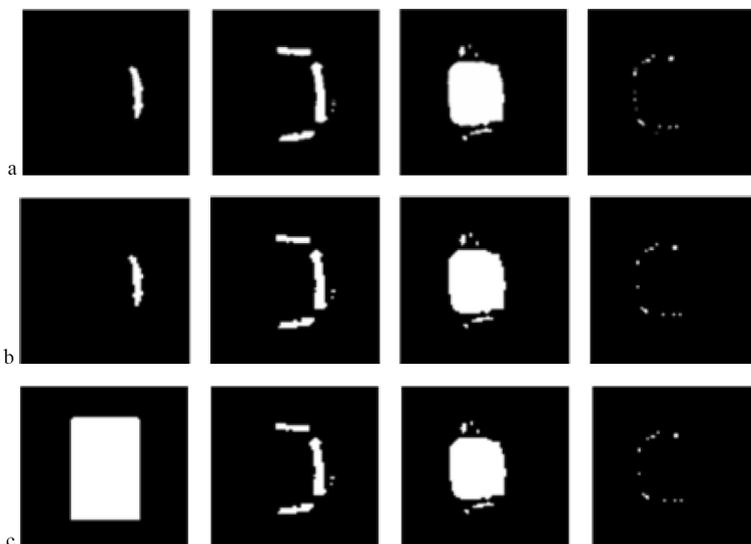
**Abbildung 4.7:** Tiefenabhängige Objektschichtung.

**Erkennung** Das neuronale Netz setzt sich aus verschiedenen Schichten zusammen. In der Arbeit von [Constantini et al., 2006] wird jeder Schicht ein Bit des Gray-Codes eines Pixelwertes zugeordnet. Im Rahmen der vorliegenden Arbeit repräsentieren die Pixelwerte Tiefeninformation. Um die räumliche Auflösung zu erfassen, wird der Raum unter der Kamera in Schichten aufgeteilt, die den Schichten des neuronalen Netzes entsprechen (siehe Abb. 4.7). Jede Schicht wird daraufhin binarisiert, wobei binomiale Werte verwendet werden (siehe Abb. 4.8a).

Während des Erkennungsprozess wird ein Muster als Eingabe für alle Netze verwendet. Da Konvergenz für Hopfield-Netze garantiert ist, wird jedes Netz so häufig aktiviert, bis Konvergenz erreicht wird. Demzufolge kann die Klassifizierung nicht über den Fakt der Konvergenz durchgeführt werden, da spontane Attraktoren eintreten können (siehe Abb. 4.8c). Stattdessen werden folgende Kriterien analysiert:

- Anzahl der Aktivierungen bis Konvergenz erreicht ist,
- das resultierende Energieniveau des Netzes nach der ersten Aktivierung,
- die Summe der resultierenden Energieniveaus nach den ersten beiden Aktivierungen.

Falls das Eingabemuster ähnlich zum gelernten Muster ist, sind die Unterschiede zwischen den Energieniveaus zwischen den Aktivierungen klein und wenige Aktivierungen sind nötig. Falls das Eingabemuster sehr unterschiedlich zum gelernten Muster ist, ist ein großer Energieunterschied zu überwinden, wodurch bei den ersten Aktivierungen große Niveauunterschiede auftreten. Bei nachfolgenden Aktivierungen können die Änderungen wieder gering sein, so dass auch in solchen Fällen wenige Aktivierungen nötig sind. Die Effekte werden im Abschnitt 4.2.4 untersucht.



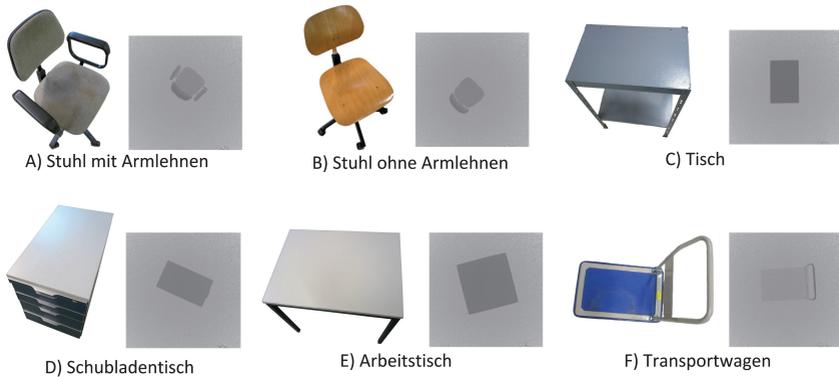
**Abbildung 4.8:** Beispiel eines Eingabemusters (a), konvergierte Muster eines Netzes, das das Muster gelernt hat (b), und eines Netzes, das ein anderes Muster gelernt hat (c).

**Prozessbeschleunigung** Da alle neuronalen Netze evaluiert werden müssen, um eine Klassifizierung durchführen zu können, skaliert die Berechnungszeit mit der Anzahl der modellierten Objekte. Um eine solche direkte Abhängigkeit zu vermeiden, ist es zielführend, nur Netze zu evaluieren, die auf ähnliche Objekte trainiert wurden. Unnötige Berechnungen können somit ausgelassen werden.

Während der Objektsegmentierung werden die generellen Objektmerkmale Objektgröße und Tiefengradient bestimmt. Diese Merkmale können ausgenutzt werden, um eine grobe Vorklassifikation durchzuführen. Dementsprechend wird beim Einlernen eines Objektes das Netz mit einer korrespondierenden groben Klasse markiert. Während der Erkennung wird das Objekt vorklassifiziert und nur relevante Netze werden evaluiert.

## 4.2.4 Evaluation und Diskussion

Um die Performanz und die Erkennungsqualität der vorgestellten Methode zu untersuchen, wurden verschiedene Objekte genutzt, die in produktiven Umgebungen zu finden sind. Diese sind in Abbildung 4.9 dargestellt. Diese Objekte weisen zum Teil Ähnlichkeiten auf, die die Diskriminierung der Objekte erschweren können. Die Bürostühle unterscheiden sich bspw. nur im Vorhandensein der Armlehnen. Der Tisch und Schubladentisch unterscheiden sich nur wenig in Größe und Höhe. Zur Evaluation wurden die Objekte mit unterschiedlichen Orientierungen in der Szene positioniert, so dass diese in der Zentrum bzw. in Randbereichen der Sensordaten vorzufinden waren. Die Sensordaten wurden manuell annotiert, um die Erkennungsqualität bestimmen zu können.



**Abbildung 4.9:** Zur Evaluation verwendete Objekte. Dargestellt sind Fotos und korrespondierende Tiefeninformation vom Sensor.

Die Erkennungsergebnisse sind von einer Vielzahl von Kriterien abhängig. Die folgenden Kriterien wurden untersucht, um deren Einfluss auf den Erkennungsprozess zu analysieren:

**Netzauswahl** Verschiedene Auswahlkriterien zur Klassifikation sind möglich und wurden im vorherigen Abschnitt eingeführt.

**Objektposition** Sensorrauschen und Verzerrungen treten vor allem im Randbereich der Sensordaten auf.

**Netzdimensionen** Kleine Netze wirken sich positiv auf Prozessgeschwindigkeit aus, jedoch sollten die Auswirkungen auf die Erkennungsqualität bekannt sein.

**Schichtanzahl** Granularität der Tiefendiskretisierung wirkt sich ebenfalls auf Verarbeitungsgeschwindigkeit und Qualität aus.

**Laufzeit** Betrachtungen zur Verabreichungsgeschwindigkeit.

**Vorklassifizierung** Auswirkungen bezüglich Beschleunigung mittels Vorklassifizierung.

#### 4.2.4.1 Erkennung ohne Vorklassifikation

Die Erkennungsraten für Objekte in der Bildmitte sind in Tabelle 4.2 dargestellt. Die Ergebnisse zeigen den Einfluss der Schichtdimensionen auf die Erkennungsqualität. Jedes Netz bestand aus acht Schichten. Es ist ersichtlich, dass die Schichtgröße nur marginale Auswirkungen hat, wobei diese ebenfalls von Objekt zu Objekt unterschiedlich sind.

**Tabelle 4.2:** Erkennungsraten ohne Vorklassifikation für Objekte in der Bildmitte in Abhängigkeit der Größe der Schichten. Die Anzahl der Schichten sind jeweils 8. Als Kriterium wurde die Anzahl der benötigten Aktivierungsschritte verwendet.

Objekt	80x80	60x60	48x48
Stuhl mit Lehne	54%	61%	58%
Stuhl	72%	77%	80%
Tisch	73%	72%	73%
Schubladentisch	91%	85%	84%
Arbeitstisch	97%	97%	89%
Transportwagen	90%	88%	87%

Die Erkennung des *Stuhls mit Armlehne* schneidet mit Werten zwischen 54% und 61% am schlechtesten ab. Alle anderen Objekte erreichen wesentlich bessere Ergebnisse. Anhand der entsprechenden Konfusionsmatrix, dargestellt in Abbildung 4.10, wird erkennbar, dass *Stühle mit Armlehne*

am häufigsten mit *Stühlen* ohne solche Lehnen verwechselt werden. Auf Grund der großen Ähnlichkeit dieser zwei Objekte ist ein solches Verhalten zu erwarten und klärt die geringe Performanz aus Tabelle 4.2 auf.

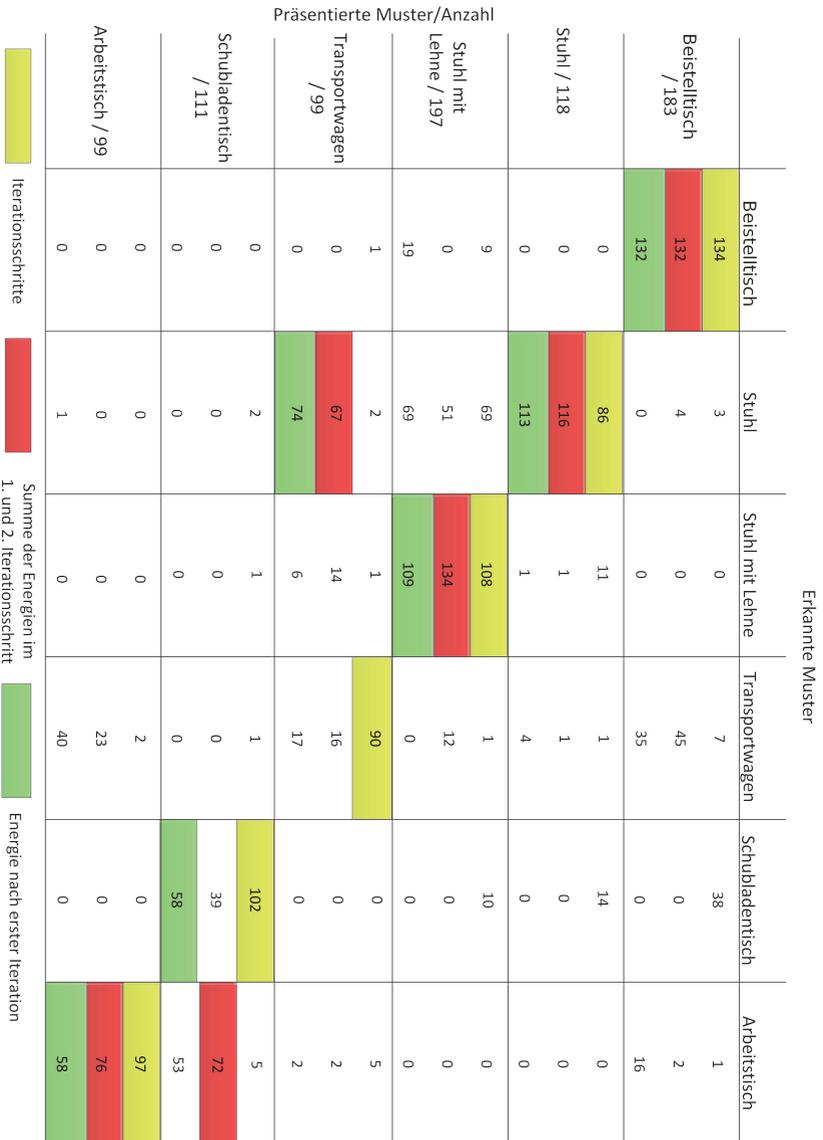
In Abbildung 4.10 wird der Einfluss der verschiedenen Klassifikationskriterien deutlich. Die verschiedenen Einträge je Muster entsprechen unterschiedlichen Kriterien. Die gleiche Analyse für Objekte nahe des Bildrandes ist in Abbildung 4.11 dargestellt. Vergleichend wird der Einfluss der Kriterien offensichtlich. Entsprechend ist das geeignete Kriterium zur Erkennung der Stühle die Energiesumme nach den ersten zwei Aktivierungsschritten. Im Gegensatz dazu ist das Kriterium über die Anzahl der Aktivierungsschritte für alle anderen Objekte aussagekräftig.

Gleichermaßen wird deutlich, dass die Position der Objekte einen klaren Einfluss auf die Erkennungsqualität hat. Die Ergebnisse aus Abbildung 4.11 weisen eine größere Diversität im Vergleich zu Abbildung 4.10 auf.

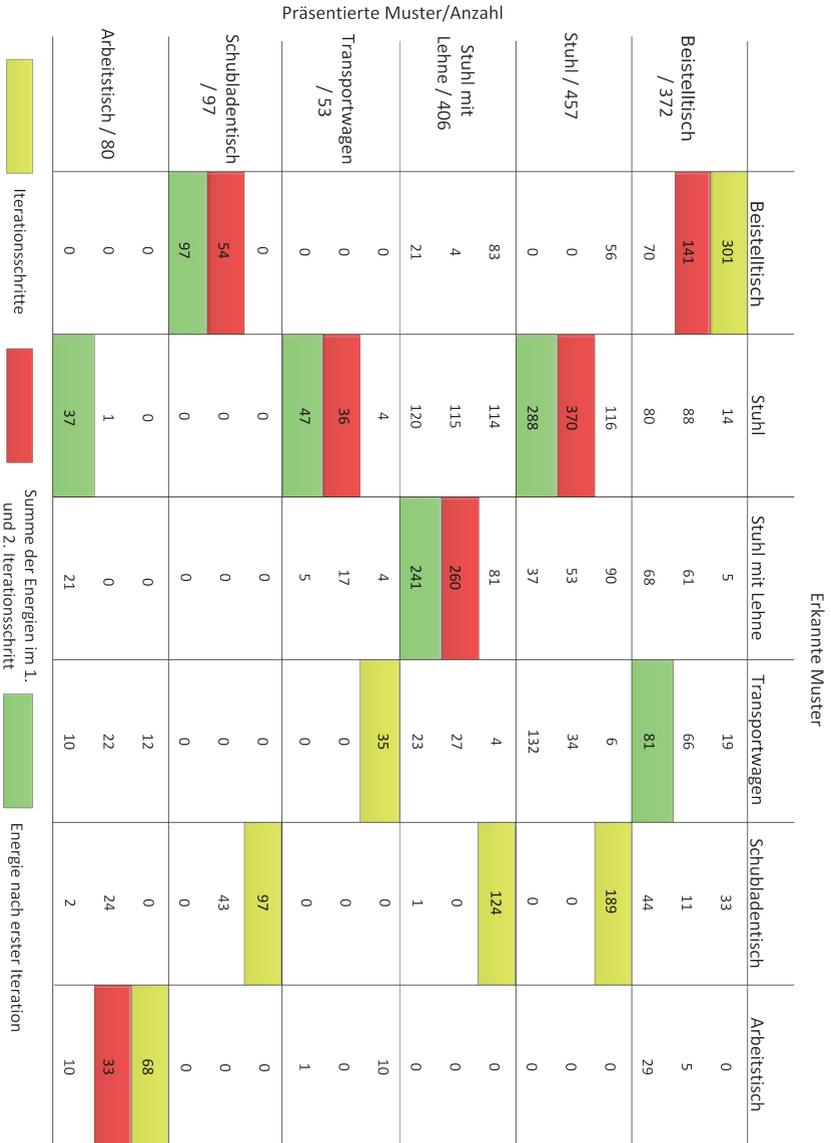
Um den Einfluss der Schichtanzahl zu untersuchen, wurde die Evaluation aus Tabelle 4.2 mit konstanter Schichtgröße vom  $80 \times 80$  Neuronen wiederholt. Die Ergebnisse sind in Tabelle 4.3 aufgeführt. Daraus wird ersichtlich, dass keine klare Aussage zur Wahl der Schichtzahl getroffen werden kann. Auch eine Tendenz zur Erhöhung oder Verminderung der Schichtanzahl kann nicht abgelesen werden. Anhand der Werte aus Tabelle 4.3 kann entschieden werden, dass generell acht Schichten durchschnittlich die besten Ergebnisse liefern. Folglich bringt eine Netzkonfiguration mit  $80 \times 80 \times 8$  Neuronen die besten Ergebnisse.

#### 4.2.4.2 Erkennung mit Vorklassifikation

Basierend auf allgemeinen Objektmerkmalen wurden verschiedene grobe Klassen definiert. Im ersten Schritt wird die Objektgröße als Merkmal genutzt, wodurch die Stühle einer Klasse und alle anderen Objekte einer zweiten Klasse zugeordnet werden. Dieser Ansatz ermöglicht den gezielten selektiven Einsatz der Klassifikationskriterien. Folglich wird für die erste grobe Klasse die Klassifikation anhand der Energiesumme der ersten zwei Aktivierungsschritte durchgeführt. Dieses Vorgehen resultiert in Erkennungsraten von 74% und 99% für den Stuhl mit und ohne Armlehnen



**Abbildung 4.10:** Konfusionsmatrix der Erkennungsergebnisse für Objekte in Bildmitte.



**Abbildung 4.11:** Konfusionsmatrix der Erkennungsergebnisse für Objekte am Bildrand.

**Tabelle 4.3:** Erkennungsraten ohne Vorklassifikation für Objekte in der Bildmitte in Abhängigkeit der Anzahl der Schichten. Die Größen der Schichten beträgt 80x80 Neuronen. Als Kriterium wurde die Anzahl der benötigten Aktivierungsschritte verwendet.

Objekt	12 Schichten	8 Schichten	4 Schichten
Stuhl mit Lehne	60%	54%	86%
Stuhl	61%	72%	55%
Tisch	95%	73%	76%
Schubladentisch	56%	91%	63%
Arbeitstisch	100%	97%	100%
Transportwagen	74%	90%	76%

(siehe Tabelle 4.4). Auf Grund der Entfernung von wetteifernden Modellen werden gleichermaßen die Ergebnisse der anderen Objekte verbessert.

In einem zweiten Schritt kann das zusätzliche Merkmal des Tiefengradienten genutzt werden, wodurch drei grobe Klassen entstehen (siehe Tabelle 4.4). Da der Transportwagen sich speziell durch seinen hohen Griff und dem daraus folgenden Gradienten von den Tischen unterscheidet, bildet dieser das einzige Element der dritten Klasse. Es ist offensichtlich, dass diese Einteilung für die Erkennung des Transportwagen voreingenommen ist, jedoch erlaubt diese Vorklassifikation bessere Performanz und schränkt die zukünftige Hinzunahme weitere Objekte nicht ein.

Bezüglich der Positionierung der Objekte im Bild sinkt die Erkennungsrate für Objekte nahe des Bildrandes. Bei den Stühlen können Erkennungsraten von 71% (mit Armlehnen) und 87% (ohne Armlehnen) erreicht werden.

#### 4.2.4.3 Laufzeitanalyse

Anhand der Ergebnisse in Tabelle 4.4 wurde das Hauptaugenmerk bei der Laufzeitanalyse auf den Vergleich der Klassifikation ohne Vorklassifikation und Klassifikation mit drei groben Vorklassen. Die Ergebnisse sind in Tabelle 4.5 aufgeführt. Dabei wird ersichtlich, dass die Verwendung der Vorklassifikation generell bessere Laufzeiten zur Folge hat, da weniger Netze

**Tabelle 4.4:** Vergleich der Erkennungsergebnisse für verschiedene Vorklassifizierungen.

<b>Objekte / # Vorklassen</b>	<b>1</b>	<b>2</b>	<b>3</b>
Stuhl mit Armlehnen	54%	74%	74%
Stuhl ohne Armlehnen	72%	99%	99%
Beistelltisch	73%	73%	73%
Schubladentisch	91%	94%	95%
Arbeitstisch	97%	97%	100%
Transportwagen	90%	92%	100%

evaluiert werden müssen. Gleichmaßen wird ohne Vorklassifikation deutlich, dass die Laufzeiten mit der Reihenfolge der Modelle zusammenhängt, da der Transportwagen als letztes Modell evaluiert wird.

Ebenfalls wird deutlich, dass die Positionierung der Objekte Einfluss auf die Laufzeit hat. Für alle Objekte, ausser dem Schubladentisch, verschlechtern sich die Zeiten wenn die Objekte am Bildrand positioniert sind.

**Tabelle 4.5:** Vergleich der Laufzeitanalyse in Bilder pro Sekunde ohne (links) und mit Vorklassifizierung (rechts) und für verschiedene Objektpositionen: im Bild zentriert (Z) oder nahe des Bildrandes (R).

<b>Objekt</b>	<b>Z</b>	<b>R</b>	<b>Z</b>	<b>R</b>
Stuhl mit Armlehnen	9	7	22	15
Stuhl ohne Armlehnen	11	10	26	17
Beistelltisch	12	9	20	16
Schubladentisch	7	16	14	28
Arbeitstisch	10	5	17	9
Transportwagen	5	3	23	16

#### 4.2.4.4 Diskussion

Eine weitere Methode basierend auf globalen Gabor-Merkmalen [Kamarainen et al., 2007] wurde implementiert, um die Ergebnisse zu vergleichen. Durch die Arbeit von [Kamarainen et al., 2007] wurde nachgewiesen, dass Objekterkennung mittels Gabor-Merkmalen robust gegenüber verrauschten Daten ist und bessere Ergebnisse liefert als Methoden, die auf direkten Invarianten arbeiten, wie bspw. Momenten-Invariante wie Hu-Momente. Die mit dieser Methode erreichten Erkennungsraten für ausgewählte Objekte sind in Tabelle 4.6 aufgeführt. Die Ergebnisse schwanken zwischen 61% und 98%, wobei die Stühle am schlechtesten abschneiden. Zur Auswertung werden 83 ms pro Bild benötigt, wodurch etwa 12 Bilder pro Sekunde verarbeitet werden können. Im Vergleich erreicht der vorgestellte neuronale Klassifikator mit Vorklassifikation bessere Ergebnisse und schnellere Bildraten.

**Tabelle 4.6:** Erkennungsergebnisse der Objekterkennung für ausgewählte Objekte mittels Gabor-Merkmalen.

Objekt	Erkennungsrate
Schubladentisch	98%
Beistelltisch	77%
Stuhl	70%
Stuhl mit Armlehne	61%

Dabei ist anzumerken, dass die zur Vorklassifizierung nötigen Informationen zwangsläufig für den neuronalen Klassifikator bestimmt werden, um Positions- und Orientierungsinvarianz zu erreichen. Bei den Gabor-Merkmalen werden diese Informationen nicht ermittelt, wodurch diese Form der Vorklassifizierung zusätzliche Berechnungen nötig machen würde und somit die geschlossene Form der Gabor-Merkmale ihre Vorteile verliert.

Folglich zeigen die Ergebnisse, dass die entwickelte Methode des neuronalen Klassifikators mittels assoziativen Speichers unter den gegebenen Rahmenbedingungen gute Resultate liefert. Die evaluierten Objekte können

mittels Vorklassifizierung zuverlässig erkannt werden, wobei hinreichende Bildraten zu erreichen sind.



# 5 Handlungserkennung und Situationsverstehen

Als wesentlicher Aspekt der vorliegenden Arbeit stehen die Aufgaben Situationen in einem industriellen Arbeitsumfeld zu analysieren, zu erkennen und entsprechende Schlussfolgerungen abzuleiten. Zusammengefasst kann somit Situationsverständnis des technischen Systems erreicht werden. Das Verständnis wirkt sich auf die Handlungen des Roboters aus, in dem Roboteraktionen ausgelöst und anhand resultierender Sicherheitsaspekten angepasst werden.

Zur Situationsanalyse gehört die Erkennung von Umwelteinflüssen, d.h., die Erkennung menschlicher Handlungen und Bewegungen. Durch Interpretation der Analyseergebnisse kann auf die vorliegende Situation geschlossen werden, wodurch entsprechende Reaktionen des Systems abgeleitet werden können. Darüber hinaus können die Ergebnisse genutzt werden, um laufende Arbeitsprozesse zu analysieren.

Im Folgenden werden die entwickelten Methoden und Ansätze zu den einzelnen Teilaspekten detailliert dargestellt und erörtert.

## 5.1 Begriffsbildung

Bevor die Herangehensweisen und Methoden erläutert werden, sollen vorweg die wesentlichen Kernbegriffe herausgestellt werden, die im Rahmen dieser Arbeit wichtig sind.

**Gesten, Handlungen und Kooperation** Die nonverbale Kommunikation zwischen Menschen besteht zu einem wesentlichen Teil aus Gesten. Aussagen werden verdeutlicht und untermalt, indem zeichenhafte Bewegungen bestimmter Körperteile verwendet werden. Beim Beispiel der Gebärdensprache stellen Gesten die Hauptform der Kommunikation dar [Kendon, 2004].

Handlungen stellen Prozesse dar, während derer etwas getan oder durchgeführt wird. Somit wird nicht allein ein kommunikativer Prozess betrachtet, wodurch der Handlungsbegriff mehr als Gesten umfasst. Gesten sind wiederum Handlungen, denen Bedeutungen zugeordnet werden. Andere Handlungen können jedoch auch an Bedeutung gewinnen, wenn diese im Kontext analysiert und interpretiert werden sollen. So können Handlungen nachvollziehbar werden, wenn Rahmenbedingungen oder vorherige Handlungen bekannt sind.

Handlungen können andere Handlungen umschließen oder wiederum Teil von anderen sein. So reihen sich Prozessschritte als Handlungen bei der Produktion aneinander und sind gleichzeitig Teil der gesamten Produktionshandlung. Somit können durch Betrachtung des Kontextes zukünftige Handlungen prädictiert werden.

Aus dem Zusammenwirken von Handlungen verschiedener Partner kann sich Kooperation entwickeln. Dazu müssen die Handlungen zu einem gemeinsamen Ziel führen, weshalb diese aufeinander abgestimmt sein müssen. Folglich bedingen die Handlungen eines Kooperationspartners die Handlungen des anderen Partners. Dabei wird gleichermaßen ein der Kooperation entsprechendes Verhalten gegenseitig von den Partnern erwartet. Daher sind Kooperationen auf eine Domäne, einen Sachverhalt oder eine Situation beschränkt [Axelrod, 1985]. Eaton zufolge kann Kooperation darüber hinaus als direkt oder indirekt eingeteilt werden [Eaton, 1948]. Da bei der MRK verschiedene Teilaufgaben durch Mensch und Maschine abgearbeitet werden sollen, um die Stärken beider zu vereinen, handelt es sich folglich um indirekte Kooperation.

**Situationsverständnis und -bewusstsein** Situationsbewusstsein<sup>1</sup> ist vorrangig in der Forschung über menschliche Einflussfaktoren in dynamischen

---

<sup>1</sup>engl.: Situation Awareness

Systemen behandelt worden. Entsprechend sind in diesem Zusammenhang erste Definitionen und Begriffsbildungen zu finden. Laut der Arbeit von [Wittbrodt et al., 2009], ist die erste Definition des Begriffs Situationsbewusstsein in [Endsley, 1995] aufgeführt. Darin heißt es:

„Situation awareness is the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future.“

Diese Aussage lässt sich wie folgt übersetzen:

„Situationsbewusstsein ist die Wahrnehmung der Elemente einer Umgebung innerhalb eines örtlich und zeitlich begrenzten Bereichs, das Verstehen deren Bedeutung und die Projektion deren jeweiligen Status in die nahe Zukunft.“

Entsprechend dieser Definition von Endsley ist, analog zur Beschränkung von Kooperation auf eine Domäne bzw. einen Sachverhalt oder eine Situation, der Begriff des Situationsbewusstsein auf einen räumlich und zeitlich begrenzten Bereich festgelegt. Somit kann man folgern, dass Kooperation eng mit Situationsverständnis gekoppelt ist, da sich der Kooperation entsprechende Erwartungshaltungen aus einer Situation ableiten.

Bei der Definition von Endsley wird jedoch nicht ausdrücklich festgelegt, ob sich Situationsbewusstsein auf einen Prozess oder einen Zustand als solchen bezieht. In [Wittbrodt et al., 2009] wird daher zur Erweiterung dieser Definition zwischen *Situation Assessment*, *Situational Model* und *Situation Awareness* unterschieden. Dabei ist, nach Wittbrodt et al., unter diesen Schlagwörtern folgendes zu verstehen:

**Situation Assessment** ist der „Prozess zum Erreichen eines bestimmten Grades von Situationsbewusstsein.“

**Situational Model** ist das „Produkt dieses Prozesses in Form des aktuellen und sich dynamisch verändernden Wissens über die Situation.“

**Situation Awareness** ist der „Grad der Vollständigkeit und Fehlerfreiheit mit dem das Situational Model die relevanten Elemente der Situation repräsentiert bzw. ihre Entwicklung antizipiert.“

Demzufolge handelt es sich bei Situational Model und Situation Awareness um Zustände, die eine interne Repräsentation der Situation darstellen und bewerten. Das Situation Assessment als Prozess erstellt diese interne Repräsentation.

Im Rahmen der vorliegenden Arbeit wird die erweiterte Definition von Wittbrodt et al. zu Grunde gelegt, wobei die Begriffe Situationsbewusstsein und -verständnis häufig synonym verwendet werden. Das Verstehen einer Situation ist mit dem Bewusstsein über die Situation gleichbedeutend.

**Ontologie und Wissen** Der Begriff *Ontologie* ist im großen Maße philosophischer Natur und befasst sich mit „der Lehre vom Sein, von den Ordnungs-, Begriffs- und Wesensbestimmungen des Seienden“ [Duden, 1990]. Neben der philosophischen *klassischen* Ontologie, die sich hauptsächlich mit metaphysischen Themen auseinandersetzt, gibt es die *analytische* Ontologie, die allgemeine Strukturen und Eigenschaften der Wirklichkeit analytisch herausarbeitet. Eine Ontologie stellt daher auch die Frage nach grundlegenden Kategorien bzw. Konzepten und nach den Beziehungen zwischen den Teilen und dem Ganzen.

Bei der Wissensabbildung der Konzepte und Beziehungen spielen deren Definitionen, Begriffsbildung und -abgrenzung eine wesentliche Rolle. Die Bedeutung eines Begriffs wird durch das *Denotat* und *Konnotat* gegeben. Das Denotat bezeichnet die außersprachliche Wirklichkeit, wobei das Konnotat auch emotionale Wirkungen berücksichtigt. Beispielsweise ist der Begriff *Roboter* im Denotat nur durch das physische Objekt gekennzeichnet. Das Konnotat von *Roboter* verbindet auch Begriffe wie Präzision, Stärke und Gefahr.

Die Trennung von Denotat und Konnotat ist für viele Begriffe schwierig und wird mit steigender Komplexität der Objekte und Sachverhalte nahezu unmöglich. Folglich wird eine klare Abgrenzung und deutliche Unterscheidung von Begriffen erschwert. Es bilden sich daraus Begriffslandschaften, die durch scharfe und schwache Begriffsbegrenzungen gekennzeichnet sind [Proske, 2006]. Schwache bzw. unbestimmte Begrenzungen führen zu Unklarheiten und werden als lexikalische Unschärfe bezeichnet. Diese Unschärfe ist häufig ein Grund dafür, dass wissensbasierte Systeme in der Praxis auf gut abgrenzbare Anwendungsfelder beschränkt sind.

Die situative Beschränkung von Kooperation und die räumlich-zeitliche Beschränkung von Situationsbewusstsein können ausgenutzt werden, um die lexikalische Unschärfe bei Begriffsdefinitionen zu minimieren.

**Handlungsziele** Die Analyse menschlicher Entscheidungsprozesse wird im Forschungsbereich des Operation Research untersucht [Montibeller and Belton, 2006; Bond et al., 2010]. Es zielt darauf ab, die Entscheidungsfindung von Entscheidungsträgern zu verstehen, zu modellieren und zu verbessern. In der Arbeit von [Leonard, 2009] werden menschliche Entscheidungsprozesse mit Roboterfeedback gekoppelt, um dem Menschen optimale Entscheidungen zu ermöglichen.

Bevor ein Robotersystem in der Lage ist zu planen, wie eine Aufgabe gelöst werden kann, muss zuvor ein Ziel identifiziert werden. Solche Ziele können auf verschiedene Weise definiert werden. Eine Form ist die direkte Vorgabe durch einen menschlichen Nutzer im Bereich der Haushaltsroboter, wie bspw. „Bereite Pfannkuchen zu“ [Beetz et al., 2011]. Im Bereich der Fussballroboter sind die Handlungsziele durch Regeln und dem Streben nach Spielgewinn definiert [Tucnik, 2007].

Im Bereich der industriellen Mensch-Roboter-Kooperation ist für die Zielfindung ein Entscheidungsprozess nötig, der situativ Handlungsziele anpasst bzw. überhaupt erst definiert.

## 5.2 Erkennung und Interpretation der menschlichen Position

Menschliche Handlungen können im Kontext ihres Ausübungsortes betrachtet werden. Auf diese Weise können eventuelle Vieldeutigkeiten von Beobachtungen aufgelöst werden, da Einschränkungen oder Voraussetzungen durch den Ort gegeben sind. Ebenfalls können gleiche Handlungen unterschiedlich interpretiert werden, wenn diese an unterschiedlichen Positionen ausgeführt werden. Im Folgenden wird sich mit der Problematik der Erkennung und Interpretation der menschlichen Position beschäftigt.

### 5.2.1 Stand der Forschung

Die Position eines Menschen ist speziell dann interessant, wenn ortsbezogene Dienste den möglichen Konsumenten ansprechen sollen. In der Arbeit von [Marmasse and Schmandt, 2002] werden GPS-Koordinaten ausgewertet, um eine Person zu verfolgen und sogenannte *virtuelle Plätze* zu identifizieren. Anhand dieser Ortsinformationen können eMails oder webbasierte Inhalte an den Nutzer weitergegeben werden. Verschiedene Ansätze werden ausgewertet um Ortskorrelationen zu lernen und Bewegungsmuster vorherzusagen. Zu diesen Ansätzen zählen Bayes Klassifikatoren, Histogramm Modelle und HMMs. In der Arbeit wird festgestellt, dass Histogramm Modelle am besten für Datenabstraktion und Vorhersage geeignet sind.

Ein ähnlicher Ansatz wird durch [Ashbrook and Starner, 2002] verfolgt. Dabei wird argumentiert, dass die Aufgaben eines Nutzers mit dem Aufenthaltort in Bezug stehen. Durch die Prädiktion der Nutzerbewegung können Aufgaben frühzeitig zeitlich eingeordnet oder beeinflusst werden. Die Anzahl der Orte wird dabei durch Clusteranalyse der GPS Daten bestimmt. Die Verweildauer an einem Ort wird bei der Analyse berücksichtigt. Die Beziehungen zwischen Orten werden durch HMMs modelliert, welche dann auch für die Prädiktion eingesetzt werden.

In der Arbeit von [Eagle et al., 2009] werden Daten von Funkmasten analysiert, um Positionsinformation von Handybenutzern zu erhalten. Die Daten werden mittels unüberwachter Clusteranalyse basieren auf Graphsegmentierung verarbeitet. Die ermittelten Orte werden als Zustände in mehreren dynamischen Bayes Netzen (DBN) für die Nutzerbewegungsprädiktion genutzt. Darüber hinaus können abnormale Nutzerverhalten durch eine Anpassung der DBNs erkannt werden.

Die Ausnutzung des semantischen Wissens eines Aufenthaltsortes eines Menschen für die Erkennung menschlicher Aktivitäten wird in [Yamada et al., 2007] vorgestellt. Einerseits wird festgestellt, dass Orte verschiedene Aktivitäten auslösen können, andererseits sind diese Aktivitäten eng mit genutzten Objekten verbunden. Entsprechend werden Orte als *Aktivitätsräume* definiert und können indirekt durch die Verwendung von Objekten modelliert werden. Anhand der Aktivitätsräume können dem

Nutzer verschiedene Dienste proaktiv angeboten werden, wodurch explizite Anfrageprozeduren vermieden werden können.

Die Ausnutzung der Vorhersage der menschlichen Position zur Anpassung einer Robotertrajektorie wird in [Tadokoro et al., 1993] umgesetzt. Die menschliche Bewegung ist als Markovprozess modelliert, wobei die Prädiktion für die Bestimmung eines Gefährungsmaßes verwendet wird. Anhand dessen wird die Robotergeschwindigkeit proaktiv angepasst. In der vorliegenden Arbeit wird eine situationsabhängige Risikobestimmung zur Roboterbahnanpassung genutzt. Dabei ist neben Kopforientierung und Bewegungsrichtung auch die relative Position von Mensch und Roboter von Bedeutung.

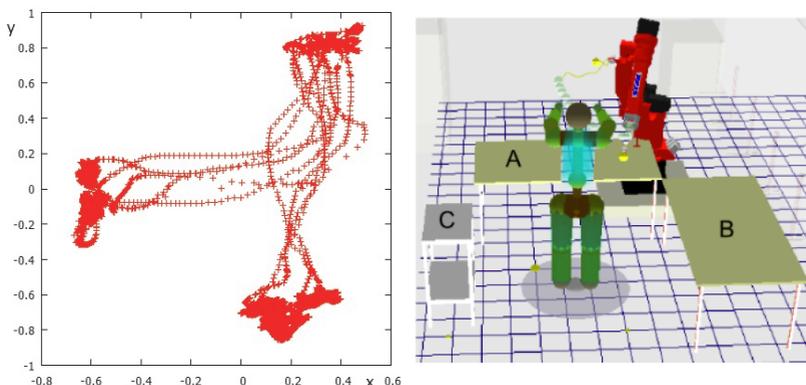
### 5.2.2 Ortsbestimmung, Lokalisierung und Prädiktion

In der vorliegenden Arbeit wurden Gaußsche Funktionen genutzt, um Potentiale im Arbeitsraum zu definieren, welche zur Lokalisierung und Prädiktion dienen. Ein erster Ansatz hierzu wurde in [Puls and Wörn, 2013] vorgestellt. Der hauptsächliche Vorteil Gaußscher Funktionen ist deren Fähigkeit Orte als Positionsverteilungen abzubilden anstatt auf singuläre Punkte im Raum angewiesen zu sein. In Abbildung 5.1 ist ein Beispiel für ein Trackingergebnis eines Menschen im Arbeitsraum des Roboters abgebildet. In diesem Fall enthält der Arbeitsraum drei verschiedene Orte, von denen zwei kooperative Bereiche sind.

Wissen über den Aufenthaltsort des Menschen basiert auf einer semantischen Abbildung der Trackingkoordinaten auf die Konzepte der Orte. Diese Abbildung stellt die Grundlage für anschließende Interpretationen und Schlussfolgerungen dar. Der Abbildungsprozess unterscheidet sich anhand der Aufgabe: Lokalisierung oder Prädiktion.

Wie in Abbildung 5.1 (links) zu sehen ist, bilden die dargestellten Koordinaten Anhäufungen, die mit unterschiedlichen Orten (siehe Abb. 5.1 rechts) korrespondieren. Entsprechend wird die Ortsbestimmung mithilfe unüberwachter Clusteranalyse durchgeführt. Der DBSCAN-Algorithmus [Ester et al., 1996] erlaubt die Clusteranalyse ohne vorheriges Wissen über die Anzahl der zu erwartenden Cluster. Entsprechend ist die Ortsbestimmung unabhängig vom Anwendungsfall, wodurch ein genereller Ansatz

erreicht wird. Als Ergebnis der Analyse kann ein jeder Datenpunkt zu einem Cluster zugeordnet werden. Während der Analyse werden Datenpunkte als Rauschen betrachtet, die zu keinem Cluster zugeordnet werden können. Somit werden Trackingpositionen bei Übergängen zwischen Orten entsprechend aussortiert.



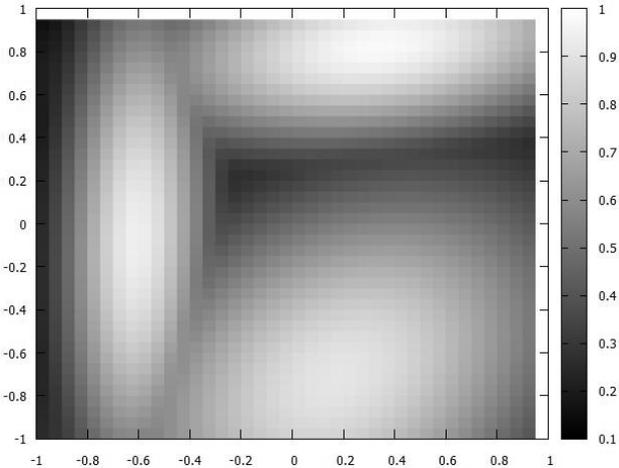
**Abbildung 5.1:** Verfolgungsergebnis eines Menschen im Arbeitsraum (links). Arbeitsraumdarstellung von der Seite (rechts).

Nach der Ortsbestimmung werden die extrahierten Cluster genutzt, um anziehende Potentialfunktionen zu bestimmen. Diese Funktionen werden für die Lokalisierung und Prädiktion herangezogen. Zur Bestimmung der Gaußschen Funktionen werden die Mittelwerte und Kovarianzen der Cluster ermittelt. Anschließend werden die Funktionen normalisiert (siehe Abb. 5.2).

Für Lokalisierungszwecke stellen die Gaußschen Funktionen Positionsverteilungen dar und definieren einen Grad der Zugehörigkeit einer Position zu einem Ort. Entsprechend wird die Lokalisierung auf die Anwendung einer Schwellwertfunktion auf die Gaußschen Funktionen zurückgeführt. Falls ein Wert über dem Schwellwert liegt, wird angenommen, dass die Person an dem korrespondierenden Ort ist.

Die Prädiktion basiert auf zwei Faktoren: Bewegungsrichtung und anziehende Wirkung der Potentialfunktion. Die Anziehung einer Potentialfunktion

ist als deren Gradient in Bewegungsrichtung definiert. Entsprechend kann der nächste Aufenthaltsort prädiziert werden sobald sich die Person in Bewegung setzt. Die prädizierte Bewegungsrichtung wird durch die Anwendung der Kalman Prädiktion erreicht [Bar-Shalom et al., 2001].



**Abbildung 5.2:** Normalisierte Gaußsche Funktionen als Repräsentanten der Cluster.

## 5.2.3 Evaluation und Diskussion

Um die Lokalisierung und Prädiktion zu evaluieren, wurde ein Produktionsprozess imitiert, in dem eine Person Objekte an verschiedenen Arbeitsplätzen zu benutzen hatte. Die Plätze entsprechen der Darstellung in Abbildung 5.1.

### 5.2.3.1 Lokalisierung

Auf Grund der Modellierung der Orte als Verteilungen erreichte die Lokalisierung eine hohe Zuordnungsrate von 100%. Dafür ist auch die klare Anordnung der Arbeitsplätze mitverantwortlich, die eine genaue Clusteranalyse

erlaubt. In produktiven Umgebungen sind solche klaren Abgrenzungen zu erwarten, da Prozesse und Einrichtungen eine Organisationsstruktur im Arbeitsbereich erfordern. Tenoth & Beetz zu folge, gilt diese Feststellung auch für häusliche Umgebungen, in denen Utensilien und Möbel Strukturen definieren und Beschränkungen auf Aufenthaltsorte vorgeben [Tenorth and Beetz, 2009].

### 5.2.3.2 Prädiktion

Bezüglich der Evaluation der Prädiktion wurde der prädizierte Wert mit dem tatsächlich eintreffenden Wert verglichen. Dazu wurden die während der Prozessimitation Prädiktions- und Lokalisierungsergebnisse aufgezeichnet. Es wurde eine Prädiktionsrate von 90,9% erreicht. Dabei stellten speziell diagonale Bewegungen eine Herausforderung dar. Wie in Abbildung 5.1 erkennbar ist, spiegeln die Trajektorien eher Bewegungen um Ecken als direkte Wege dar. Das kann zwischenzeitlich zu falschen Prädiktionen führen. Ebenfalls von Interesse war der frühest mögliche Prädiktionszeitpunkt. Etwa 1,98 Sekunden vor dem Erreichen eines neuen Arbeitsplatzes hat sich eine Prädiktion eingestellt. Im Vergleich zu den vorliegenden Abmessungen der Arbeitszelle von 1,5 m  $\times$  1,5 m sind diese Zeiten positiv zu bewerten.

### 5.2.3.3 Diskussion

In Fällen, wo semantisch deutlich zu trennende Orte sehr eng beieinander liegen, kann die Clusteranalyse nicht zu den gewünschten Ergebnissen führen. Einerseits kann dem durch händische Anpassung entgegengewirkt werden, oder die semantische Repräsentation muss dahingehend erweitert werden, so dass alle semantischen Beziehung abgebildet werden. Folgerungen über beobachtete Handlungen können dann den entsprechenden Ort identifizieren. Die Prädiktion hingegen ist auch negativ betroffen, sobald Arbeitsplätze kollinear angeordnet sind. So überdecken örtlich nähere Potentialfunktionen weiter entfernte. Dem kann entgegnet werden, in dem die Prädiktion Wissen über den Prozessablauf berücksichtigt. Entsprechend können die Gaußschen Funktionen mit der zu erwartenden Position gewichtet werden. Um Prozessabweichungen zuzulassen, könnte ebenfalls die

Bewegungsgeschwindigkeit bei der Gradientenbestimmung berücksichtigt werden. Eine Bewegungsverlangsamung könnte nähere Funktionen höher gewichten. Jedoch leidet dabei wiederum der Prädiktionszeitpunkt, der auf diese Weise zeitlich nach hinten verlagert wird.

## 5.3 Erkennung von Handlungen und Aktionen

Im Verlauf von Interaktionen muss ein jeder Handlungspartner Wissen über ausgeführte Handlungen haben, um über eigene korrespondierende Handlungen zu entscheiden. Im Fall von Mensch-Maschine-Interaktion ist es wesentlich, menschliche Handlungen zu erkennen, um darauf aufbauend mögliche Intentionen zu folgern und Erwartungen bezüglich Systemverhalten abzuleiten. Im Folgenden wird sich daher mit der kontinuierlichen Handlungserkennung auseinandergesetzt.

### 5.3.1 Stand der Forschung

Erkennung menschlicher Aktionen und Handlungen ist ein sehr aktiver Forschungsbereich. Die Vielfalt der Klassifikationsmethoden ist in der Übersicht von [Poppe, 2010] in drei Kategorien aufgeteilt:

1. Klassifikation ohne explizites Modell von Handlungsvariationen über die Zeit,
2. Ansätze mit temporalen Zustandsräumen,
3. Aktionserkennung ohne Aktionsmodellierung.

Nach Poppe versuchen die Methoden der letztgenannten Kategorie Beobachtungen mit annotierten Videosequenzen zu korrelieren. Daher werden ähnliche Techniken eingesetzt wie bei Arbeiten aus den anderen Kategorien. Auf Grund der Annahme, dass in industriellen Anwendungen praktische bzw. ausführbare Handlungen und Prozesse bekannt sind, werden im Folgenden entsprechende Arbeiten aus den Kategorien 1 und 2 betrachtet. Darüber hinaus ist im Falle neu auftretender Aktionen eine konkrete Modellierung praktikabler als die aufwändige und umfassende Annotation von Sensordaten, wie es bei Kategorie 3 vorausgesetzt wird.

In die erste Kategorie passend, verwenden Sun und Wang den *k-nächste Nachbarn* Algorithmus, um Aktionen anhand von Pixelprojektionen und Merkmalen über Schwerpunktbewegungen zu klassifizieren. Ohne die Betrachtungen der Bewegungsmerkmale können laut Sun und Wang ähnliche Aktionen kaum differenziert werden, die sich vornehmlich durch ihre Geschwindigkeit unterscheiden, wie bspw. *laufen* und *gehen*. Entsprechend bilden die Bewegungsmerkmale auch Informationen temporaler Art ab.

Um die temporale Natur von Handlungen direkt erfassen zu können, basieren viele Ansätze auf Bayes Netzen, wie unter anderem [Loccoz et al., 2003] und [Park and Aggarwal, 2004], oder Hidden Markov Modellen (HMM). Diese Ansätze können der Kategorie 2 zugeordnet werden. Sie versuchen Beobachtungen mit probabilistischen Aktionsmodellen zu korrelieren. Speziell HMMs sind bei Erkennungsaufgaben populär, da ihre theoretischen Grundlagen gut verstanden sind und sie sich bei Anwendungen der Spracherkennung bewährt haben [Krüger et al., 2007].

In der Arbeit von Elmezain et al. wird ein System zur Erkennung von Handgesten vorgestellt [Elmezain et al., 2008]. Dabei wird argumentiert, dass die Orientierung der Bewegung das beste Merkmal im Vergleich zu Position und Geschwindigkeit der Hand ist, um eine Geste zu beschreiben. Entsprechend werden in der Arbeit nur Orientierungsmerkmale der Bewegung der verfolgten Hand für die Klassifizierung genutzt. Diskretisierungen der Orientierungen werden als Eingabe für ein HMM-Erkennungsmodul verwendet. Damit aufeinander folgende Gesten erkannt werden können, wurde eine sogenannte Null-Codewort-Erkennung eingeführt. Das Null-Codewort wird durch eine statische Geschwindigkeit definiert. Durch die Integrierung von 3D Informationen über die Handposition und Bewegung konnten die Erkennungsergebnisse weiter verbessert werden [Elmezain et al., 2009].

Mehrschichtige HMMs kommen in der Arbeit von Wu et al. zum Einsatz, um Aktionen des Oberkörpers zu klassifizieren [Wu et al., 2008]. Lokale und globale Merkmale, entsprechend in Bezug auf den menschlichen Körper oder die gesamte Bildszene, werden separat in verschiedenen HMMs ausgewertet. Diese Modelle bilden die unterste Schicht und sind für die Erkennung individueller Armaaktionen verantwortlich. Die Ausgaben dieser Schicht werden in einem neuen Merkmalsvektor zusammengefasst, welcher als Eingabe der nächst höheren Schicht dient. Diese Schicht modelliert

die Beziehungen der zwei Arme. Folglich kann dieser Ansatz auch als hierarchische Modellierung angesehen werden.

Basierend auf einer vollständigen Verfolgung des menschlichen Körpers sollen in der Arbeit von [Junxia et al., 2008] Aktionen erkannt werden. Ähnlich wie bei [Wu et al., 2008] werden lokale und globale Merkmale genutzt, um den Zustand des menschlichen Körpers zu repräsentieren. Durch eine modellbasierte Darstellung des Körpers werden globale Bewegungsinformationen und lokale Gelenkmerkmale bestimmt. Für die Klassifikation werden verschiedene HMMs parallel, jedoch nicht hierarchisch, eingesetzt. Das Klassifikationsergebnis bestimmt sich aus einem kombinierten Maximum der HMM-Ergebnisse.

Über die Handlungswiedererkennung hinausgehend versuchen Kulic et al. menschliche Bewegungsprimitive zu extrahieren [Kulić et al., 2012]. Hierzu werden menschliche Bewegungen mittels Motion-Capture-System aufgezeichnet und die resultierenden Gelenkwinkel werden durch HMMs probabilistisch segmentiert und hierarchisch geordnet. Ebenfalls werden Zusammenhänge zwischen den HMMs bestimmt und ergeben einen Bewegungsprimitive-Graphen, der bspw. die Generierung von Bewegungen für humanoide Roboter erlaubt. Im Rahmen der vorliegenden Arbeit ist jedoch das Erkennen bekannter Handlungsvorgänge wesentlich, wodurch auf weniger komplexe Methoden zurückgegriffen werden kann.

### 5.3.2 Kontinuierliche Handlungserkennung

Im Rahmen dieser Arbeit werden menschliche Aktionen mittels mehrschichtiger HMMs erkannt, um sie bei anschließenden Schlussfolgerungsschritten auszuwerten. In [Puls and Wörn, 2012] wurde der folgende Ansatz eingeführt und in [Puls and Wörn, 2013] erweitert.

#### 5.3.2.1 Hidden Markov Modelle

Eine ausführliche Einführung in das Thema von Hidden Markov Modellen wird durch die Arbeit von Rabiner gegeben [Rabiner, 1989]. Hier sollen HMMs zum Überblick eingeführt werden.

HMMs beschreiben zwei probabilistische Prozesse. Ein Prozess stellt eine sogenannte Markovkette dar, die aus Zuständen und Übergangswahrscheinlichkeiten besteht. Dieser Prozess kann als probabilistischer Zustandsautomat repräsentiert werden. Die Markov Modelle tragen das Attribut *Hidden*<sup>2</sup>, da diese Zustände nicht beobachtbar sind. Der zweite Prozess generiert beobachtbare Symbole anhand einer zustandsabhängigen Wahrscheinlichkeitsverteilung. Entsprechend kann durch die Beobachtung der Symbole der zugrunde liegende Zustand probabilistisch gefolgert werden. Dazu kann der *Viterbi*-Algorithmus eingesetzt werden.

Falls es verschiedene Modelle und eine Beobachtungssequenz von Symbolen gibt, kann dasjenige HMM identifiziert werden, welches die Beobachtung am besten widerspiegelt. Für die Evaluation der Wahrscheinlichkeit  $P(O|\lambda)$  einer Beobachtungssequenz  $O$  und eines gegebenen Modells  $\lambda$  kann der *Vorwärts*-Algorithmus genutzt werden. Um die Parameter eines HMM anhand gegebener Beobachtungen einzulernen, kann der *Baum-Welch*-Algorithmus angewendet werden.

Im Anwendungsfall der Handlungserkennung werden verschiedene HMMs mittels Vorwärts-Algorithmus evaluiert. Jede Handlung wird durch ein eigenes Markov Modell repräsentiert. Diejenige Handlung, deren Modell höchste Wahrscheinlichkeit hat, wird als erkannt behandelt.

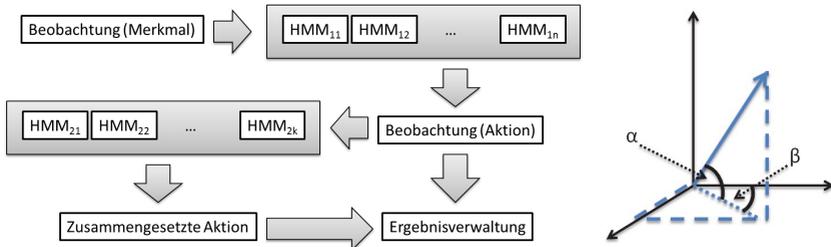
Es haben sich auf Grund verschiedener Anwendungsfälle diverse Abwandlungen von HMMs herausgebildet. Als grundlegendes Modell haben sich diskrete HMMs etabliert, bei denen diskrete, beobachtbare Symbole generiert werden. Im Gegensatz dazu gibt es auch kontinuierliche HMMs, wobei die Symbole folglich kontinuierlich definiert sind. Darüber hinaus gibt es weitere komplexere Markov Modelle [Karg, 2012], wie bspw. *semi Markov Modelle*, wobei eine Zustandshistorie über den aktuellen Zustand entscheidet, oder *faktorielle HMMs*, bei denen Kombinationen von Markov Ketten über die generierten Symbole entscheiden. Diese finden ebenfalls ihre Anwendungen bei der Modellierung menschlicher Bewegungen, bspw. bei [Kulic et al., 2007; Karg et al., 2013]. Jedoch geht deren Anwendung über die Handlungswiedererkennung hinaus. Die Verwendung diskreter HMMs hat sich, wie in Kap. 5.3.1 aufgezeigt, bewährt. Entsprechend wird im Rahmen dieser Arbeit auf diskrete HMMs Bezug genommen.

---

<sup>2</sup>engl.: Versteckt

### 5.3.2.2 Die Erkennungsmethode

Der Zweck der Handlungserkennung ist es, grundlegende menschliche Aktionen zu identifizieren, wie bspw. *gehen* mit der jeweiligen Richtungsangabe *vorwärts*, *rückwärts*, *seitwärts* nach *links* oder *rechts*. Darüber hinaus werden Handlungen wie *stillstehen*, *in die Hocke gehen*, *hinsetzen* und *aufstehen* erkannt. Ebenfalls werden diese Handlungen in modellierten Kombinationen betrachtet. Dazu gehört bspw. *etwas aufheben*, das sich aus *hocken* und *aufstehen* zusammensetzt. Die Kombination des Aufstehens nach dem Hinsetzen wird gleichermaßen behandelt. Entsprechend bilden die HMMs ein zweischichtiges hierarchisches Netz, in dem Erkennungsergebnisse der ersten Schicht als Beobachtungen der zweiten Schicht zur Verfügung stehen (siehe Abb. 5.3 links). Alle erkannten Handlungen werden einem Ergebnismanagementmodul übergeben, das diese abspeichert und bei Bedarf wieder zur Verfügung stellt.



**Abbildung 5.3:** Datenfluss in der zweischichtigen HMM-Architektur (links). Orientierungsabbildung (rechts).

Die Auswahl der passenden Merkmale zur Nutzung als Beobachtungen stellt die Grundlage für eine erfolgreiche Handlungserkennung dar. Basierend auf den berichteten Erfahrungen von Elmezain et al. sind Bewegungsorientierungen am besten geeignet, um Handlungen zu unterscheiden [Elmezain et al., 2008]. Entsprechend wird in dieser Arbeit die dreidimensionale Bewegung des Oberkörpers auf eine Menge diskreter Richtungen abgebildet (siehe Abb. 5.3 rechts). Somit ist die Beobachtung eine Funktion der entsprechenden Winkel:  $O = f(\alpha, \beta)$ , wobei sich die Winkel  $\alpha$  und  $\beta$  aus der Polarkoordinatendarstellung des Bewegungsvektors ergeben.

Um der Körperbewegung und globalen Orientierungswechseln der Bewegungsrichtung Rechnung zu tragen, müssen diese bezüglich eines lokalen Referenzsystems bestimmt werden. Demzufolge muss eine Bewegung nach links die gleichen Beobachtungen erzeugen – unabhängig von Blickrichtung und Ausrichtung des Menschen. Anhand der Kinematikrekonstruktion ist die globale Orientierung des Menschen bekannt und kann somit direkt zur Kompensation herangezogen werden.

Die Kontinuität der Erkennung wird durch die Auswertung aller Modelle in jedem Prozesszyklus erreicht. Um fehlerhafte Erkennungen auf Grund von Ausführung unbekannter oder unvollständiger Handlungen zu vermeiden, muss die höchste Wahrscheinlichkeit aller Modelle über einem Schwellwert liegen. Andernfalls wird die Erkennung einer *unbekannten* Handlung angenommen.

### 5.3.3 Evaluation und Diskussion

Zur experimentellen Analyse der HMM-basierten kontinuierlichen Handlungserkennung wurden verschiedene Handlungsabfolgen von drei verschiedenen Personen durchgeführt, die erfahren und unerfahren mit dem System waren. Die Erkennungsergebnisse wurden abgespeichert (siehe Tabelle 5.1). Die gesamte Länge der ausgeführten Handlungen beträgt 20 Minuten. Die Handlungen wurden in keiner festen Reihenfolge sondern willkürlich ausgeführt. Für die Evaluation wurden ebenfalls die Sensordaten aufgezeichnet und manuell annotiert. Der Vergleich der Erkennungsergebnisse und der annotierten Daten hat die Anzahl *korrekt*, *nicht* und *falsch erkannter* Handlungen ergeben.

Insgesamt wurden 121 Handlungen ausgeführt, wovon 101 richtig erkannt wurden. Nicht erkannt wurden 14 und 6 Handlungen wurden falsch klassifiziert. Demzufolge wurde eine Erkennungsrate von 83,47% erreicht.

Dieser Wert erlaubt eine sichere Erkennung, da die Ergebnisse jedoch direkt als logische Fakten im Schlussfolgerungssystem genutzt werden, kann das System anfällig für Fehler sein. Dieser Umstand ist jedoch auch gegeben, wenn die Erkennungsrate deutlich besser wäre, da anzunehmen ist, dass eine 100%ige Genauigkeit nur eine theoretische Möglichkeit ist.

**Tabelle 5.1:** Beispiel für erzeugte Ausgabe der mehrschichtigen HMM-basierten Handlungserkennung.

1	16-3-3-0	Beobachtungssequenz
2	Aufstehen	Erkannte Handlung
3	0,1225	Wahrscheinlichkeit der Handlung
4	1	Generiertes Symbol
5	Sitzen zu stehen	Erkannte zusammengesetzte Handlung
6	0,9704	Wahrscheinlichkeit zusammengesetzter Handlung

Folglich bedarf es einer Instanz, die fehlerhafte Ergebnisse bemerkt, wie es im Abschnitt 6.2 behandelt wird.

## 5.4 Situationsverstehen durch Schlussfolgerung

Um Situationsverstehen erreichen zu können, muss allgemeines Wissen über Handlungen, Orte, deren semantischen Beziehungen und zeitliche Abhängigkeiten modelliert werden. Mithilfe dieses Wissens und den Erkennungsergebnissen kann über Situationen gefolgert und Erwartungshaltungen angenommen werden. Als Konsequenz können explizite und aufwändige Interaktionsprozeduren vermieden werden.

### 5.4.1 Stand der Forschung

In der Arbeit von [Shi et al., 2004] wird argumentiert, dass HMMs nicht gut geeignet sind, um parallel stattfindende Aktivitäten zu erkennen. Daher führen sie sogenannte *Ausbreitungsnetze* ein. In solchen Netzen wird jeder Knoten mit einem Aktionsprimitiv assoziiert. Die Primitive enthalten probabilistische Modelle über die Dauer der Handlungen. Um temporale und logische Beschränkungen einzuhalten, werden bedingte Verbundwahrscheinlichkeiten eingesetzt. Analog zur Auswertung von HMMs werden mehrere

Ausbreitungsnetze evaluiert, um die Beobachtungswahrscheinlichkeiten zu approximieren.

Im Gegensatz dazu wird in der Arbeit von [Minnen et al., 2003] argumentiert, dass rein probabilistische Methoden für die Erkennung ausgedehnter Handlungen nicht ausreichend geeignet seien. Als Konsequenz wird ein Ansatz basierend auf parametrisierten stochastischen Grammatiken entworfen.

Die Anwendung von wissensbasierten Methoden zur Aktionserkennung wird kaum untersucht. Jedoch wurden Arbeiten zu Szeneninterpretation mittels Beschreibungslogiken (DLs<sup>3</sup>) durchgeführt. DLs werden bspw. von [Hummel et al., 2007] eingesetzt, um Schlussfolgerungen über Verkehrssituationen anzustellen und Straßenkreuzungen zu verstehen. Zum einen werden für die Reduktion des Hypothesenraumes Inferenzdienste genutzt. Zum anderen werden diese Dienste auch genutzt, um dem Fahrer nützliche Informationen zur Verfügung zu stellen.

Die Anwendung von Wissensverarbeitung im Zusammenhang mit Robotern wird durch [Tenorth and Beetz, 2009] untersucht. Dabei wird im Speziellen die Nutzung persönlicher Roboter behandelt. Für die Wissensverarbeitung kommt als Grundlage Prolog zum Einsatz. Die Wissensrepräsentation basiert auf DLs und wird mittels eines *Web Ontology Language (OWL)* Prolog-Plugin verarbeitet. Dieser Ansatz wird genutzt, um Anfragen über das Umgebungsmodell zu stellen. Da Aktionen und Ereignisse durch separate Module erkannt werden, werden diese als Wissensfakten in die Wissensbasis aufgenommen und nicht durch Schlussfolgerungen erkannt. Die Folgerungsmechanismen in Prolog werden auch nicht verwendet, um Rückschlüsse auf Aktivitäten und Situationen zu ziehen.

DLs werden ebenfalls in der Arbeit von [Neumann and Möller, 2008] für Szeneninterpretationen eingesetzt. Räumliche und temporale Beziehungen zwischen visuell zusammengefassten Konzepten bilden die Grundlage für die Analyse und Interpretation von gedeckten Tischen. Der vorgeschlagenen Interpretationsansatz stellt einen schrittweisen Prozess dar, der visuelle Eindrücke und Kontextinformationen ausnutzt. Probabilistische Information wird in dem wissensbasierten Rahmenwerk verwendet, um bevorzugte Interpretationen zu generieren. Dieser Ansatz wurde zu einem

---

<sup>3</sup>engl. Description Logics

allgemeinen Interpretationsrahmenwerk weiterentwickelt, das basierend auf DLs mit allgemeinen Multimediadaten zurechtkommt [Möller and Neumann, 2008].

Ein umfassender Ansatz zum Situationsverstehen wird durch [Springer et al., 2010] vorgestellt. Das System beinhaltet Kontexterfassung, Kontextabstraktion und Entscheidungsfindung in einem Rahmenwerk. Verschiedene Sensoren und Folgerungskomponenten werden durch das Rahmenwerk verwaltet. Entsprechend können verschiedene Folgerungsdienste, darunter auch DLs, für die Entscheidungsfindung eingesetzt werden.

Diese letzten Beispiele und die eigene vorliegende Arbeit zeigen, dass die Verwendung von Beschreibungslogiken großes Potential offenbart. Daher wurde dieser Ansatz für das Situationsverstehen im Rahmen dieser Arbeit untersucht.

### 5.4.2 Modellierte Wissensbasis und Beschreibungslogiken

Im Rahmen dieser Arbeit werden Beschreibungslogiken eingesetzt, um Wissen über Situationen, Handlungen und Zielsetzungen zu formalisieren. Im Folgenden werden Beschreibungslogiken kurz eingeführt und anschließend die modellierte Wissensbasis erläutert.

#### 5.4.2.1 Beschreibungslogiken

Beschreibungslogik ist ein zwei-Variablen-Fragment der Logik erster Ordnung und die meisten Beschreibungslogiken sind entscheidbar. Folglich gibt es robuste, vollständige und terminierende Schlussfolgerungsalgorithmen. Entsprechend wurden im Lauf der Zeit verschiedene effiziente Algorithmen entwickelt und als Teil diverser Folgerungssysteme implementiert. Für eine umfassende Einführung zu den theoretischen Aspekten sowie der Anwendung von Beschreibungslogiken wird auf die Arbeit von [Baader et al., 2010] verwiesen.

Eine beschreibungslogische Wissensbasis ist klar unterschieden in *Allgemeinwissen* und *Wissen über Individuen*, wobei jeweils ein eindeutiger Bezug zu einer Domäne gegeben ist. Somit ist ebenfalls das abgebildete

Allgemeinwissen auf die Domäne begrenzt und definiert ihre Terminologie. Die entsprechenden Axiome sind in der sogenannten *Terminologiebox* angegeben, welche auch *TBox* genannt wird. Die Aussagen über Individuen werden in der *Aussagenbox*, oder *ABox*, zusammengefasst. Hummel et al. zufolge erlaubt diese Trennung modulare und wiederverwendbare Wissensbasen, wodurch eine effizientere Wissensrepräsentation möglich ist [Hummel et al., 2007].

Hummel et al. argumentieren auch, dass die sogenannte *Offene-Welt-Annahme* von Beschreibungslogiken das Verarbeiten unvollständiger Information unterstützt, was für Folgerungen mit Bezug auf Sensordaten essentiell ist.

#### 5.4.2.2 Wissensbasis

Die Wissensbasis definiert Konzepte für Situationen, Handlungen, Zielsetzungen, Orte sowie Intentionen und modelliert inter-konzeptionelle Beziehungen. Hinzu kommen Konzepte, die Aussagen über Menschen und Roboter ermöglichen. Die entsprechenden Konzeptbezeichner und Relationen sind in Abbildung 5.4 als ER-Modell aufgeführt und wurden in Teilen in [Puls et al., 2012b] vorgestellt.

Bei den Situationen werden in erster Linie zwei Unterscheidungen betrachtet: mit und ohne Menschen. Dabei wird das Attribut *nrMenschen* des Konzeptes *Situation* ausgewertet. Somit lässt sich bspw. das Konzept *kein Mensch* folgendermaßen definieren:

$$\text{kein\_Mensch} = \text{Situation} \cap \forall =_0.\text{nrMenschen}$$

Jedoch sind Situationen mit menschlichem Handlungspartner von größerer Bedeutung, so dass diese weiter untergliedert sind. Im Wesentlichen können die Situationskonzepte folgendermaßen beschrieben werden:

**Beobachten** Situationen, in denen der Mensch den Roboter bei der Arbeit beobachtet, bspw. wie es bei Qualitätssicherung zu erwarten ist.

**Vorübergehen** Der Mensch bewegt sich durch den Arbeitsraum, greift jedoch nicht in den Prozess ein.

## 5.4 Situationsverstehen durch Schlussfolgerung

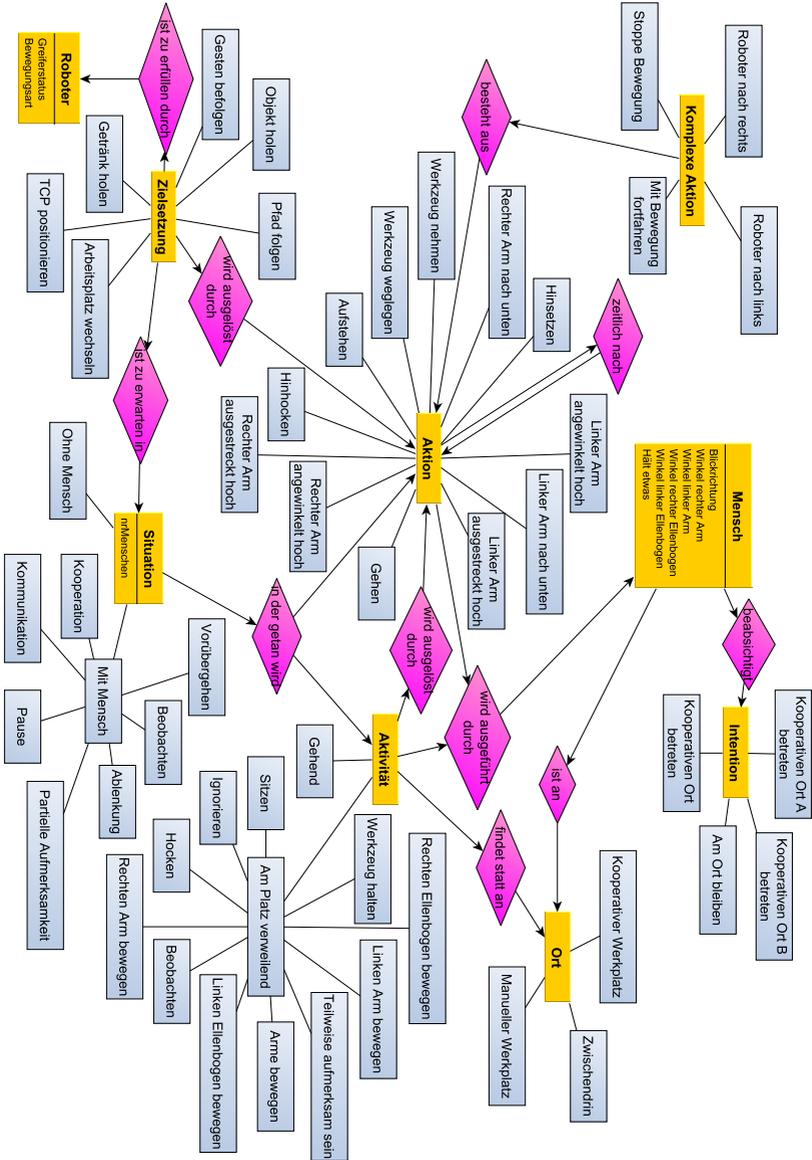


Abbildung 5.4: Modellerte Wissensbasis als ER-Modell.

**Ablenkung** Der Mensch beachtet den Roboter nicht. Er kann mit seiner eigenen Aufgabe beschäftigt sein, oder durch äußere Ereignisse abgelenkt sein.

**Partielle Aufmerksamkeit** Der Mensch ist dem Roboter soweit zugewandt, dass anzunehmen ist, dass der Mensch den Roboter zumindest peripher wahrnimmt.

**Kommunikation** Der Mensch gestikuliert, wodurch offensichtliche Kommunikation vom Menschen ausgeht.

**Kooperation** Situationen, in denen eine Kooperation zwischen Mensch und Roboter angenommen wird.

**Pause** Der Mensch hat sich für eine Pause hingesetzt, befindet sich jedoch noch im Arbeitsbereich. Zugrunde gelegt ist die Annahme, dass produzierende Prozesse im Stehen durchgeführt werden.

In den Situationen können *Aktionen* und *Aktivitäten* durch den Menschen ausgeführt werden. Aktionen bezeichnen kurzzeitige Handlungen, während Aktivitäten andauernde Handlungen darstellen. So ist die Handlung *hin-hocken* eine Aktion, wobei *hocken* eine Aktivität beschreibt. Aktionen und Aktivitäten sind durch die Relation *wird ausgelöst durch* verknüpft. Die Aktivität *hocken* wird somit durch das Ereignis *hinhocken* ableitbar.

Um zeitliche Bezüge zwischen Aktionen zu realisieren, sind diese über die Relation *zeitlich nach* miteinander verbunden. Eine Instanz des Konzeptes *Werkzeug weglegen* kann nur erzeugt werden, wenn eine Instanz des Konzeptes *Werkzeug nehmen* zuvor erstellt wurde. Andernfalls führt die Verarbeitung der ABox zu einer inkonsistenten Wissensbasis und somit zu einem Fehler. Darüber hinaus können Aktionen zu sogenannten *komplexen Aktionen* zusammengesetzt werden, wobei diese aus mindestens zwei anderen Aktionen eine Einheit bilden. Zur Definition wird eine entsprechende minimale Kardinalität angegeben:

`komplexe_Aktion = Aktion  $\cap$   $\exists_{>2}$ besteht_aus.Aktion`

Aktionen stellen auf diese Weise atomare Konzepte dar. Die Konzepte *Werkzeug nehmen* und *Werkzeug weglegen* werden als solche atomaren Konzepte behandelt, da sie auf die Auswertung des Attributes *Werkzeug halten* des Konzeptes *Mensch* zurückzuführen sind. Die modellierten komplexen

Aktionen stellen im Rahmen dieser Arbeit *Gesten* dar, so dass die Bewegung des Roboters durch Armbewegungen gesteuert werden kann. Damit jedoch Gesten auch als solche interpretiert werden und korrespondierende Verfahrbefehle umgesetzt werden können, muss zuvor die Aktion *stoppe Bewegung* vom Menschen ausgeführt worden sein. Durch die Ableitung komplexer Aktionen vom Konzept *Aktion* wird die temporale Beziehung übernommen.

Durch Aktionen können Erwartungen des Menschen an das Robotersystem ausgelöst werden, die als Zielsetzungen modelliert sind [Puls et al., 2013a]. Damit Zielsetzungen auch abgeleitet werden können, ohne eine auslösende Aktion feststellen zu müssen, ist bspw. das Situationskonzept *Kooperation* durch Implikation mit dem Konzept *Zielsetzung* verknüpft. Zielsetzungen sind ebenfalls direkt mit den Situationen verknüpft, in denen diese tatsächlich zu erwarten sind. Zielsetzungen sind darüber hinaus auf einen Roboter eingeschränkt, der diese erfüllen kann. Durch die Auswertung der Roboterattribute *Greiferstatus* und *Bewegungsart* können die Möglichkeiten eingegrenzt werden:

```
TCP_positionieren = Zielsetzung
    ∩ ∇ ist_zu_erfüllen_durch.(X
        ∩ ∇ Greiferstatus.Greifend
        ∩ ∇ Bewegungsart.Gestoppt)
    ∩ ∇ wird ausgelöst durch.Nehme_Werkzeug
    ∩ ∇ ist_zu_erwarten_in.Kooperation
```

Die Aktivitäten sind durch die Relation *findet statt an* mit den möglichen Orten des Arbeitsraumes verknüpft. Ebenfalls verbindet die Relation *ist an* den Menschen mit seinem Aufenthaltsort. Die Konzepte *Mensch* und *Intention* sind wiederum durch die Beziehung *beabsichtigt* verbunden.

### 5.4.3 Das Schlussfolgerungssystem

Das Folgerungsmodul hat zur Aufgabe, Beobachtungen und Erkennungsergebnisse anderer Systemmodule in die Wissensbasis als Aussagen der ABox zu übertragen und Folgerungsergebnisse weiterzuverarbeiten. Im Folgenden wird auf diese Aufgaben und Folgerungssysteme eingegangen.

### 5.4.3.1 Folgerungssysteme und Schnittstellen

Durch Entwicklungen im Bereich semantischer Netzwerke wurden verschiedene Folgerungssysteme entwickelt, um effiziente Algorithmen zu entwerfen und zu implementieren. Dazu gehören *RacerPro* [Haarslev et al., 2007], *FaCT++* [Tsarkov and Horrocks, 2006] und *Pellet* [Sirin et al., 2007]. Diese Systeme bieten zur Nutzung ihrer Dienste verschiedene Schnittstellen an. Ein erster Ansatz der vorgestellten Arbeit basierte auf dem *Pellet*-System und nutzte die DIG-Schnittstelle, da *Pellet* effizientes Folgern erlaubt [Pan, 2005; Gardiner et al., 2006]. Die DIG-Schnittstelle hat zum einen den Vorteil der Trennung von Applikation und Folgerungssystem bezüglich Programmiersprache und Ausführungsort, da es die Kommunikation mittels TCP und XML Nachrichten bewerkstelligt. Zum anderen ist diese Schnittstelle jedoch veraltet und überholt durch Entwicklungen, die einen größeren Funktionsumfang der Folgerungssysteme zur Verfügung stellen. Dazu gehört bspw. die OWL API [Horridge and Bechhofer, 2009]. Auf Grund der Konzentration der Entwicklung im Bereich semantischer Netze, sind diese Schnittstellen auf Javaimplementierungen ausgelegt. Um effiziente Folgerungsalgorithmen nicht durch zusätzliche Kommunikation und Konvertierungen zu verlangsamen, ist die Implementierung in C++ vorteilhaft.

Das System *FaCT++* basiert auf C++ und konnte direkt in das MAROCO Rahmenwerk eingebettet werden. Wie in [Tsarkov and Horrocks, 2006; Pan, 2005; Gardiner et al., 2006] gezeigt, stellt *FaCT++* effiziente und schnelle Algorithmen bereit. Diese basieren auf dem Tableaux-Kalkül, welche Optimierungstechniken einsetzen, die strukturelle Merkmale typischer Ontologien ausnutzen [Tsarkov and Horrocks, 2006].

### 5.4.3.2 Das Folgerungsmodul

Das Folgerungsmodul hat vorrangig die Aufgaben ein Folgerungssystem zu instanzieren und die Wissensbasis sowie die Folgerungsergebnisse zu verwalten.

**Wissensverwaltung** Die Verwaltung der Wissensbasis folgt einem funktionellen Ansatz, der auch *Sage&Frage*<sup>4</sup> genannt wird [Baader et al., 2010]. Nach dem Anlegen der Wissensbasis – die *Sage*-Operation – können Ergebnisse und Informationen entnommen werden – die *Frage*-Operation. Die Änderung einer existierenden Wissensbasis kann nach einer *Frage*-Operation durch Zurückziehen des entsprechenden Axiome in FaCT++ erreicht werden. Auf diese Weise werden einzelne Axiome als *unbenutzt* markiert und die Wissensbasis als *geändert* gekennzeichnet. In einem anschließenden Prozesszyklus können neue Axiome hinzugefügt werden. Auf Grund der internen Datenhaltungsstrukturen werden zurückgezogene Axiome in FaCT++ nicht gelöscht. Entsprechend können bei wiederholtem Zurückziehen und Neuanlegen Speicheranforderungen steigen. Aus diesem Grund wurde das FaCT++-System im Rahmen dieser Arbeit durch eine Funktionalität zur Axiomaktualisierung mittels *volatiler Axiome* erweitert, wodurch steigender Speicherverbrauch vermieden werden kann.

Die Wissensverwaltung benötigt ein aktuelles Modell der Wissensbasis, die sich aus domänenspezifischen Wissen und Aussagen zu menschlichen Kinematikmodell sowie roboterspezifischen Parametern zusammensetzt. Die Aussagen werden in jedem Zyklus durch die Wissensverwaltung aktualisiert. Die entsprechenden Werte der Parameter werden durch eine Merkmalsextraktion bestimmt (siehe Abb. 3.3). Bezüglich des Konzeptes *Mensch* sind folgende Attribute wichtig:

- Winkel beider Ellenbogen,
- Winkel beider Schultergelenke,
- Blickrichtung des Menschen bezüglich Roboterposition,
- Benutztes Werkzeug.

Die ersten drei Attribute können direkt aus dem Menschmodell und dem Umweltmodell entnommen werden. Da die Benutzung kleinerer Objekte durch die bisherige Sensorausstattung nicht möglich ist, wird der Wert des Attributes *Benutztes Werkzeug* durch direkte Nutzereingabe, bspw. mittels Wii-Remote, festgelegt.

Die Komponente *Roboter* stellt folgende Parameter bereit:

---

<sup>4</sup>engl.: Tell&Ask

**Greiferstatus** Der enthaltene Wert kann *leer* oder *greifend* annehmen.

**Bewegungsart** Der Wert kann sein: *gestoppt*, *vorgegebenem Pfad folgend* oder *gegebener Aufgabe folgend*.

Durch die Betrachtung von Parametern des menschlichen Kinematikmodells wird es möglich Zustandsänderungen durch Folgerung direkt wahrzunehmen, ohne auf eine gesonderte Klassifikationsinstanz zurückgreifen zu müssen. Es ist darüber hinaus möglich, im Rahmen des betrachteten Kooperationssystems einfache Armgesten des Menschen damit zu modellieren und zu klassifizieren. Da jedoch die benötigte Verarbeitungszeit höher ist als bei dedizierten Gestenklassifikationsmethoden (siehe Abschnitt 5.4.5), sind kontinuierliche bzw. schnelle Bewegungen nur begrenzt modellierbar.

**Kombination mit unscharfer Logik** In einem ersten Ansatz wurden die extrahierten Werte der Merkmale auf scharfe Mengen abgebildet. Die Wissensbasis wurde dann mit korrespondierenden Zeichenketten befüllt, die vergleichende Operationen während der Schlussfolgerung erlauben [Puls et al., 2012b]. Der aktuelle Ansatz kombiniert unscharfe Logiken mit dem beschreibungslogischen System FaCT++.

Die Kombination von Unschärfe und Beschreibungslogiken kann auf verschiedene Arten erreicht werden. In der Arbeit von [Lukasiewicz and Straccia, 2006; Lukasiewicz and Straccia, 2008] wird die Darstellung von Unschärfe in Form von Wahrscheinlichkeiten und unscharfen Mengen behandelt, wobei entsprechende Syntax und Semantik eingeführt wird. Die Kombination von unscharfer Logik mit Beschreibungslogiken in einem Folgerungssystem wurde durch Bobillo et al. untersucht, wobei unscharfe Konzepte und Relationen in einem Konvertierungsschritt auf scharfe Konzepte und Relationen abgebildet wurden [Bobillo et al., 2008]. In der Arbeit von Stoilos et al. wird für eine unscharfe Beschreibungslogik ein Tableaux-Algorithmus vorgestellt, der eine Konvertierung vermeidet, jedoch in der Ausdrucksmöglichkeit beschränkt ist [Stoilos et al., 2008].

Im Rahmen der vorliegenden Arbeit wird der Folgerungsprozess durch die Kombination mit unscharfer Logik zu einem zweistufigen Prozess. Mittels einer Regelbasis der unscharfen Logik werden Eingabewerte der Merkmalsextraktion klassifiziert und können direkt als Konzeptdefinitionen

im beschreibungslogischen Teil ausgewertet werden. Somit werden in der ersten Stufe die Regeln der unscharfen Logik ausgewertet, so dass der jeweilige Typ einer Merkmalsinstanz festgelegt wird. Der Typ ist äquivalent zum speziellsten anwendbaren Konzept in der Konzepthierarchie korrespondierend mit der Instanz. Die zweite Stufe des Folgerungsprozesses nutzt diese Typfestlegungen aus, um den Hauptteil der Wissensbasis zu klassifizieren. An den Stellen, wo Bezug auf die unscharfen Konzepte genommen wird, wird direkt der scharfe, konkrete Typ verwendet.

**Einbindung externer Erkennungsergebnisse** Externe Erkennungsergebnisse beinhalten:

- menschliche Handlungen, die mittels HMMs klassifiziert wurden,
- Aufenthaltsort des Menschen,
- Bewegungsabsicht des Menschen,
- durch den Menschen benutztes Werkzeug.

Wie bereits beschrieben, wird das benutzte Werkzeug als Attribut des Menschen behandelt und durch die Merkmalsextraktion bereitgestellt. Die anderen Ergebnisse werden durch die Verwendung von sogenannten Verbindungskonzepten in die Wissensbasis als Instanzen eingetragen und stehen auf diese Weise während der Klassifikation der Wissensbasis zur weiteren Verwertung zur Verfügung.

**Verwaltung von Ergebnissen** Damit die modellierten Relationen zwischen Konzepten mit entsprechenden Instanzen, auch Individuen genannt, verknüpft werden können, bedarf es der Ergebnisverwaltung. Speziell bei Handlungen, die eine zeitliche Beziehung zueinander haben und aus anderen Handlungen zusammengesetzt sein können, ist diese Verwaltung essenziell. Vergangene Aktionen müssen in jedem Prozesszyklus in die Wissensbasis eingetragen werden und mit anderen Aktionen verknüpft werden. Demzufolge werden auch die Ergebnisse externer Module in der Verwaltung berücksichtigt.

Um den zeitlichen Horizont der Verwaltung zu begrenzen werden die Folgerungsergebnisse untersucht, wobei auf spezielle Ergebniszustände

geachtet wird. Verlässt der Mensch den Arbeitsraum wird eine zeitliche Unabhängigkeit der zukünftigen Ereignisse von bisherigen Beobachtungen angenommen. Entsprechend werden veraltete Ergebnisse durch die Verwaltung entfernt. Gleiches gilt für das Auftreten zusammengesetzter Handlungen, die in erster Linie nach dem Erkennen der Geste *stoppe Bewegung* und vor der Erkennung *führe Bewegung fort* relevant sind. Folglich können andernfalls korrespondierende Ergebnisse durch die Verwaltung verworfen werden.

#### 5.4.4 Untersuchung zur Optimierung durch inkrementelle Folgerung

In der Anwendung zur zyklischen Auswertung von Beobachtungen und Ereignissen in der MRK ist es wünschenswert, dass Schlussfolgerungen schnellst möglich durchgeführt werden. Dabei liegt die Idee nahe, dass hauptsächlich Veränderungen betrachtet werden anstatt wiederholt die gesamte Wissensbasis mit vergangenen und aktuellen Informationen füllen und auszuwerten. Entsprechend wurde im Rahmen dieser Arbeit untersucht, ob inkrementelle Folgerung einen Nutzen bringen kann.

In der Arbeit von Grau et al. wurde eine Methode zur inkrementellen Folgerung vorgestellt, die auf Modularisierung der Wissensbasis basiert. Dabei wird argumentiert, dass ein beliebiges Folgerungssystem ohne Änderungen der internen Algorithmen verwendet werden kann, da die Wissensbasis selbst angepasst wird [Grau et al., 2010]. Ein *Modul für ein Axiom* wird von Grau et al. folgendermaßen definiert:

**Definition 5.1** Sei  $O$  eine Ontologie und  $O_1 \subseteq O$  eine Teilmenge von Axiomen von  $O$ .  $O_1$  wird Modul für ein Axiom  $\alpha$  genannt, falls gilt:  $O_1 \models \alpha \Leftrightarrow O \models \alpha$ .

Demzufolge ist jede Übermenge von  $O_1$  ein Modul für  $\alpha$ , so auch  $O$  selbst. Jedes Modul enthält mindestens eine *Begründung* für  $\alpha$ , wobei *Begründung* als minimale Axiomenmenge definiert wird, aus der  $\alpha$  gefolgert werden kann. Jede Begründung ist somit ein *minimales Modul* für  $\alpha$ . Entsprechend genügt das Wissen über alle Begründungen in  $O$  für  $\alpha$ , um alle Module für  $\alpha$  zu bestimmen. Die Idee dabei ist, dass für die Folgerung der gesamten

Wissensbasis diejenigen Module nicht betrachtet werden müssen, deren Axiome in den Änderungen nicht beteiligt sind.

Grau et al. schlagen einen Algorithmus vor, um möglichst kleine Module zu bestimmen – die Bestimmung von minimalen Modulen ist sehr rechenintensiv und kann auf das Problem der Wissensbasisklassifizierung zurückgeführt werden. Im Rahmen dieser Arbeit wurde der Algorithmus zur Modulbestimmung in das FaCT++-System integriert. Resultierend kann festgestellt werden, dass die ermittelten Module sehr ungleich verteilt sind. Insgesamt enthält die Wissensbasis 136 Axiome. Die Modularisierung teilt diese in 61 Module auf, wovon 51 Module identisch sind und 131 Axiome umfassen. Es wurden sieben Module identifiziert, die jeweils ein unterschiedliches Axiom enthalten. Drei Module enthielten die leere Menge. Folglich ist die entwickelte Wissensbasis intern stark zusammenhängend, so dass Änderungen eines Axioms unterschiedliche Ergebnisse der Auswertung fast aller anderen Axiome nach sich ziehen kann.

Eine inkrementelle Folgerung durch Modularisierung der Wissensbasis ist für den vorgestellten Anwendungsfall somit kaum sinnvoll durchführbar. Andere Möglichkeiten der inkrementellen Verarbeitung von Ontologien sind denkbar, bspw. [Parsia et al., 2006; Halaschek-Wiener and Hendler, 2007], jedoch sind damit grundlegende Eingriffe in die Ontologieverarbeitung und Folgerungsalgorithmen verbunden.

### 5.4.5 Evaluation und Diskussion

Für die Analyse des Folgerungsmoduls wurden verschiedene Handlungsabläufe ausgeführt und aufgezeichnet. Um eine effiziente Analyse zu ermöglichen, wurde darüber hinaus wurde eine Methode zur automatischen Merkmalswertvorgabe implementiert. Die Analyse stützt sich somit insgesamt auf die Wertvorgabe und der Verarbeitung tatsächlicher Sensorwerte. Entsprechend können natürliche Handlungen betrachtet werden und Übergänge zwischen Aktionen sowie spezielle Nutzfälle untersucht werden. Dabei spielt die Verarbeitungsdauer eine wesentliche Rolle, da es sich um die Anwendung während der Mensch-Roboter Kooperation handelt. Zeitliche Aspekte können hierbei Aussagen über Verwendbarkeit des Systems ermöglichen.

### 5.4.5.1 Ergebnisse

In Tabelle 5.2 sind die Ergebnisse der zeitlichen Betrachtung von 2830 Zyklen aufgeführt, wobei die Merkmalswertvorgabe genutzt wurde. Die durchschnittliche Verarbeitungsdauer liegt bei etwa 263 ms. Die untere Grenze wird durch 54 ms dargestellt. In seltenen Fällen werden bis zu 1,2 Sekunden benötigt, wobei der Anteil der Zyklen über einer Sekunde 3,53 % erreicht. Fast alle dieser Ausreißer benötigen zwischen 1,2 und 1,22 Sekunden.

**Tabelle 5.2:** Ergebnisse zur Verarbeitungsdauer bei Verwendung der Merkmalsvorgabe.

# Zyklen	2830	# > 500 ms	441 (15,58 %)
⊙ Antwortzeit [ms]	263,19	# > 1000 ms	100 (3,53 %)
Minimum [ms]	54	# > 1200 ms	100 (3,53 %)
Maximum [ms]	1221	# > 1220 ms	1 (0,03 %)
Standardabweichung [ms]	284,17		

**Tabelle 5.3:** Ergebnisse zur Verarbeitungsdauer bei Verwendung von Sensordaten.

# Zyklen	2680	# > 500 ms	381 (13,46 %)
⊙ Antwortzeit [ms]	236,68	# > 1000 ms	112 (3,96 %)
Minimum [ms]	37	# > 1200 ms	79 (2,79 %)
Maximum [ms]	1543	# > 1220 ms	41 (1,45 %)
Standardabweichung [ms]	320,24		

In Tabelle 5.3 sind die entsprechenden Ergebnisse unter Verwendung tatsächlicher Sensordaten aufgeführt. Aufgezeichnet wurden 2680 Verarbeitungszyklen mit einer durchschnittlichen Verarbeitungszeit von etwa 237 ms, was geringer ist als bei Merkmalswertvorgabe und eine durchschnittli-

che Verarbeitungsrate von 4,2 Hz ergibt. Jedoch sind die Ergebnisse zu Ausreißern und Standardabweichung schlechter.

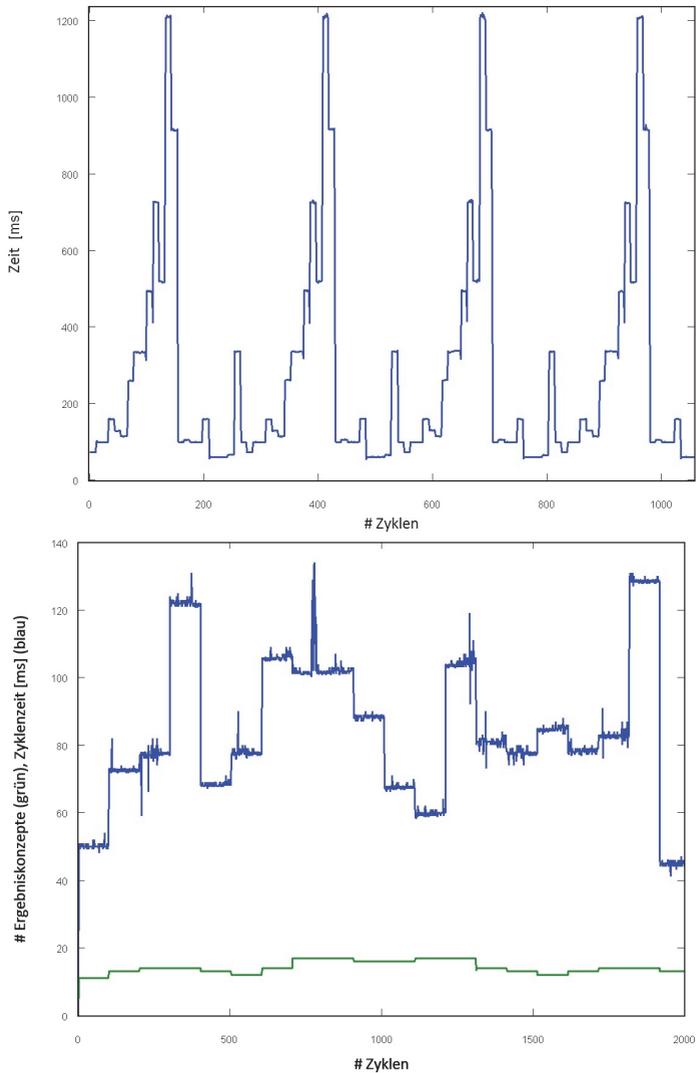
In Abbildung 5.5 oben ist die Laufzeitanalyse einiger Verarbeitungszyklen aufgeführt, wobei Merkmalswertvorgaben genutzt wurden. Der wiederholende Charakter der Vorgaben kann klar erkannt werden. Ebenfalls zu erkennen ist die Systematik und Wiederholbarkeit der Ausreißer, was auf eine Abhängigkeit von Merkmalswerten hindeutet. In Abbildung 5.5 unten ist die Zyklenverarbeitungszeit mit der korrespondierenden Anzahl der Ergebniskonzepten abgetragen. Es ist ersichtlich, dass die Anzahl der abgeleiteten Konzepten nicht direkt mit der Verarbeitungszeit in Verbindung gebracht werden kann.

### 5.4.5.2 Diskussion

Die Fähigkeiten des vorgestellten Ansatzes gehen über eine einfache Situationserkennung hinaus, da Folgerungen ausgenutzt werden, um direkt Zielsetzungen und somit Erwartungen an das technische System abzuleiten. Dazu werden die Informationen verschiedener externer Erkennungsmodule mit Beobachtungen über das menschliche Kinematikmodell und Roboterparameter fusioniert.

Die erreichte Verarbeitungsfrequenz von etwa 4 Hz ermöglicht zeitnahe Situationsanalysen und Verständnisentwicklung für Vorgänge im Arbeitsbereich, wobei die Verarbeitungszeit von etwa 250 ms keine sichere Kooperation allein basierend auf dem Folgerungssystem erlaubt. Entsprechend ist ein hierarchischer Systemaufbau notwendig, der die höheren kognitiven Funktionen mit sicherheitsrelevanten Modulen koppelt. Auf diese Weise können Sicherheitsbetrachtungen im Gesamtsystem durchgesetzt und durch höhere kognitive Funktionen gelenkt werden. Ein solcher hierarchischer Ansatz wird in dieser Arbeit mit den in Kapitel 6 vorgestellten Methoden verfolgt.

Mit dem vorgestellten Ansatz zur Situationsanalyse können Bewegungen des Menschen direkt erfasst werden, jedoch ist anzunehmen, dass geübte Werker Prozessschritte sehr schnell ausführen können. Betrachtungen zu solchen Aspekten sind jedoch nach bestem Wissen nicht verfügbar. Es ist trotzdem zu erwarten, dass weitere Beschleunigungen nötig sind.



**Abbildung 5.5:** Oben: Laufzeitanalyse mit wiederkehrenden Merkmalswertvorgaben. Unten: Der untere Verlauf (grün) gibt die Anzahl der Ergebniskonzepte. Die obere Linie (blau) zeigt die entsprechende Zyklusverarbeitungszeit. Merkmalswertvorgabe wurde genutzt.

Die untersuchte Optimierung zur inkrementellen Folgerung hat nicht die gewünschten Effekte erzielen können. Insbesondere der Grad der Verknüpfungen der modellierten Konzepte erlaubt keine Modularisierung der Wissensbasis. Dieser Umstand kann sich bei Betrachtung komplexerer Sachverhalte und Prozesse ändern, jedoch ist zu erwarten, dass Produktionsprozesse nur unter Einführung von Konzeptredundanzen aufgetrennt werden können.

## 5.5 Ermittlung von Handlungszielen für das Robotersystem

Im Bereich der industriellen Mensch-Roboter-Kooperation ist das allgemeine Handlungsziel dadurch gegeben, dass Mensch und Maschine gemeinschaftlich Produkte herstellen, so dass der Roboter den Menschen unterstützt und körperliche Anstrengungen für den Menschen reduziert wird. Um eine sinnvolle Kooperation zu erreichen sind diese Vorgaben jedoch mehrdeutig und nicht ausreichend. Auf Grund dessen, dass die industrielle Anwendung der MRK eine indirekte Form der Kooperation darstellt, kann der Roboter nicht einfach Aktionen des Menschen kopieren (siehe Kap. 5.1). Stattdessen komplementieren sich die Handlungen von Mensch und Roboter. Daher wird in dem zuvor vorgestellten System zur Situationsanalyse die Aufgabenzielsetzung durch das wissensbasierte System modelliert und bestimmt.

Im Folgenden wird die Methode zur Umsetzung dieser allgemeinen Zielsetzung in einzelne Handlungsziele vorgestellt, damit diese durch den Roboter abgearbeitet werden können.

### 5.5.1 Reaktive Handlungsplanung

Die Themen der automatisierten Aufgaben- und Ausführungsplanung sind aktive und populäre Forschungsbereiche [Ghallab et al., 2004]. Entsprechend wurden verschiedene Methoden entwickelt, um gegebene komplexe Aufgaben zu erreichen. Die Anwendung des Situationskalküls ist eine dieser Methoden, die im Bereich der Robotik eingesetzt wurde, bspw. in

der Form der *GOLOG* Aktionssprache [Ferrein et al., 2005]. Weitere Forschung und Entwicklung hat den Einsatz logikbasierter Robotersteuerung in dynamischen Domänen ermöglicht, wie es im Bereich des Roboterfußballs demonstriert wurde [Ferrein and Lakemeyer, 2008; Ferrein, 2010]. Demzufolge zeigt die Anwendung des Situationskalküls gute Ergebnisse und Potential für den Einsatz im Bereich der industriellen Robotik. Entsprechend wurde dieser Ansatz zur Handlungsplanung im Rahmen dieser Arbeit ausgewählt und untersucht [Puls et al., 2013a].

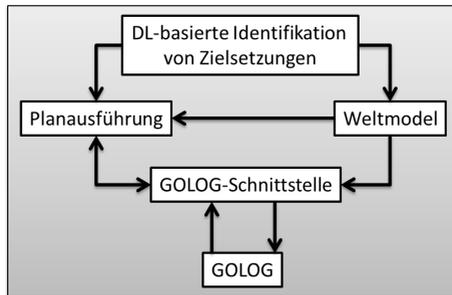
Das Situationskalkül wurde erstmals von McCarthy 1963 vorgestellt [McCarthy, 1963]. Es ist eine Logik zweiter Ordnung und wird durch Terme erster Ordnung charakterisiert. Diese Terme sind Aktionen  $a$  und Situationen  $s$ . Situationen sind axiomatische Zustandsbeschreibungen des Weltmodells. Aktionen repräsentieren die Übergänge zwischen Situationen. Eine detaillierte Einführung zum Situationskalkül wird in [Reiter, 2001] gegeben. Die *GOLOG* Aktionssprache basiert auf dem Situationskalkül. Die im Rahmen dieser Arbeit untersuchte Methode zur Handlungsplanung geht auf die Resultate von Hähnel zurück, der ein *GOLOG*-basiertes Laufzeitsystem als Planungssystem vorstellte [Hähnel, 1998].

In Abbildung 5.6 sind die beteiligten Komponenten schematisch dargestellt. Die identifizierte Zielsetzung wird im Weltmodell abgelegt, damit andere Komponenten darauf Zugriff haben. Die *GOLOG*-Schnittstelle übersetzt die Informationen des Weltmodells zu *GOLOG*-Aussagen, die eine symbolische Repräsentation der Orte und situationsabhängigen Parametern benutzt. Die Zielsetzungen werden dabei direkt als Teil des Zustands des Menschen formuliert. Entsprechend formuliert die folgende Aussage, dass der Mensch mehr Material benötigt, wobei er sich an Position 1 in der Situation  $s_0$  befindet:

`mensch(Material, 1,  $s_0$ )`

Falls kein Mensch präsent ist oder dieser mit eigenen manuellen Aufgaben beschäftigt ist, wird als Zielsetzung angenommen, dass der Roboter einem vordefinierten Plan folgen soll.

Die *GOLOG*-Schnittstelle führt eine Vorverarbeitung der resultierenden Pläne durch, indem die Ergebniszeichenkette in symbolische Ausführungsschritte aufgeteilt wird. Ein Beispiel eines resultierenden Planes kann folgendermaßen angegeben werden:



**Abbildung 5.6:** Schematische Darstellung der Komponenten des Handlungsplanungssystems.

`verfahren(7), aufnehmen(Werkstück),  
verfahren(2), ablegen(Werkstück)`

Die Planausführungskomponente übersetzt jeden dieser Schritte in eine Anweisungsfolge, die aus Bewegungs- und Greifvorgaben zusammengesetzt ist. Symbolische Informationen, wie Orte oder Objektbeschreibungen, werden auf korrespondierende Koordinaten oder Objektreferenzen abgebildet. In manchen Fällen, kann diese Abbildung durch eine tabellarische Zuordnung erfolgen. In Fällen von Vorgaben, die von der menschlichen Position abhängen, wie bspw. Objektübergabe, wird Szenenwissen und die rekonstruierte menschliche Kinematik verwendet, um entsprechende Roboterkonfigurationen zu bestimmen.

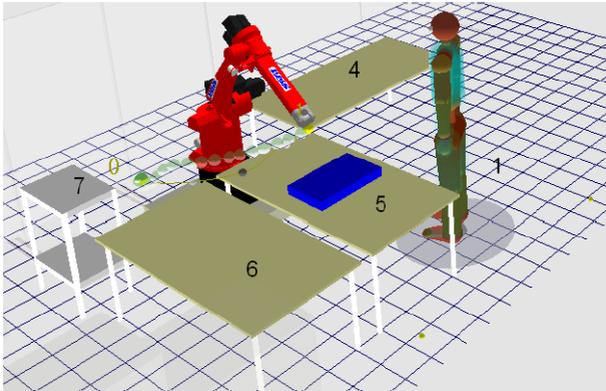
Um dynamische Situationen zu behandeln und dynamisches Planungsverhalten zu erreichen, werden erwartete Weltmodelländerungen mit beobachteten verglichen. Falls Unterschiede auffallen, wird durch die GOLOG-Schnittstelle eine erneute Planung mit geändertem Weltmodell angestoßen. Eine Neuplanung wird ebenfalls ausgeführt, falls eine geänderte Zielsetzung vom System gefolgert wird.

## 5.5.2 Evaluation und Diskussion

Um das System zur Handlungsplanung und Planausführung zu untersuchen, wurden verschiedene Ansätze gewählt. Einerseits wurden a-priori

Weltmodelle definiert, um die Plausibilität und Umsetzbarkeit der Planungsergebnisse zu überprüfen. Andererseits wurde ein Prozess simuliert, um die dynamischen Aspekte der Planausführung zu untersuchen.

Für die Untersuchungen wurden verschiedene Orte und Bereiche im Arbeitsraum definiert. Diese sind in Abbildung 5.7 dargestellt. Das Szenario beinhaltet die vordefinierte Aufgabe für den Roboter, Werkstücke von Ort 4 in die Box an Ort 5 zu transportieren. Falls der Mensch weiteres Material benötigt, soll der Roboter entsprechende Objekte an Ort 7 aufnehmen und zum Werker verfahren. Für den Fall, dass eine Box gefüllt ist, soll die volle Box an Ort 6 transportiert werden, so dass die darunterliegende Box genutzt werden kann.



**Abbildung 5.7:** Darstellung des Arbeitsraumes. Die verschiedenen symbolischen Orte sind durch Nummern markiert.

Um die Plausibilität der Planung zu untersuchen, wurden Weltmodelle vordefiniert. Die korrespondierenden Pläne wurden manuell auf Konsistenz und Umsetzbarkeit analysiert und überprüft. Einige Beispiele dieser Weltmodelle mit entsprechenden Plänen sind in Tabelle 5.4 aufgeführt. Diese zeigen, dass einfache und komplexe Kombinationen von Aktionen zur Zielerfüllung erstellt werden können. Das zweite Beispiel in Tabelle 5.4 zeigt gleichermaßen die Zielannahme eines vordefinierten Prozesses, da keine andere Zielsetzung durch den Menschen vorgegeben wird.

Diese Ergebnisse zeigen das erwartete statische Verhalten der GOLOG-basierten Planung. Um die dynamischen Aspekte zu beleuchten, wurden Sensordaten durch das Gesamtsystem ausgewertet. Somit wurde ein Produktionsprozess wie beschrieben durch menschliche Interaktionen mit dem System simuliert. In Tabelle 5.5 ist ein Auszug des aufgezeichneten Protokolls aufgeführt.

**Tabelle 5.4:** Beispiele für Weltmodelle mit entsprechenden Plänen.

Weltmodell:	<code>mensch(Material, 1, s0), tcp(Objekt, s0), roboter(1, s0), boxvoll(s0)</code>
Plan:	<code>s0 ablegen(Objekt)</code>
Erläuterung:	Der Werker benötigt Material. Der Roboter greift ein Objekt und ist nahe des Menschen positioniert. Somit kann der Roboter das Objekt direkt übergeben.
Weltmodell:	<code>mensch(beschäftigt, 1, s0), tcp(leer, s0), roboter(7, s0), boxvoll(s0)</code>
Plan:	<code>s0 verfahren(5) aufnehmen(Box) verfahren(6) ablegen(Box) verfahren(4) aufnehmen(Werkstück) verfahren(5) ablegen(Werkstück)</code>
Erläuterung:	Der Werker ist beschäftigt mit eigener Arbeit. Die Box ist gefüllt und der Roboter hat kein Objekt im Greifer. Entsprechend muss der Roboter die volle Box transportieren, um dann ein Werkstück in die leere Box zu legen.

Die aufgeführten Daten in Tabelle 5.5 enthalten Zeitstempel und korrespondierende Information bezüglich Weltmodell (WM), Plänen und Ausführungsschritten. Wie zu erkennen ist, wird der Plan ein Werkstück zu transportieren aufgegeben, da der Mensch in die Zelle tritt und Material benötigt. Dies verdeutlicht die dynamische Anpassungsfähigkeit von Handlungsplänen.

Zur Analyse des Laufzeitverhaltens wurden zufällige Weltmodelle generiert. Verschiedene Durchläufe erzeugten jeweils Protokoll Daten, die untersucht wurden. In Abbildung 5.8 sind die kumulativen Zeiten für bis zu 30000

**Tabelle 5.5:** Auszug der Protokolldaten zur Verdeutlichung der Plananpassung auf Grund von verändertem Weltmodell.

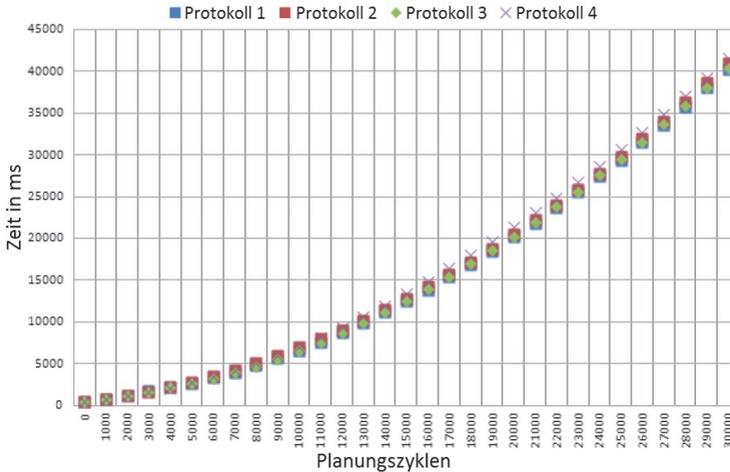
---

39408	WM:	keiner leer 5
39412	Plan:	s0 verfare(4) aufnehmen(Werkstück) verfare(5) ablegen(Werkstück)
39412	Schritt 1:	verfare(4)
45093	fertig.	Check WM -> Unterschied.
45094	WM:	Material leer 4
45148	Plan:	s0 verfare(7) aufnehmen(Objekt) verfahren(1) ablegen(Objekt)

---

Planungszyklen aufgezeigt. Es ist ersichtlich, dass die Relation zwischen Anzahl der Planungszyklen und der Planungszeit nicht linear ist, jedoch kann diese durch eine quadratische Funktion approximiert werden. Für die ersten 5000 Zyklen liegt die durchschnittliche Zeit bei etwa  $0,5ms$ , womit sehr schnelle Zeiten erreicht werden. Bei linearer Mittellung der 30000 Zyklen steigt diese Zeit auf etwa  $1,3ms$ , was weiterhin für schnelle online Planung ausreicht.

Die vorgestellte Methode zur Handlungsplanung bedarf, wie beschrieben, einer Abbildung von symbolischen Orten zu Roboterkonfigurationen. Weiterhin müssen die Orte im GOLOG-System zusätzlich zum Situationsanalysesystem definiert werden. Auf diese Weise werden Änderungen des geometrischen Weltmodells erschwert bzw. die manuelle Änderung birgt eine Fehlerquelle. Eine Möglichkeit, dies zu vermeiden, liegt in der Integration eines Aktionsformalismus und Planungsfunktionalität in das beschreibungslogische System. Ein Ansatz hierzu wurde durch Sirin vorgestellt, indem hierarchische Aufgabennetze mit Beschreibungslogiken gekoppelt wurden [Sirin, 2006]. Jedoch wurde bereits im Abschnitt 5.4.5 gezeigt, dass die Laufzeiten des momentanen Folgerungsmoduls bei etwa 4 Hz liegen. Eine Einführung komplexer Planung in das Folgerungsmodul würde diese Zeit weiter verschlechtern.



**Abbildung 5.8:** Kumulative Laufzeiten für Planungszyklen.

## 5.6 Analyse des laufenden Arbeitsprozesses

Die Anwendung der im Rahmen dieser Arbeit beschriebenen Mensch-Roboter Kooperation zielt vor allem auf kleine und mittelständische Firmen, die ihre Prozesse teilautomatisieren wollen. Bei solchen Anwendungen wechseln jedoch die auszuführenden Prozesse häufig, so dass Änderungen bei systemrelevanten Prozessabhängigkeiten neu eingepflegt werden müssen. So ist bspw. die Erkennung der Bewegungsabsicht des Menschen von der zu erwartenden nächsten Position, d.h. der Prozessfolge, abhängig. Durch Änderungen im Ablauf können solche Abhängigkeiten zu Fehlern führen. Darüber hinaus ist es für Prozessoptimierungen interessant zu erkennen, welche Schritte tatsächlich durch den Werker ausgeführt werden. Eine übersichtliche Darstellung ist dabei einer reinen textuellen Aneinanderreihung von internen Variablenbelegungen vorzuziehen. Im Folgenden wird die Herangehensweise zur Prozessanalyse und Prozessfolgenermittlung beschrieben und deren Untersuchung detailliert. Ein erster Ansatz hierzu wurde in [Puls et al., 2013b] vorgestellt. Im Rahmen dieser Arbeit wird dieser erweitert.

### 5.6.1 Stand der Forschung

Zur Identifikation von Prozessfolgen in komplexen Prozessen müssen Muster variabler Länge in einer Zeitreihe diskreter Zustandsänderungen erkannt werden. Viele fundamentale Methoden wurden im Bereich der künstlichen Intelligenz zur Folgenerkennung entwickelt. In diesem Zusammenhang unterscheiden Sun und Giles zwischen Musterwiedererkennung sowie Musterdetektion und führen einige mögliche Lösungen an [Sun and Giles, 2001].

Eine dieser Lösungen kann die Verwendung von *induktive logische Programmierung* (ILP) sein. Santosh et al. nutzen ILP, um Symbole zu lernen und wiederzuerkennen. Auf positiven und negativen Beispielen basierend werden Regeln gelernt, die eine anschließende Klassifikation von Bildern ermöglicht [Santosh et al., 2009].

Durch Padmanabhan und Tuzhilin wird ein Ansatz vorgestellt, der temporale Logik verwendet, um häufige Muster in temporalen Datenbanken zu detektieren [Padmanabhan and Tuzhilin, 1996]. Dazu werden temporale Operatoren und temporale logische Programme eingesetzt. Diese Programme waren jedoch spezifisch für die Muster. Darüber hinaus stellen die Autoren fest, dass das System zu langsam für praktische Anwendungen ist.

Hidden Markov Modelle (HMMs) werden häufig eingesetzt, um temporale Muster zu lernen und zu erkennen [Arica and Yarman-Vural, 2001; Niu and Abdel-Mottaleb, 2005; Krajcovic et al., 2011]. Auf Grund der überwachten Lernstrategie von HMMs kann dieser Ansatz nicht zur Detektion unbekannter Muster eingesetzt werden. Chudova und Smyth präsentieren einen Ansatz, der HMMs zu Musterdetektion nutzt. Jedoch müssen Länge der Muster und Struktur der HMMs a-priori bekannt sein. Somit wird die Aufgabe der Entdeckung von unbekanntem Mustern zu einem Unterscheiden von ähnlichen Mustern.

Der Forschungsbereich des *Process Mining* (PM) zielt darauf ab, Wissen aus Ereignislogs zu extrahieren. Diese Ereignislogs bestehen aus Prozessereignissen mit deren temporaler Ordnung. PM wird hauptsächlich zur Identifikation und Optimierung von geschäftlichen Prozessen eingesetzt.

Verschiedene Methoden wurden entwickelt, die Prozesse sowie deren interne Struktur zu identifizieren und als sogenannte Workflow-Netze zu modellieren [van der Aalst, 2011a; van der Aalst, 2011b].

Der Alpha-Algorithmus ist einer der ersten entwickelten Methoden zur Prozessmodellierung [van der Aalst et al., 2004]. Dieser modelliert temporale Beziehungen von Ereignissen auf Grund ihres Auftretens im Ereignislog. Waimer zeigt auf, dass der Alpha-Algorithmus anfällig für unvollständige und rauschbehaftete Daten ist und somit fehlerhafte Prozessmodelle erzeugt [Waimer, 2006]. Der Tsinghua-Alpha-Algorithmus ist eine Weiterentwicklung und betrachtet auch die Dauer von Aktivitäten und erlaubt die Modellierung von Prozessparallelitäten [Wen et al., 2009]. Jedoch wird auch für diesen Ansatz ein vollständiges und rauschfreies Ereignislog benötigt.

Chang und Lui präsentieren ein System zur Entdeckung von Mustern in Webseiten, um Informationen aus wiederholten Strukturen zu extrahieren [Chang and Lui, 2001]. Dazu wird der HTML-Inhalt binär und mit fester Länge kodiert. Die Daten werden in Patricia-Bäumen (siehe [Morrison, 1968]) verwaltet und mit weiteren Informationen zu Auftrittshäufigkeiten und Referenzpositionen angereichert. Während eines Baumdurchlaufes können unpassende Knoten verworfen werden und wiederholte HTML-Strukturen extrahiert werden. Diese Strukturen können dann in Form regulärer Ausdrücke zur Informationsabfrage genutzt werden.

Der Einsatz der Autokorrelation zum Finden von Periodizitäten von Elementen einer Zeitserie wird durch Berberidis et al. vorgestellt [Berberidis et al., 1968]. Für jedes Element der Serie wird die beobachtete Serie separat in einer binominalen Sequenz kodiert. Durch die Verwendung von circularer Autokorrelation wird die Periode und Frequenz jedes Elements bestimmt. Elemente, die gleiche Periode und Frequenz besitzen werden gruppiert und als Muster betrachtet. Der Vorteil liegt in der Unabhängigkeit von Annahmen zu Musterlängen. Den Autoren zufolge sind die Korrelationswerte jedoch abhängig von der Serienlänge, was sich negativ auswirkt, falls die Serienlänge kein Vielfaches der Periode ist. In einem solchen Fall entstehen Ergebnisvektoren, die Rauschen und unerwartete Werte enthalten.

Der sogenannte *heuristische Miner*<sup>5</sup> (HM) modelliert Prozesse anhand der Häufigkeit und Anordnung von Ereignissen [Weijters et al., 2006]. Im Gegensatz zum Alpha und Tsinghua-Alpha Algorithmus kann der HM mit rauschbehafteten Datensätzen umgehen, was eine Grundvoraussetzung in einem sensorbasierten kognitiven System ist. Daher wird dessen Adaption an die online Prozessanalyse im Rahmen dieser Arbeit untersucht.

## 5.6.2 Prozessanalyse

Die wesentlichen Methoden zum Process Mining, wie sie oben aufgeführt sind, analysieren Geschäftsprozess offline. Datenlogs werden über einen gewissen Zeitraum aufgezeichnet und anschließend untersucht. Entsprechend ist für den Einsatz in veränderlichen MRK-Szenarien eine Anpassung zur online Analyse notwendig. Im Folgenden wird kurz der zugrunde gelegte Heuristic Miner eingeführt sowie anschließend die entsprechenden Anpassungen und Anwendungen erläutert.

### 5.6.2.1 Heuristic Miner

Mittels Heuristic Miner Algorithmus werden temporale Beziehungen und Abhängigkeiten auf Grund der Auftrittshäufigkeiten modelliert. Speziell wird analysiert, wie oft welche Ereignisse nach welchen anderen auftreten. Somit kann dieses Vorgehen auch als n-gram-Analyse aufgefasst werden.

Weijters et al. zufolge können verschiedene temporale Beziehungen unterschieden werden:

- $a > b$  - Ereignis  $b$  ist direkter Nachfolger von  $a$ .
- $a \rightarrow b$  - Ereignis  $a$  tritt generell vor  $b$  auf.
- $a \gg b$  - Es gibt einen Zyklus der Form  $a - b - a$ .
- $a \ggg b$  - Markiert Abhängigkeiten über große Zeiträume.

---

<sup>5</sup>engl.: Heuristic Miner

Durch die Verwendung dieser Beziehungen können Abhängigkeitsgraphen extrahiert werden. Für jede Relation und Kombination von Ereignispaaren wird anhand des Ereignislogs  $L$  ein Koeffizient bestimmt, wobei  $|\cdot|$  die Anzahl der Instanzen bezüglich der jeweiligen spezifizierten Relation angibt:

$$a \Rightarrow b = \frac{|a >_L b| - |b >_L a|}{|a >_L b| + |b >_L a| + 1} \quad (5.1)$$

$$a \Rightarrow_2 b = \frac{|a >>_L b| + |b >>_L a|}{|a >>_L b| + |b >>_L a| + 1} \quad (5.2)$$

Die Werte der Koeffizienten liegen, analog zum Korrelationskoeffizient, im Intervall  $[-1, 1]$ . Dabei gibt der Wert 1 vollständige direkte Abhängigkeit und  $-1$  vollständige indirekte Abhängigkeit an. Mittels eines Koeffizientenschwellwertes kann der Abhängigkeitsgraph bestimmt werden. Der Vorteil dieses Vorgehen liegt darin, dass geringes Prozessrauschen nicht zu wesentlichen Änderungen der Koeffizienten führt.

### 5.6.2.2 Online Analyse

Die Ergebnisse der Situationsanalyse werden als Menge von Ereignisstrippel  $E$  verarbeitet:

$$E = (t, d, c), \quad (5.3)$$

wobei  $t$  einen Zeitstempel angibt. Der Wert  $d$  stellt eine Identifizierung des Ereignisses dar, bspw. *Werkzeug nehmen*. Der Wert  $c$  gibt an, zu welcher Klasse das Ereignis gehört. Zu den Ereignisklassen gehören *Aktionen*, die vom Menschen durchgeführt werden, *Situationen*, die erkannt werden, und *Orte*, an denen sich der Mensch befindet. Anhand des Zeitstempels  $t$  kann die temporale Reihenfolge der Ereignisse rekonstruiert werden.

Die isolierte und kombinierte Betrachtung von Ereignisklassen führt entweder zu eindimensionalen oder mehrdimensionalen Ereignislogs. Für die online Analyse werden drei verschiedene Mengen von Ereignisklassen betrachtet:

- Aktionen,
- Aktionen und Situationen,

- Aktionen in Kombination mit ihren Auftrittsorten.

Die ersten beiden Mengen können als eindimensional angesehen werden, da die Ereignisse nacheinander aufgezeichnet werden. Die dritte Menge ist zweidimensional, da eine direkte Beziehung der Daten vorliegt.

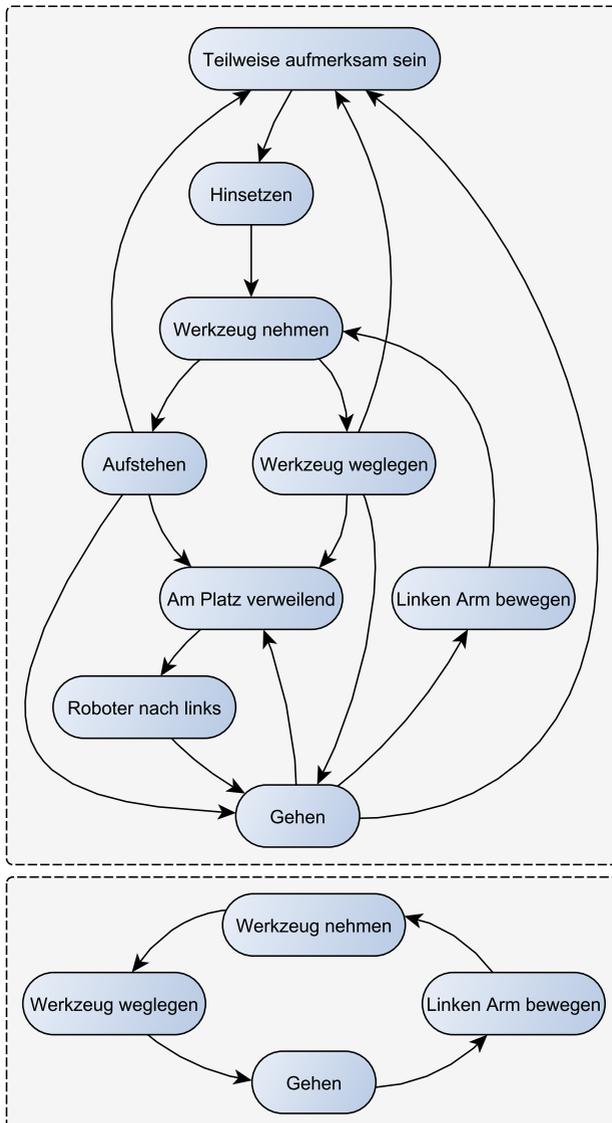
Ein weiterer Aspekt der Untersuchung betrifft die analysierte Datenmenge. Es ist nicht wünschenswert, dass Daten über Wochen oder Tage gesammelt werden müssen, um aussagekräftige Abhängigkeitsgraphen zu erstellen. Die tatsächlich betrachtete Datenmenge ist abhängig von der Komplexität des Prozesses und kann das Prozessmodell nachhaltig beeinflussen, wie in den Ergebnissen gezeigt wird.

Der untersuchte Ansatz extrahiert mittels Rückschau einen aktuellen Datensatz aus der aufgezeichneten Historie von Ereignissen. Durch die Weite der Rückschau lässt sich die Anzahl der untersuchten Ereignisse eingrenzen. Sobald der Datensatz ausreichend viele Ereignisse enthält, wird die Verarbeitung angestoßen. Anschließend wird eine definierte Menge von Ereignissen aus dem Datensatz entfernt, um neue Ereignisse eintragen zu können. Eine direkte Analyse mit jedem neuen Ereignis wäre denkbar, würde jedoch kaum Änderungen zum erzeugten Prozessmodell der vorherigen Analyse bringen.

In Abbildung 5.9 oben ist ein Beispiel für ein generiertes Prozessmodell dargestellt. Um die Komplexität des Graphen zu reduzieren wird eine Zyklensuche auf dem Graphen angewendet (siehe Abb. 5.9 unten). Die extrahierten Zyklen vereinfachen die Analyse des Prozesses. Jedoch können auch Zyklen extrahiert werden, die im Prozess zum einen nicht vorkommen und zum anderen nicht möglich sind. So kann in dem dargestellte Modell die Person sich wiederholt hinsetzen, ein Werkzeug nehmen und wieder aufstehen ohne je ein Werkzeug abzulegen. Daher ist die Analyse bezüglich Prozessfolgen notwendig, auf die im nächsten Abschnitt eingegangen wird.

### 5.6.3 Prozessfolgenermittlung

Wie im vorigen Abschnitt gezeigt, werden Prozessfolgen beim Heuristic Miner abstrahiert, so dass Informationen über Ereignisketten zusammengefasst in einem Modell dargestellt werden. Eine solche anschauliche Mo-



**Abbildung 5.9:** Beispiel eines Prozessmodells bezüglich Aktionen und Aktivitäten (oben). Beispiel eines extrahierten Zyklus (unten).

dellierung kann für Menschen hilfreich sein, die Prozesse zu analysieren und zu optimieren. Für eine automatisierte Weiterverarbeitung der Daten ist diese Form jedoch unzureichend. Im Folgenden wird die untersuchte Prozessfolgenmittlung eingeführt.

Angelehnt an die Autokorrelationsanalyse, wie sie bei der Signalverarbeitung vorkommt, werden Sequenzen  $s$  von Ereignissen  $e$  im Zusammenhang des gesamten Ereignisdatensatz  $L$  betrachtet. Die Länge einer Sequenz ist variabel, da die Anzahl der Elemente der Prozessfolge unbekannt ist und nicht vorausgesetzt wird. Somit gilt:

$$s_i = \{e_t, e_{t-1}, \dots, e_{t-k^i}\}_L, \quad (5.4)$$

wobei  $t$  den aktuell betrachteten Zeitpunkt definiert. Für  $k$  gilt  $k > 1$  mit  $k \in \mathbb{N}$ , wobei  $k^i$  den Wert von  $k$  für die  $i$ te Sequenz angibt.

Bei der Analyse werden die Sequenzen aufeinander aufbauend aus dem Datensatz  $L$  erzeugt. Sequenz  $s_{i+1}$  ergibt sich entsprechend aus Sequenz  $s_i$  durch Hinzunahme eines weiteren Elements aus  $L$ . Durch Beobachtung neuer Ereignisse schreitet  $t$  voran und es können, abhängig von der Prozessfolgenlänge, verschobene Sequenzen beobachtet werden, die jedoch eine bereits erstellte Sequenz beschreibt. Somit bilden Sequenzen gleicher Länge Äquivalenzklassen, wenn diese die gleiche Prozessfolge darstellen.

Analog zur Autokorrelationsanalyse wird jede Sequenz  $s_i$  mit jeder gleichlangen Sequenz  $\hat{s}_i$  aus  $L$  verglichen und ein Koeffizient bestimmt, der ein Maß für die Ähnlichkeit darstellt. Da die Elemente von  $L$  als Zeichenketten vorliegen, bestimmt sich der Koeffizient aus der Levenshtein-Distanz. Der Koeffizient  $C_{s_i}$  einer Sequenz  $s_i$  bestimmt sich durch:

$$C_{s_i} = \frac{|D_{s_i}^{>n}|}{\frac{|L|}{k^i+1}}, \quad (5.5)$$

dabei gibt  $|L|$  die Anzahl der Ereignisse des Datensatzes  $L$  an und  $|D_{s_i}^{>n}|$  gibt die Häufigkeit bezüglich der Äquivalenzklasse der Sequenz  $s_i$  an, mit der die Levenshtein-Distanz  $D$  über dem Schwellwert  $n$  lag. Ähnlich zum Heuristic Miner kann auf diese Weise Prozessrauschen behandelt werden, da nicht nur vollständig identische Sequenzen zum Ergebnis bei-

tragen. Abhängig vom zu erwartenden Prozessrauschverhalten kann der Schwellwert  $n$  angepasst werden.

Als resultierende Prozessfolge wird diejenige bestimmt, für die  $C_{s_i}$  maximal ist und über einem Schwellwert  $m$  liegt.

## 5.6.4 Evaluation und Diskussion

Die Evaluation wurde in zwei Abschnitte aufgeteilt. Zur Analyse der Parametereinflüsse wurde eine Methode zur automatischen Ereignisvorgabe implementiert. Zur Beurteilung des online Verhaltens der Prozessanalyse wurden Sensordaten verarbeitet. Somit kann Sensorrauschen die Erkennungsergebnisse und damit die Prozessanalyse beeinflussen.

Der generelle Aufbau der Evaluationsszenarien basiert auf der Arbeitsraumaufteilung aus Abbildung 5.1 und damit auf drei Orten, an denen der Werker tätig sein kann.

### 5.6.4.1 Qualitätskriterium für Heuristic Miner

Die Ergebnisse des Heuristic Miner sind Modellgraphen. Damit diese beurteilt werden können, wurde ein Qualitätskriterium entworfen. Basierend auf dem Vergleich der Ergebnissen und erwarteten Resultaten wird mittels Kriterium versucht die folgenden Fragestellungen zu beantworten:

- Wurden die richtigen Übergänge identifiziert?
- Wurden zu viele Knoten identifiziert?
- Wurden alle richtigen Knoten identifiziert?

Entsprechend ergibt sich für das gesamte Qualitätskriterium  $Q$  die Aufteilung in drei weitere Kriterien:

$$C_1 = \frac{\text{Richtige Anzahl Übergänge}}{\text{Anzahl gefundener Übergänge}} \Rightarrow Q_1 = |C_1 - 1| \quad (5.6)$$

$$C_2 = \frac{\text{Anzahl richtig identifizierter Knoten}}{\text{Anzahl identifizierter Knoten}} \Rightarrow Q_2 = |C_2 - 1| \quad (5.7)$$

$$C_3 = \frac{\text{Anzahl richtig identifizierter Knoten}}{\text{Richtige Anzahl Knoten}} \Rightarrow Q_3 = |C_3 - 1| \quad (5.8)$$

$$Q = \max(Q_1, Q_2, Q_3) \quad (5.9)$$

Durch die Wahl des Maximums wird das Kriterium ausgewählt, welches das Ergebnis am meisten negativ beeinflusst. Folglich ist ein Wert 0 für  $Q$  optimal.

#### 5.6.4.2 Analyse anhand synthetischer Daten

Zur effizienten Analyse der Einflussnahme der verschiedenen Parameter des Heuristic Miner wurde eine Menge an zyklischen Ereignisdatensätze definiert. Enthalten waren verschiedene Folgen von Aktionen während verschiedenen Situationen an verschiedenen Orten. Abhängig vom untersuchten Parameter wurden die Vorgaben isoliert oder in beliebiger Kombination getestet. Die folgenden Parameter wurden mittels synthetischer Daten untersucht:

- Anzahl verschiedener Zyklen im Datensatz,
- Länge der Zyklen,
- Einfluss von Prozessrauschen im Datensatz,
- Größe des Datensatzes.

Die Variation der Zyklenzahl resultierte in komplexeren Prozessmodellen. Wie erwartet blieb der Wert des Qualitätskriteriums optimal. Die gleiche Beobachtung gilt auch für Variationen der Zyklenlängen.

Robustheit gegenüber Rauschen ist ein wesentlicher Aspekt, wenn es um Sensordatenverarbeitung geht. Zur Evaluation wurde Rauschen zur Vorgabe der Ereignisse eingeführt. Zum einen wurden zufällige Ereignisse nach vollständigen Zyklen eingeführt, im Folgenden angegeben mit  $R_1$ . Zum anderen wurden zufällige Ereignisse innerhalb der Zyklen eingeführt, angeben als  $R_2$ . Diese zwei Rauschverhalten wurden auch in Kombination betrachtet. In Tabelle 5.6 sind die Ergebnisse bezüglich Rauschverhalten aufgeführt. Für Zellen mit der Angabe  $x$  sind die Qualitätswerte unzureichend. Der Rauschwert  $R_2$  hat generell geringeren Einfluss.

**Tabelle 5.6:** Variationen von additivem Rauschen. Das Qualitätskriterium wurde für verschiedene Rauschwerte evaluiert.

$R_2 \setminus R_1$	0%	3%	5%	8%	12%	20%
0%	0	0,05	0,12	0,19	x	x
3%	0	0,12	0,19	0,12	x	x
5%	0,05	0,05	0,12	x	x	x
8%	0,05	0,12	x	0,26	x	x
12%	x	x	x	x	0,26	x
20%	x	x	x	x	x	0,46

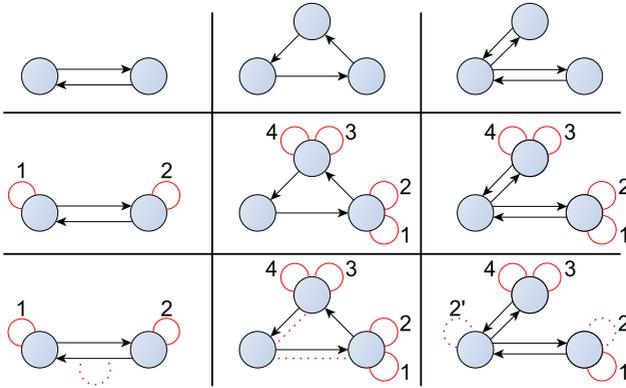
Ebenfalls von Interesse ist die Menge an Daten, die für eine hinreichende Analyse notwendig ist. In Tabelle 5.7 sind die Ergebnisse zur Analyse der Sequenzlänge aufgeführt. Entsprechend werden akzeptable Resultate mit 800 analysierten Ereignissen erreicht. Auf Grund der größeren Einflussnahme von  $R_1$  wurde  $R_2$  nicht im Besonderen untersucht.

**Tabelle 5.7:** Variationen von Sequenzlängen. Das Qualitätskriterium wurde für verschiedene Längen und Rauschwerte evaluiert.

$R_1 \setminus R_2 \setminus$ Länge	100	200	400	800
0% 0%	8,5	0,72	0,35	0
3% 0%	8,5	0,72	0,58	0,19
5% 0%	8,5	0,9	0,9	0,19

### 5.6.4.3 Analyse anhand Sensordaten

Zur Analyse anhand Sensordaten wurden verschiedene Handlungsabläufe definiert. Diese sind in Abbildung 5.10 schematisch dargestellt. Um Prozessfehler und unerwartete Ereignisse zu simulieren, wurden absichtlich zusätzliche Handlungen an zufälligen Zeitpunkten ausgeführt.

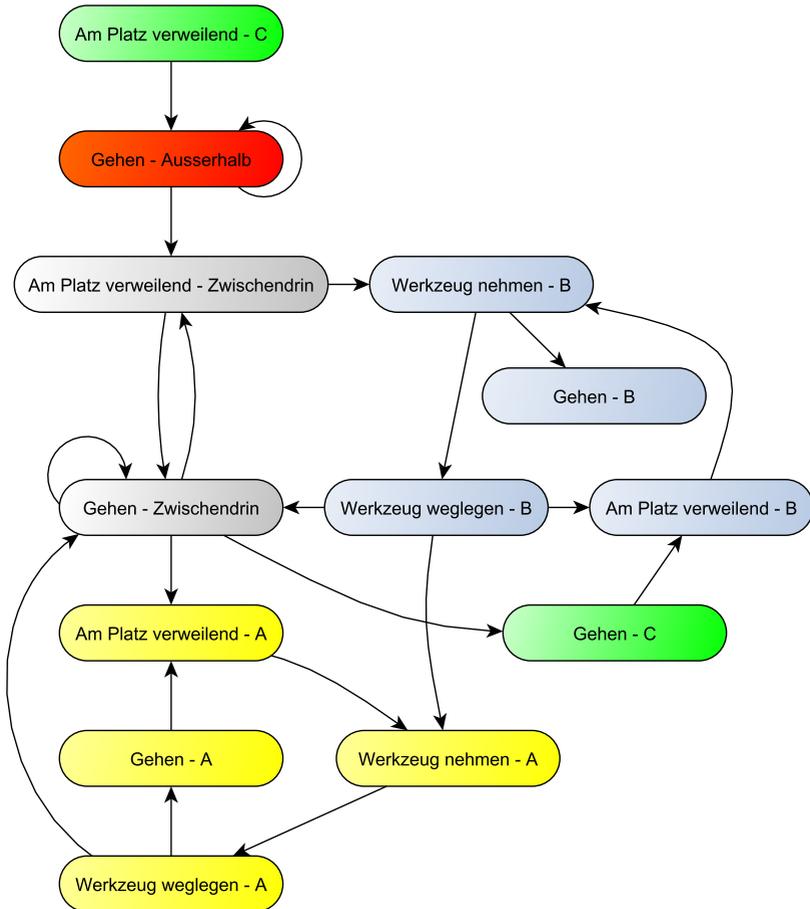


**Abbildung 5.10:** Schematische Darstellung verschiedener Handlungsabläufe. Jede Spalte repräsentiert ein Szenario, welches mit jeder Zeile erweitert wird. Punkte stellen Orte dar. Schwarze Pfeile repräsentieren Ortswechsel, rote Kreise geben Handlungen an und gepunktete rote Linien deliberativ ausgeführtes Rauschen.

Das Ergebnis der Prozessanalyse für die zweite Sequenz des zweiten Szenarios (siehe Abb. 5.10, zweite Spalte und Reihe) ist in Abbildung 5.11 dargestellt. Die generelle Struktur der Sequenz wird ersichtlich und wurde gut abgebildet.

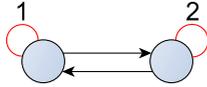
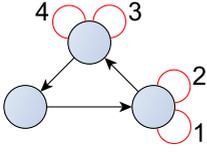
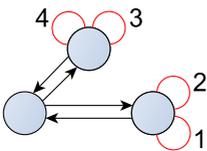
Wie bei den synthetischen Daten sind die betrachteten Sequenzlängen von Interesse. Jedoch werden bessere Ergebnisse bereits mit 400 Datensatzeintragen erreicht, wie in Tabelle 5.8 aufgeführt.

Ebenfalls Untersuchungsgegenstand war der Koeffizientenschwellwert, der festgelegt wird, um Abhängigkeiten zwischen Ereignissen zu bestimmen. In Tabelle 5.9 sind die Ergebnisse zu Untersuchungen für verschiedene Szenarien mit unterschiedlichen Schwellwerten aufgeführt. Die Sequenzlänge beträgt 400 Ereignisse.  $Q^1$  repräsentiert das Kriterium für die erste Sequenz,  $Q^2$  entsprechend die zweite Sequenz jeweils des zweiten Szenarios. Ein Schwellwert von 0,9 erreicht die besten Ergebnisse in der Untersuchung. Das kann sich jedoch mit zunehmender Komplexität ändern. Darüber



**Abbildung 5.11:** Identifiziertes Prozessmodell der zweiten Sequenz des zweiten Szenarios. Verschiedene Farben repräsentieren verschiedene Orte: Gelb – A, Blau – B, Grün – C, Grau – Zwischendrin und Rot – Ausserhalb.

**Tabelle 5.8:** Variationen von Sequenzlängen. Das Qualitätskriterium wurde für verschiedene Längen und Szenarien evaluiert.

Szenario \ Länge	100	200	400	800	1200
	4	0,5	0,29	0,29	0,4
	x	4	0,25	0,55	0,63
	x	1,5	0,2	0,42	0,47

hinaus erlaubt ein niedriger Schwellwert schnellere Anpassung an sich ändernde Prozesse.

**Tabelle 5.9:** Variationen von Koeffizientenschwellwert.

Schwellwert	0,85	0,875	0,9	0,925
$Q^1$	0,5	0,29	0,285	1
$Q^2$	0,55	0,36	0,25	0,43

Die Betrachtung der Laufzeitanalyse hat ergeben, dass bei der Analyse von 100 bis 500 Ereignissen etwa 300 ms benötigt werden. Bei 500 bis 1000 Ereignissen steigt der Wert auf etwa 450 ms. Ungefähr 650 ms werden bei 1000 bis 1500 Ereignissen benötigt. Somit ist die Verarbeitung der Prozessanalyse ausreichend schnell, um während des Prozesses ausgeführt

zu werden. Darüber hinaus sind Änderungen der Prozesse in sehr viel größeren Zeitabschnitten zu erwarten, so dass die Analyse nicht kontinuierlich durchgeführt werden muss.

#### 5.6.4.4 Prozessfolgernermittlung

Die Methode zur Prozessfolgernermittlung wurde ebenfalls anhand der oben aufgeführten Szenarien evaluiert. In Tabelle 5.10 ist die Entwicklung von der Basissequenz des zweiten Szenarios zur Basissequenz des dritten Szenarios aufgeführt. Dabei wurde die Betrachtung auf die Orte beschränkt, wobei mit A, B und C die Arbeitsplätze im Arbeitsraum bezeichnet sind. Der Buchstabe O steht für die Beobachtung, dass der Mensch den Arbeitsraum verlassen hat.

In der Tabelle 5.10 sind die ermittelten Koeffizienten für Sequenzen aufgeführt, wobei nur Sequenzen mit einem Koeffizienten größer 0,4 aufgeführt sind. Es ist dabei zu erkennen, dass seltene Ereignisse bei der Betrachtung als Rauschen behandelt werden und nicht in den möglichen Prozessfolgen auftreten.

**Tabelle 5.10:** Betrachtung der Sequenzentwicklung.

CBACOCBACBACBOACBCACBCACB	ACB:0,6		
BACOCBACBACBOACBCACBCACBC	ACB:0,6	ACBC:0,48	
ACOCBACBACBOACBCACBCACBCA	ACB:0,6	ACBC:0,48	
COCBACBACBOACBCACBCACBCAC	ACB:0,6	ACBC:0,48	
OCBACBACBOACBCACBCACBCACB	ACB:0,72	ACBC:0,48	
CBACBACBOACBCACBCACBCACBC	ACB:0,72	ACBC:0,64	
BACBACBOACBCACBCACBCACBCA	ACB:0,72	ACBC:0,64	BCA:0,48
ACBACBOACBCACBCACBCACBCAC	ACB:0,72	ACBC:0,64	BCA:0,48
CBACBOACBCACBCACBCACBCACB	ACB:0,72	ACBC:0,64	BCA:0,48
BACBOACBCACBCACBCACBCACBC	ACB:0,72	ACBC:0,8	BCA:0,48



# 6 Sicherheit und Risikobeurteilung

Die Sicherheit der Menschen, die Umgang mit technischen Systemen haben, hat systemrelevante Bedeutung. Wie bereits im Kapitel 2.2 gezeigt, sind Sicherheitsbetrachtungen und Risikobeurteilung Gegenstand aktiver Forschungsarbeiten. Im Rahmen der vorliegenden Arbeit stellt die Risikobetrachtung einen wesentlichen Punkt in der Kooperation zwischen Mensch und Roboter dar. Im Folgenden werden die entwickelten und untersuchten Methoden sowie Ansätze zur Erkennung und Behandlung von risikobehafteten Situationen erläutert.

## 6.1 Situationsabhängige Risikobeurteilung

Die aufgeführten Arbeiten im Kapitel 2.2 verdeutlichen, dass Sicherheitsbetrachtungen häufig auf Aspekte der Roboterdynamik, Prozesskräfte oder Regelungsansätze zurückgeführt werden. Am Beispiel der Untersuchungen von Haddadin et al. wird deutlich, dass manche Robotersysteme in der Lage sind, die Unversehrtheit des Menschen zu gewährleisten [Haddadin et al., 2012]. Jedoch bedarf es der manuellen Auswahl der gültigen Reglerparametersätze, um es dem Roboter zu ermöglichen, gewollte Prozesskräfte, bspw. am Werkstück, aufzubringen.

Eine situationsabhängige Risikobeurteilung kann hierzu genutzt werden, um ein sicheres Verfahren trotz gewollter physischer Roboter-Objekt- und Roboter-Mensch-Interaktion zu ermöglichen. Wie im Kapitel 3 aufgeführt, wurde für die Risikobeurteilung auf der Vorarbeit von Graf (siehe [Graf, 2010]) aufgebaut. Darin wurde ein erster Ansatz für eine Regelbasis basierend auf unscharfer Logik entwickelt. Ein erweiterter Ansatz wurde in

[Puls and Wörn, 2013] vorgestellt und wird im Rahmen der vorliegenden Arbeit aufgegriffen.

### 6.1.1 Grundlagen

Die Rekonstruktion der menschlichen Kinematik stellt die Grundlage für die Risikobestimmung in [Graf, 2010] dar. Es werden Parameter speziell bezüglich Blickrichtung, Position und Bewegungsgeschwindigkeit durch die modellierte Regelbasis ausgewertet. Ein Ausschnitt dieser Regelbasis ist der Übersicht halber in Tabelle 6.1 aufgeführt. Im Gegensatz zur klassischen unscharfen Logik wird eine zweisträngige Logik verwendet, die es erlaubt positive und negative Regeln zu formulieren. Entsprechend können Warnungen und Verbote modelliert werden. Darüber hinaus soll eine reduzierte Regelbasis erreichbar sein.

**Tabelle 6.1:** Auszug aus der grundlegenden Regelbasis zur Risikobestimmung. Abkürzungen: R - Risiko, D - Distanz, OKO - Oberkörperorientierung, KO - Kopforientierung, V - Geschwindigkeit

<b>Positive Regeln</b>	
<i>Antezedent</i>	<i>Konsequent</i>
$D \leftarrow \textit{gro\ss}$	$R \leftarrow \textit{sicher}$
$D \leftarrow \textit{klein} \wedge V \leftarrow \textit{klein}$	$R \leftarrow \textit{moderat}$
$\text{OKO} \leftarrow \textit{gedreht} \wedge \text{KO} \leftarrow \textit{gedreht}$ $\wedge V \leftarrow (\textit{negativ} \vee \textit{null})$	$R \leftarrow \textit{gef\ahrl\ich}$
<b>Negative Regeln</b>	
<i>Antezedent</i>	<i>Konsequent</i>
$\text{OKO} \leftarrow \textit{fokussiert} \wedge \text{KO} \leftarrow \textit{fokussiert}$	$R \leftarrow \{ \textit{gef\ahrl\ich}, \textit{prek\ar}, \textit{nicht sicher} \}$ <b>unerw\unscht</b>
$\text{OKO} \leftarrow \textit{fokussiert} \wedge \text{KO} \leftarrow \textit{etwas gedreht}$	$R \leftarrow \{ \textit{gef\ahrl\ich}, \textit{prek\ar} \}$ <b>unerw\unscht</b>

## 6.1.2 Modellierung der Situationsabhängigkeit

Die Interpretation der Situation ermöglicht die Schlussfolgerung, ob ein Kontakt bzw. geringer Abstand zwischen Mensch und Roboter beabsichtigt ist. Jedoch führt die Anwendung der bestehenden Regelbasis dazu, wie in der Arbeit von Graf gezeigt, dass der Roboter dem Menschen generell ausweicht. Auf diese Weise ist keine kontaktbasierte Kooperation möglich. Trotzdem ist es das Ziel, dass bei einer nicht-kooperativen Situation, die Sicherheit des Menschen gewährleistet werden kann. Da die Regelbasis von Graf getestet und evaluiert wurde, soll diese als Grundlage für die Systemerweiterung dienen.

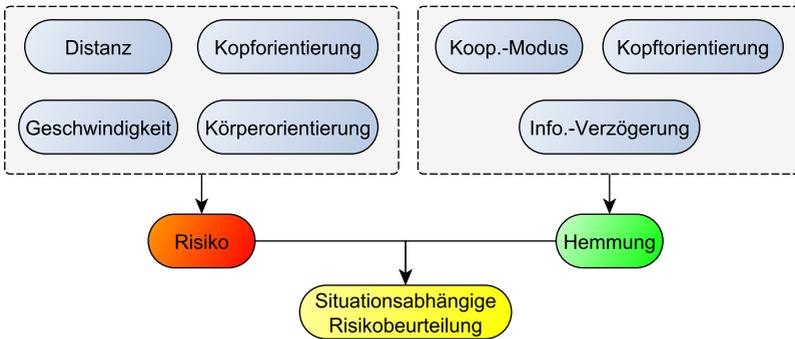
Die bloße Anpassung bestehender Regeln kann zu unterschiedlichem Systemverhalten führen, weshalb die bereits durch Graf evaluierten Basisfälle erneut überprüft werden müssten. Ein anderer Ansatz ist das Ausnutzen der zweisträngigen Regelbasis, in dem neue negative Regeln eingeführt werden, die das Risiko reduzieren, falls Kooperation vorliegt. Jedoch ist der Inferenzoperator, der positive und negative Regeln kombiniert, nicht selektiv steuerbar. Dadurch kann es auch bei Kooperation zu verhindernden Risikoannahmen kommen.

Als Konsequenz wurde ein System entworfen, das angelehnt an Neuronale-Unscharfe-Netze, einen Knoten zur Risikobestimmung und einen Knoten zur Hemmung aufweist (siehe Abb. 6.1). Die Hemmung bestimmt sich ebenfalls durch die Auswertung einer zweisträngigen unscharfen Regelbasis. Der Risikowert und die Hemmung werden nach der jeweiligen Defuzzifizierung miteinander verrechnet.

Die Hemmung  $H$  ist indirekt proportional zu der Risikoreduzierung. Entsprechend wird die resultierende Risikobeurteilung  $R$  aus dem vorläufigen Risikowert  $r$  folgendermaßen bestimmt:

$$R = r * (1 - H) \tag{6.1}$$

Zur Abbildung der aktuellen Situation wird die unscharfe Variable *Kooperationsmodus* eingeführt. Der Wert dieser Variablen bestimmt sich aus einer Kombination von gefolgelter Situation und Zielsetzung. Zur Belegung der Variablen werden drei Zustände definiert:



**Abbildung 6.1:** System zur situationsabhängigen Hemmung der Risikobeurteilung.

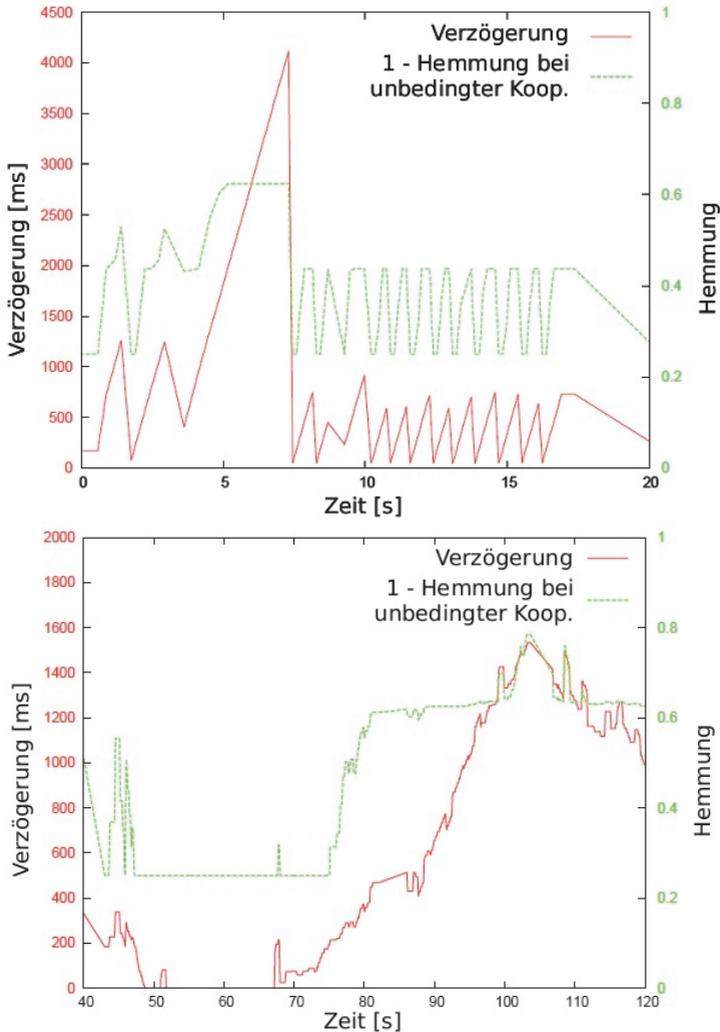
**Aus** Zeigt an, dass keine Kooperation nötig ist.

**Unbedingt** Der Roboter kann mit dem Menschen kooperieren, ohne dass dieser direkte Aufmerksamkeit auf den Roboter richtet. Eine Anwendung kann hierbei die Platzierung von Objekten am Arbeitsplatz nahe des Menschen sein, wenn bspw. Nachschubmaterial dem Menschen bereitgelegt wird.

**Bedingt** Der Mensch muss seine Aufmerksamkeit dem Roboter zuwenden, bspw. wenn ein Objekt übergeben wird.

Wie in Abschnitt 5.4.5 ersichtlich wurde, erreicht die Laufzeit des Folgerungsmoduls eine Frequenz von etwa 4 Hz im Durchschnitt. In einzelnen Fällen kann die Berechnung jedoch länger andauern. Folglich kann es passieren, dass zur Bestimmung der Risikohemmung veraltete Informationen vorliegen. In solchen Fällen ist es nötig, die Hemmung abzuschwächen und den originalen Risikowert stärker zu gewichten. Entsprechend wurde die Variable *Informationsverzögerung* definiert, die sich in folgende unscharfe Mengen gliedern lässt: *Aktuell*, *Verspätet* und *Alt*.

Die Verzögerung wird in Millisekunden angegeben und ergibt sich direkt aus der notwendigen Verarbeitungszeit des Folgerungsmoduls. Wie in Abbildung 6.2 oben zu sehen ist, kann die direkte Verwendung des tatsächlichen Verzögerungswertes zu wechselhaftem Verhalten führen. Entsprechend



**Abbildung 6.2:** Oben: Erratischer Verlauf der Risikobeurteilung, falls der tatsächliche Verzögerungswert direkt genutzt wird. Unten: Verhalten der Hemmung mit adaptiver Mittelung. Der Verzögerungswert wurde zur Verdeutlichung manuell manipuliert.

wird zur Glättung die Verzögerung anhand einer adaptiven Mittelung gesetzt. Der Verzögerungswert wird über Durchschnittsbildung ermittelt. Falls jedoch die aktuelle Verzögerung größer als das Mittel ist, wird der aktuelle Wert verwendet (siehe Abb. 6.2 unten). Auf diese Weise wird immer die pessimistische Betrachtung verwendet, da es direkt Einfluss auf die Risikobeurteilung haben kann.

Die unscharfe Ausgabevariable *Hemmung* wird durch drei Mengen zusammengesetzt: *Aufheben*, *Reduzieren* und *Ungehemmt*. Der minimale Wert der defuzzifizierten Variable ist auf 0,25 festgelegt, damit der Risikowert nicht vollständig aufgehoben werden kann und weiterhin Informationen der grundlegenden Risikobestimmung weitergereicht werden.

Anhand der eingeführten Variablen und unscharfen Mengen wurden zusätzliche unscharfe Regeln definiert. Diese sind in Tabelle 6.2 aufgeführt. Neben den eingeführten Variablen werden ebenfalls Kopforientierung und relative Geschwindigkeit zwischen Mensch und Roboter betrachtet. Somit deklarieren die positiven Regeln die Notwendigkeit einer Hemmung im Falle einer kooperativen Situation. Falls die Informationen zu alt sind, keine Kooperation vorliegt oder eine zu schnelle Abstandsreduzierung vorliegt, darf der Risikowert nicht reduziert werden.

### 6.1.3 Evaluation und Diskussion

Um Wiederholbarkeit zu erreichen, wurden Sensordaten aufgezeichnet und für Evaluationszwecke wiederholt eingelesen und wiedergegeben. Zur Analyse der Risikobeurteilung wurde hauptsächlich eine statische Szene ausgewählt und spezifische Variablenbelegungen manuell vorgegeben. Anhand dessen kann das Systemverhalten in verschiedenen Zuständen direkt beobachtet werden. Der Roboter führte eine Bewegung aus, bei der dieser sich durch den freien Arbeitsraum bewegte. In diesem Raum kann sich der Mensch befinden, so dass bei der Roboterbewegung unterschiedliche Risikosituationen auftreten können. In der statischen Szene steht der Mensch dem Roboter nahe, jedoch rückwärts gewandt.

Anhand Abbildung 6.2 unten lässt sich das Verhalten bezüglich Änderungen der Verzögerung erkennen. Die Werte für die Verzögerung wurden manuell festgelegt. Es ist ersichtlich, dass im Falle kleiner Verzögerungen

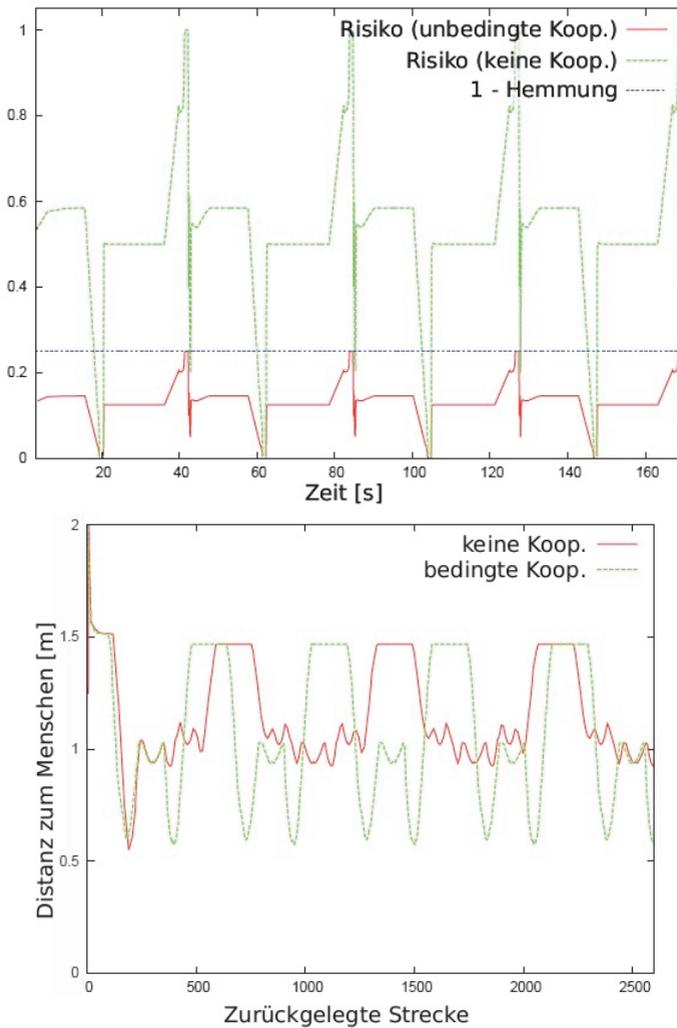
**Tabelle 6.2:** Zusätzliche Regelbasis für die Hemmung des Risikowertes. Abkürzungen: KM - Koop.-Modus, H - Hemmung, KO - Kopforientierung, IV - Info.-Verzögerung, V - Geschwindigkeit

<b>Positive Regeln</b>	
<i>Antezedent</i>	<i>Konsequent</i>
$KM \leftarrow \textit{unbedingt}$	$H \leftarrow \textit{aufgehoben}$
$KM \leftarrow \textit{bedingt} \wedge KO \leftarrow \textit{fokussiert}$	$H \leftarrow \textit{aufgehoben}$
$IV \leftarrow \textit{verspätet}$	$H \leftarrow \textit{ungehemmt}$
<b>Negative Regeln</b>	
<i>Antezedent</i>	<i>Konsequent</i>
$IV \leftarrow \textit{alt}$	$H \leftarrow \{ \textit{aufgehoben, reduziert} \}$ <b>unerwünscht</b>
$KM \leftarrow \textit{aus}$	$H \leftarrow \{ \textit{aufgehoben, reduziert} \}$ <b>unerwünscht</b>
$V \leftarrow \textit{groß}$	$H \leftarrow \{ \textit{aufgehoben, reduziert} \}$ <b>unerwünscht</b>

die Reduzierung maximal ist. Für den Fall, dass die Verzögerung zunimmt, wird, wie zu erwarten ist, die Hemmung geringer.

Die Festlegung des Kooperationsmodus wurde manuell eingestellt, um zu analysieren, wie das System in der gleichen Umgebung reagiert. In Abbildung 6.3 oben ist zu erkennen, wie das Risiko bei unbedingter Kooperation reduziert wird. Wie im Abschnitt 6.3 gezeigt wird, ist es in solchen Fällen jedoch notwendig, dass der Roboter mit einer reduzierten Geschwindigkeit verfahren wird. In Abbildung 6.3 unten ist im gleichen Zusammenhang zu erkennen, dass der Roboter mit reduziertem Risiko einen geringeren Sicherheitsabstand zum Menschen einhält. Eine räumlich enge Kooperation kann somit erreicht werden.

Der Wechsel des Kooperationsmodus entspricht einem diskreten Zustandswechsel des Systems. Entsprechend findet auch die Anwendung der Risikoreduzierung in diskreten Übergängen statt. Innerhalb eines Zustands sind Risiko- und Hemmungswerte kontinuierlich. Der Wechsel kann somit



**Abbildung 6.3:** Oben: Risikoausgabe des ursprünglichen und des erweiterten Systems. Unten: Betrachtung der Abstandseinhaltung zum Menschen.

jedoch zu plötzlichen Änderungen des Roboterverhaltens führen. Eine Möglichkeit des stetigen Zustandsübergang kann durch die Ausnutzung der unscharfen Definition des Kooperationsmodus darstellen. Der scharfe Wert, der den Modus definiert, kann entsprechend kontinuierlich an den Zielzustand angepasst werden. Somit werden die Zustandsänderung im System auch kontinuierlich. Dieser Aspekt wird im folgenden Abschnitt 6.2 weiterverfolgt.

## 6.2 Plausibilitätsprüfung

Die Plausibilitätsprüfung von Beobachtungen und Modellen ist ein Problem, das sich verschiedene Forschungs- und Ingenieursfelder teilen. Im vorliegenden Fall sollen Erkennungsergebnisse verschiedener Methoden im Zusammenhang untersucht werden, um Widersprüche bei Beobachtungen aufzudecken.

### 6.2.1 Stand der Forschung

Im Bereich der Spracherkennung müssen Beobachtungen bezüglich Mehrdeutigkeiten untersucht werden. Somit müssen nicht-plausible Hypothesen entfernt oder plausible Bedeutungen ausgewählt werden. Hinkelman und Allen beschäftigen sich mit diesem Aspekt und argumentieren, dass es Ähnlichkeiten zum Planerkennungproblem gibt [Hinkelman and Allen, 1989]. Sie schlagen einen einschränkungsbasierten Ansatz vor, um Ambiguitäten auszulösen. Die Einschränkungen bestehen aus zusammengesetzten, sprachabhängigen Regeln und Regeln zur Schlussfolgerung über Pläne. Die zusammengesetzten Regeln werden genutzt, um mögliche Interpretationen zu identifizieren. Die Schlussfolgerungsregeln werden angewendet, um Interpretationen zu erweitern oder einzuschränken.

Einen Überblick über das Thema Prozessfehlerdetektion und Diagnose bezüglich quantitativer, modellbasierter Methoden wird durch Venkatasubramanian et al. gegeben [Venkatasubramanian et al., 2003]. Dabei werden verschiedene Gründe für Prozessfehler aufgeführt und wünschenswerte Charakteristika eines Fehlerdiagnosesystems vorgeschlagen. Diese Charakteristika spannen einen Bogen von Detektionsgeschwindigkeit über

Robustheit zu Speicher- und Rechenanforderungen. Entsprechend sind diese generell zu verschiedenen Graden abhängig vom Zielsystem anwendbar.

Im Bereich dynamischer Fahrzeugregelungssysteme werden verschiedene Sensorsignale verarbeitet. Vermold und Saeger haben in diesem Zusammenhang ein System zur Plausibilitätsprüfung vorgestellt, das sich mit den wachsenden Mengen an verfügbaren Daten in modernen Fahrzeugsystemen beschäftigt [Vermold and Saeger, 2006]. Das vorgeschlagene System besteht aus Modulen zur Beobachtung, bspw. Einzelwertüberwachung oder redundanzbasiertes Überprüfen, und zur kombinierten Fehlerevaluation. Im Falle der Fehlerdetektion können abhängig vom Fehlertyp verschiedene Reglerparameter ausgewählt werden.

Von Kalal et al. wird ein System zur automatischen Detektion von Verfolgungsfehlern vorgeschlagen [Kalal et al., 2010]. Die Verfolgung von Merkmalspunkten in einer Bilderserie kann zu fehlerhaften Punkttrajektorien führen, da Verdeckungen oder fehlerhafte Merkmalsextraktion Einfluss haben. Durch den Vergleich der vorwärts und rückwärts Verfolgungsfehler können Inkonsistenzen in möglichen Trajektorien gefunden werden.

Konsistenz ist ein wesentlicher Aspekt, wenn Modelle definiert werden, wie bspw. UML, System- oder Wissensmodelle. Liu et al untersuchen eine regelbasierte Methode, um Inkonsistenzen in UML Modellen festzustellen [Liu et al., 2002]. Sie charakterisieren Inkonsistenzklassen, die im Bereich des Softwaredesign vorkommen. Die Klassen werden im Zusammenhang von Redundanz und Konformität mit Standards und Einschränkungen definiert. Basierend auf einem Produktionssystem und entsprechenden Regeln können Inkonsistenzen detektiert werden. Anhand dessen können Lösungen vorgeschlagen werden. Einen anderen Ansatz zum gleichen Problem der UML-Modellinkonsistenzerkennung wird durch Blanc et al. vorgestellt [Blanc et al., 2008]. Anstelle der Modellrepräsentation durch enthaltene Elemente werden Sequenzen von elementaren Konstruktionsoperationen verwendet. Mittels logischer Einschränkungen auf diesen Sequenzen können strukturelle und methodische Konsistenzregeln ausgedrückt werden. Dieser Ansatz wird von Blanc et al. erweitert, indem inkrementelle Erkennung von Modellinkonsistenzen eingeführt wird [Blanc et al., 2009]. Ebenfalls logisch motiviert ist der Ansatz von Van der Straeten durch Einsatz von Beschreibungslogiken, um Konsistenz bei verändernden UML-Modellen

zu bewahren bzw. Inkonsistenzen bei Änderungen festzustellen [Straeten et al., 2003].

Ein Ansatz, um Inkonsistenzen zu handhaben, wird anhand der Anwendung in semantischer Bildanalyse durch Dasiopoulou präsentiert [Dasiopoulou et al., 2010]. Es werden dabei unscharfe Beschreibungslogiken eingesetzt. Um das Abbrechen der Folgerungsmethode auf Grund von Modellinkonsistenzen zu vermeiden, wurde die Wissensbasis umgeschrieben und durch sogenannte *noConcepts* erweitert. Aus diese Weise können Instanzen dieses Konzeptes abgefangen und zurückverfolgt werden. Entsprechend können Lösungen vorgeschlagen und Inkonsistenzen behandelt werden.

Diese Beispiele zeigen, dass regelbasierte und logikbasierte Ansätze populäre Methoden sind, um Plausibilität und Inkonsistenz zu überprüfen. Obwohl die Anwendungsfelder weit gestreut sind, scheint die Anwendung in umfangreichen, multi-modalen Systemen selten. Im Folgenden wird gezeigt, dass diese Methoden großes Potential für die gegenseitige Evaluation verschiedener abstrakter Erkennungsergebnisse haben. Das gilt vor allem, da Erkennungsalgorithmen keine Fehlerrate von 0 erreichen.

## 6.2.2 Methode zur Plausibilitätsprüfung

In Anbetracht der Idee hinter sogenannten *Object Action Complexes* (OACs) [Krüger et al., 2009], werden Aktionen häufig in Verbindung mit Objekten ausgeführt. Einerseits können Objekte Handlungen ermöglichen, bspw. ermöglichen Werkzeuge produktive Handlungen. Basierend auf der Arbeit von Gibson in [Gibson, 1977] wird dies im Rahmen von OACs Affordanz genannt, was den Angebotscharakter von Objekten ausdrückt. Andererseits können Objekte die Durchführung von Handlungen unterbinden oder behindern, bspw. wird gehen verhindert, wenn ein entsprechendes Objekt auf dem Weg positioniert ist. Dieser Aspekt wird in dieser Arbeit als *Hemmung* bezeichnet und wird im Formalismus der OACs nicht genügend abgedeckt. Somit würde die Plausibilitätsprüfung basierend auf Affordanz mögliche Widersprüche vernachlässigen.

Im Situationskalkül-basierten Planen, bspw. mittels GOLOG, werden Handlungen durch spezifische Vorbedingungen eingeschränkt, die erfüllt sein müssen, damit die jeweilige Aktion im Plan berücksichtigt werden

kann. Analog können während der Plausibilitätsprüfung solche Vorbedingungen rückwirkend evaluiert werden, wobei Erkennungsergebnisse und Umweltwissen im Zusammenhang betrachtet werden [Puls and Wörn, 2014].

Basierend auf der Bibliothek CASTOR zur Logikprogrammierung in C++ [Naik, 2008] wurde eine Regelbasis für Handlungsvorbedingungen definiert. In Tabelle 6.3 sind Beispiele für definierte Klauseln angegeben. In Anlehnung an die Nomenklatur von GOLOG werden Vorbedingungen durch das Prädikat *Poss* ausgedrückt. Basierend auf der Prolog-Syntax werden in Tabelle 6.3 Variablen durch Großbuchstaben dargestellt. Wie zu erkennen ist, variiert die Komplexität der Regeln abhängig von verknüpfter Affordanz und Hemmung. Gleichmaßen müssen Abhängigkeiten zwischen Handlungen berücksichtigt werden, da manche Aktionen nicht gleichzeitig ausgeführt werden können. Es sollte somit ein Widerspruch aufgedeckt werden wenn nach der Erkennung der Aktion *Sitzen* die Aktion *Gehen* erkannt wird. Entsprechend muss das Weltmodell und die Wissensbasis in jedem Zyklus aktualisiert werden.

**Tabelle 6.3:** Beispiele für definierte Klauseln von Handlungsvorbedingungen.

<code>Poss(hocken)</code>	<code>:= wahr;</code>
<code>Poss(hinsetzen)</code>	<code>:= Stuhl(O) ∧ InUmwelt(O) ∧ Hinter(O, mensch) ∧ Macht(mensch, stehen);</code>
<code>Poss(links_gehen)</code>	<code>:= Objekt(O) ∧ ( Hinter(O, mensch)     ∨ Rechts(O, mensch)     ∨ Vor(O, mensch)     ∨ Weit(O, mensch) ) ∧ ¬Macht(mensch, sitzen) ∧ ¬Macht(mensch, hocken);</code>
<code>Poss(Werkzeug_nehmen)</code>	<code>:= Tisch(O) ∧ ( Seite(O, mensch)     ∨ Vor(O, mensch) );</code>

Im Falle eines Widerspruchs, d.h. der fehlerhaften Auswertung der Vorbedingung einer Aktion, sind verschiedene Konsequenzen denkbar. Einerseits war die Erkennung eines Objekts, einer Handlung oder beider fehlerhaft. Entsprechend könnten Erkennungsalgorithmen erneut aufgerufen werden, wobei andere Parametersätze einsetzbar wären, oder rechenintensivere Algorithmen könnten eingesetzt werden. Andererseits kann die Indikation eines Widerspruchs genutzt werden, um das Roboterverhalten anzupassen. Da in einem solchen Fall die Folgerungen der Situationsanalyse auf fehlerhaften Grundlagen beruht, sollte ein größeres Risiko angenommen werden. Dieser letztgenannte Ansatz wird in dieser Arbeit weiterverfolgt.

### 6.2.3 Adaptierte Risikobeurteilung

Im Falle der Erkennung einer Inkonsistenz von Erkennungsergebnissen sollte eine Abschwächung der Risikobeurteilung verhindert werden. Dies wird erreicht, in dem die Eingabe des kooperativen Modus auf einen *nicht-kooperativen Zustand* festgelegt wird. Somit wird jeder abschwächende Einfluss außer Kraft gesetzt.

Auf Grund möglicher unsteter Erkennungsergebnisse können auch die Ergebnisse der Plausibilitätsprüfung wechselhaft und erratisch sein. Folglich kann sich dieses erratische Verhalten auf die Risikobeurteilung ausweiten. Als Gegenmaßnahme kann die Eingangsvariable bezüglich des kooperativen Modus nicht direkt auf einen *kooperierenden Zustand* gesetzt werden, nachdem eine unplausible Kombination von Objekten und Handlungen erkannt wurde. Stattdessen wird die Unschärfe des Kooperationsmodus bei der Risikobeurteilung ausgenutzt, indem der Wert der Variablen stetig über einige Zyklen hinweg an den Zielwert, der von der Situationsanalyse vorgegeben ist, angeglichen wird. Ein erhöhter Wert erhöht auch den abschwächenden Einfluss.

### 6.2.4 Evaluation und Diskussion

Zur Analyse des Systemverhaltens mit Plausibilitätsprüfung wurden verschiedene Untersuchungen durchgeführt. Diese werden im Folgenden dargestellt.

### 6.2.4.1 Experimentaufbau

Das Gesamtsystem wurde mit aufgezeichneten Sensorwerten ausgeführt. Anhand der rekonstruierten menschlichen Kinematik wurden Handlungen erkannt, Situationen analysiert und das Risiko beurteilt. Die Plausibilitätsprüfung wurde auf zwei Arten durchgeführt: Zum einen wurden die resultierenden Werte der anderen Module so ausgewertet, wie es das Ziel der Überprüfung ist. Zum anderen wurden synthetisch unplausible Ereignisse eingeführt, die häufig auftraten, um das System bezüglich Robustheit zu untersuchen.

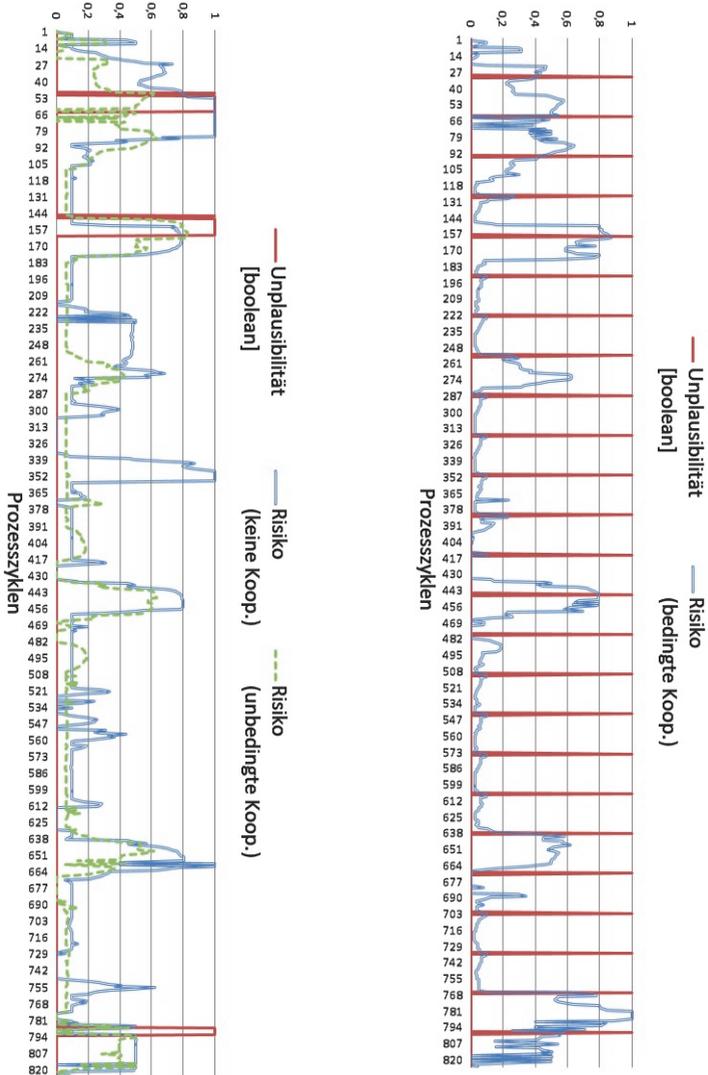
Als Aufgabe für den Roboter wurden zwei Trajektorien festgelegt:

- Eine sogenannte *Pick and Place*-Aufgabe, wobei der Roboter wiederholt von einer Seite des Arbeitsbereichs zur anderen verfahren musste, d.h. von Arbeitsplatz A zu B in Abbildung 5.1. Somit durchquerte der Roboter möglicherweise vom Menschen benötigten Raum.
- Eine simulierte Objektübergabe, wobei der Roboter wiederholt von Arbeitsplatz B in die Mitte des freien Arbeitsraumes verfährt, d.h. etwa zu der Position, an der der Mensch in Abbildung 5.1 dargestellt ist.

Des Weiteren wurde für jeden Testdurchgang der Kooperationsmodus a-priori festgelegt, um mögliche Einflüsse der adaptierten Risikobeurteilung feststellen zu können. Somit wurde der Zielwert der Eingangsvariablen bezüglich des kooperativen Modus nicht durch die Situationsanalyse vorgegeben. Der Wert konnte jedoch durch das Resultat der Plausibilitätsprüfung, wie oben beschrieben, angepasst werden.

### 6.2.4.2 Analyse mittels Sensordaten

In der Abbildung 6.4a sind einige Ergebnisse der Analyse mittels Sensordaten ersichtlich, wobei über 800 Prozesszyklen dargestellt sind. Am Anfang und am Ende der Aufzeichnung wurden bei der Plausibilitätsprüfung Inkonsistenzen festgestellt. Zwischen Zyklus 163 und 788 wurden jedoch keine Fehler detektiert. Insgesamt ist ersichtlich, dass die Notwendigkeit zur Behandlung von Inkonsistenzen bei Beobachtungen für reale Systeme gegeben ist.



(a) Verschiedene Kooperationsmodi und Plausibilitätsprüfung. (b) Bedingter Kooperationsmodus und künstlich hinzugefügte Widersprüche.

**Abbildung 6.4:** Risikowertanalyse.

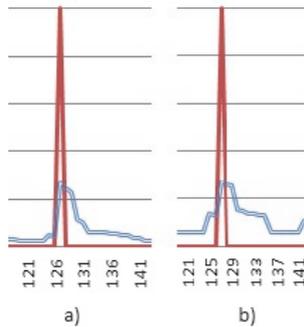
In dem dargestellten Testfall geht eine Person im Arbeitsraum umher. Ein Stuhl ist am Rand des freien Arbeitsraumes, nahe Arbeitsplatz A, positioniert. Somit wird in dem Beispiel die Komplexität reduziert, indem es um die Erkennung des Widerspruchs geht, dass Gehen durch den Stuhl verhindert wird. In der dargestellten Sequenz in Abbildung 6.4a treten in 4% der Zyklen Widersprüche auf.

Abbildung 6.4a zeigt ebenfalls Risikowerte für verschiedene Kooperationsmodi. Ohne Kooperation sind die Werte generell höher im Vergleich zum dargestellten unbedingten Kooperationsmodus. Ab Zyklus 147 kann der Effekt der Adaption der Eingangsvariable bezüglich des kooperativen Modus klar erkannt werden. Der generierte Risikowert ist dem ohne Kooperation sehr ähnlich. Nachdem der Widerspruch aufgelöst ist, fällt der Risikowert innerhalb von 4 Zyklen ab. Ein ähnlicher Vorgang ist ab Zyklus 49 zu erkennen, wobei der Risikowert bei unbedingter Kooperation nicht den Wert ohne Kooperation erreicht. Dieses Verhalten lässt sich auf die veränderte Roboterbewegung vor dem Feststellen des Widerspruchs zurückführen, da dieses direkt von der Risikobeurteilung abhängig ist.

### 6.2.4.3 Analyse mittels synthetischer Inkonsistenzen

Das Verhalten der Risikobeurteilung mit häufigen synthetischen Inkonsistenzen ist in Abbildung 6.4b dargestellt. Eine Inkonsistenz wurde alle 32 Zyklen künstlich eingefügt. Generell ist mit dem Auftreten einer Inkonsistenz ein Ansteigen des Risikowertes verbunden. Anschließend fällt der Wert wieder ab. Durch die Zielvorgabe für den Kooperationsmodus der bedingten Kooperation wird dieses Verhalten besonders anschaulich, da dieser Wert die größte Minderung des Risikowertes verursacht. Bei der Rückführung des Wertes der Kooperationsmodusvariablen von *keine Kooperation* zu *bedingte Kooperation* wird der Zustand *unbedingte Kooperation* durchlaufen. Somit tritt dieses Verhalten auch für diesen Modus auf, ist jedoch nicht so offensichtlich.

In Abbildung 6.5 wird dieser Umstand verdeutlicht. In Abbildung 6.5 a) ist die Zielvorgabe *bedingte Kooperation*, bei b) ist es *unbedingte Kooperation*. Ein Unterschied liegt darin, dass im bedingten Modus die Risikobeurteilung generell geringere Werte annehmen kann. Ein anderer ist die Abfallrate des Risikowertes nach dem Auflösen einer Inkonsistenz.



**Abbildung 6.5:** Direkter Vergleich des Einflusses von Zielvorgabe für den Kooperationsmodus.

#### 6.2.4.4 Diskussion

Das beschriebene System ermöglicht die Detektion von inkonsistenten Beobachtungen, die als Ergebnisse anderer Systemmodule zur Verfügung stehen. Es ist deutlich, dass eine Behandlung von solchen beobachteten Widersprüchen notwendig ist, da Erkennungsalgorithmen keine Fehlerfreiheit garantieren können.

Die modellierte Regelbasis behandelt zum einen den Aspekt von Affordanz durch die Berücksichtigung von für eine Handlung notwendigen Objekte. Zum anderen werden auch für eine Aktion relevante Hemmnisse eingebunden. Kritisch zu betrachten ist dabei, dass für die Affordanz nicht-relevante Objekte durch den Aspekt der Hemmung relevant werden können. Im Allgemeinen ist es somit notwendig, dass bei der Regelerstellung bezüglich einer Aktion alle Objekte und deren Interaktionsmöglichkeiten betrachtet werden. Bei entsprechender Systemkomplexität kann dieser Aspekt die händische Regelerstellung unmöglich machen. Für einen solchen Fall ist das Lernen und Abstrahieren von Aktion-Objekt-Relationen wünschenswert, wie es grundlegend im Konzept von OACs möglich ist, wobei in erster Linie Affordanzen jedoch keine Hemmungen ermittelt werden. Ein interessanter Aspekt ist dabei die Untersuchung unsachgemäßer Benutzung von Gegenständen, wie bspw. das Sitzen oder Stehen auf einem Tisch.

Die Behandlung erkannter Widersprüche ermöglicht die direkte Anpassung des Systems. Die in dieser Arbeit untersuchte Anpassung des Roboterverhalten legt den Grundstein für das sichere Verfahren des Roboters bei fehlerhafter Handlungs- und Objekterkennung. Die zusätzlich zur Verfügung stehenden Informationen können jedoch auch genutzt werden, um algorithmische Anpassungen bei den Erkennungsmodulen durchzuführen. Denkbar ist hierbei das Ausführen rechenintensiver Algorithmen, um vorliegende Erkennungsergebnisse zu widerlegen oder zu bestätigen. Darüber hinaus geben diese Informationen dem Systementwickler Mittel an die Hand, um das System zielgerichtet zu verbessern.

### 6.3 Reaktives Roboterverhalten

Die Möglichkeiten zur Reaktivität eines Roboters werden durch die Regelbarkeit festgelegt, d.h., wie schnell neue Zielvorgaben erreicht werden können. Leichtbauroboter, wie bspw. der LBR von KUKA, sind in der Lage mit entsprechender Sensorik und Regelungsverfahren schnell auf äußere Einflüsse zu reagieren. Dieser Umstand wird deutlich von Haddadin et al. aufgezeigt (z. B. [Haddadin et al., 2012]), indem erst bei Kollision auf diese reagiert wird. Jedoch lässt sich diese Vorgehensweise nicht auf generelle Industrieroboter verallgemeinern. Folglich sind weiterführende Ansätze notwendig, die Kollisionen von vornherein vermeiden. Forschungen in der Bahn- und Bewegungsplanung ermöglichen die Umsetzung solcher Ansätze.

#### 6.3.1 Stand der Forschung

Bahn- und Bewegungsplanung für Roboter sind wichtige Voraussetzungen damit sich Roboter autonom durch ihre Umwelt bewegen können. Verschiedene Varianten haben sich bei der Bahnplanung entwickelt. Zum einen gibt es rein reaktive Planer, wie bspw. verhaltensbasierte Pfadplaner [Arkin, 1998] oder Potentialfeldmethoden [Khatib, 1986]. Diese Verfahren basieren auf lokalen Informationen über die Umgebung und sind daher anfällig für lokale Minima. Zum anderen gibt es deliberative Planer, welche globale Informationen verarbeiten. Diese Planer basieren auf Graphsuchen, wobei

der Graph den Konfigurationsraum des Roboters repräsentiert [LaValle, 2006]. Um schnelle online Planung zu erreichen, wird in einer offline Phase die Graphenstruktur bestimmt.

In der Arbeit von [Kunz et al., 2010] wird eine deliberative Planungsmethode vorgestellt, die auf sogenannten *Dynamic Roadmaps* basieren [Leven and Hutchinson, 2002]. Das System wurde für einen Roboterarm auf einer mobilen Plattform für serviceorientierte Szenarien implementiert. Folglich sollen Armbewegungen auf Änderungen der Umwelt reagieren und der Bewegungspfad entsprechend bestimmt werden. Als Sensor wird eine Time-of-Flight Kamera verwendet, um Hindernisse zu detektieren und ein Voxelmodell der Umgebung zu aktualisieren. Das Voxelmodell wird wiederum zur Aktualisierung des Suchgraphen genutzt. Der Ansatz erreichte Planungszeiten mit weniger als 100 ms, was von den Autoren als schneller als menschliche Reaktionszeiten dargelegt wird.

Ein hybrider Ansatz von reaktiver und deliberativer Planung wird durch [Yoshida and Kanehiro, 2011] vorgestellt. Ein roadmapbasiertes Verfahren wird eingesetzt, um einen Pfad vor der Ausführung zu bestimmen. Während der Bewegung wird das Umgebungsmodell mit veränderten geometrischen Daten aktualisiert. Um Kollisionen zu vermeiden, wird der Pfad anhand der lokalen Informationen aus dem Umweltmodell angepasst. Falls die Anpassung den Pfad zu sehr ändern würde, wird eine komplette Neuplanung des Pfades mittels deliberativer Methode durchgeführt. Der Zweck der Pfadanpassung ist es, die Zahl der Planungsaufrufe zu reduzieren und entsprechend die Systemperformanz zu steigern. Die vorgestellten Ergebnisse wurden ausschließlich durch Simulation bestimmt und demonstrieren die Gültigkeit des Ansatzes.

Ein anderer Ansatz zur Performanzsteigerung wird durch [Pan and Manocha, 2011] verfolgt, indem GPU-basierte parallele Algorithmen zur Kollisionsberechnung eingesetzt werden. Dadurch können mehrere Roboterkonfigurationen simultan evaluiert werden, was effiziente Kollisionsabfragen erlaubt. Ähnlich zu [Kunz et al., 2010] werden kollisionsfreie Pfade in weniger als 100 ms bestimmt.

## 6.3.2 Die Bahnplanungsmethode

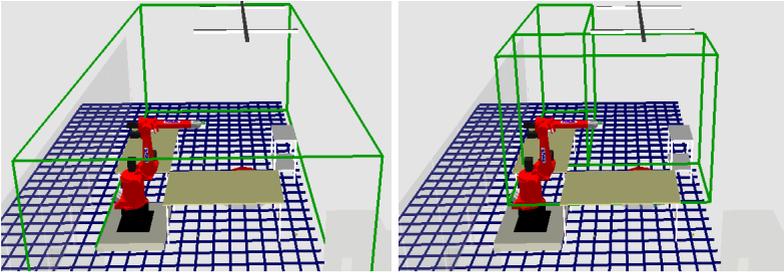
Die Planungsmethode ist angelehnt an dynamischen Roadmaps, indem zwei Phasen durchlaufen werden. In der Offline-Phase wird der Konfigurationsraum erfasst und auf eine Graphenstruktur abgebildet. Während der Online-Phase findet die eigentliche Planung statt, indem der Graph aktualisiert und durchsucht wird.

### 6.3.2.1 Die Offline-Phase

In der Forschung des letzten Jahrzehnts wurden Methoden populär, die mittels randomisierter Abtastung den freien Konfigurationsraum erfassen. Diese Methoden verbesserten die Performanz, wobei nicht nach Optimalität gestrebt wird [LaValle, 2006].

In der vorliegenden Arbeit ist der Arbeitsraum dynamisch, speziell wegen der Bewegung und Interaktion des Menschen. Folglich kann der freie Konfigurationsraum nicht vom allgemeinen Konfigurationsraum unterschieden werden. Somit wurde auf die randomisierte Abtastung verzichtet und eine systematische sowie deterministische Abtastungsmethode eingeführt. Darüber hinaus können virtuelle Zäune definiert werden, um den Arbeitsraum einzuschränken. Dies ist vor allem dann nötig, wenn sich statische Objekte, die nicht an der Interaktion teilhaben, im Arbeitsraum positioniert sind, wie bspw. Wände oder Kontrollschranken. Auf diese Weise kann ebenfalls der Arbeitsraum dem Interaktionsszenario angepasst werden (siehe Abb. 6.6).

Falls es eine Arbeitsraumaufteilung gibt und diese a-priori bekannt ist, kann in der Offline-Phase ein Geschwindigkeitsprofil für den Arbeitsraum definiert werden (siehe Abb. 6.7 links). Bei der systematischen Abtastung des Konfigurationsraumes wird die Pose des Roboters für jede Konfiguration im Arbeitsraum berechnet, wodurch die Korrelation mit dem Geschwindigkeitsprofil erreicht wird (siehe Abb. 6.7 rechts). Anhand dessen werden die Knoten des Konfigurationsraumgraphen gewichtet, wobei die Gewichtung während der Online-Phase in der Graphensuche ausgewertet wird. Liegt keine a-priori Geschwindigkeitsverteilung vor, werden die Knoten entsprechend mit 1 gewichtet. Um die Graphenerstellung abzuschließen, werden



**Abbildung 6.6:** Virtuelle Zäune zur Anpassung des Arbeitsraumes an den Produktionsprozess.

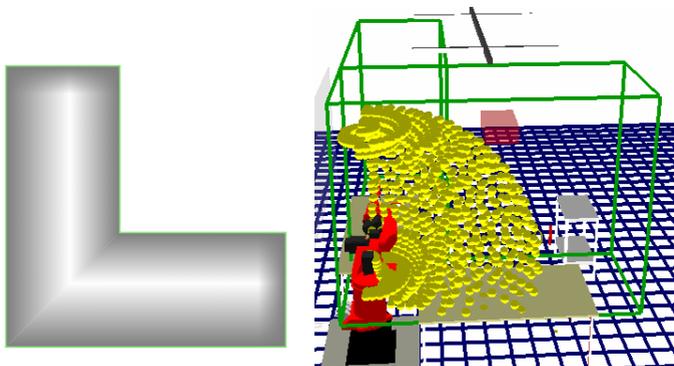
die erstellten Knoten mittels  $k$ -nächste-Nachbarn-Methode miteinander verknüpft.

### 6.3.2.2 Die Online-Phase

Die dynamische Roadmap Methode, wie sie von [Leven and Hutchinson, 2002] vorgeschlagen wird, definiert eine Abbildung vom Arbeitsraum auf die in der Roadmap enthaltenen Konfigurationen. Hierzu wird der Arbeitsraum in Zellen aufgeteilt und jede Zelle mit denen in ihr enthaltenen Knoten und Kanten verknüpft. Zur Laufzeit wird die Belegung der Zellen überprüft und falls nötig, werden die korrespondierenden Knoten als ungültig markiert.

Dieses Vorgehen weist zwei Nachteile auf, die der im Rahmen der vorliegenden Arbeit beschriebenen Ansatz vermeidet:

- Alle Konfigurationen, die mit einer Zelle verknüpft sind, werden als ungültig markiert. Somit führt eine kleine Überdeckung in der Zelle zu möglicherweise vielen ungültigen Knoten. Dieser Aspekt wird darüber hinaus durch die gewählte Zellengröße bestimmt.
- Die Knotenmarkierung basiert allein auf geometrischer Überdeckung, d.h. nur physische Objekte beeinflussen die Graphaktualisierung. Weitere Informationen, wie bspw. situationsabhängige Risikobeurteilung, können nicht abgebildet werden.



**Abbildung 6.7:** Links: Schematische Darstellung eines möglichen Geschwindigkeitsprofils als grauwertkodierte Überlagerung auf den Arbeitsraum. Die Arbeitsraumgrenze ist grün dargestellt. Weiß bedeutet maximale Geschwindigkeit, schwarz stellt minimale Geschwindigkeit dar. Rechts: Darstellung der im abgetasteten Konfigurationen im Arbeitsraum. Die mögliche Erreichbarkeit im Arbeitsraum wird deutlich.

Wie in [Graf et al., 2009] eingeführt und in [Puls et al., 2012a] ausgeführt, kann die Rekonstruktion der menschlichen Kinematik für schnelle Kollisionsberechnung genutzt werden, indem die relevanten Kinematiken durch Kugelhüllmodelle konvex approximiert werden. Somit stehen während der Suche zwei Kriterien zur Verfügung, um einen Knoten des Konfigurationsgraphen als *kollidierend* einzustufen: Distanz zwischen Roboter und Mensch sowie die situationsabhängige Risikobeurteilung. Liegt einer dieser Werte unter bzw. über jeweils festgelegten Schwellwerten, wird eine kollidierende Konfiguration angenommen. Der entsprechende Knoten im Graphen wird markiert und nicht weiter in der Suche betrachtet. Durch die Erkennung von Arbeitsraumobjekten können ebenfalls kollidierende Konfigurationen mit Hindernissen ermittelt und entsprechend markiert werden. Da diese Objekte jedoch nicht in die Risikobeurteilung eingehen, ist die Berücksichtigung der Distanz ein wichtiger kollisionsvermeidender Faktor während der Suche.

Ein wesentlicher Aspekt, der die Laufzeitanforderung und Pfadqualität beeinflusst, stellt die verwendete Heuristik der A\*-Suche dar. In den Arbeiten

von [Kunz et al., 2010] und [Leven and Hutchinson, 2002] werden Arbeitsraummetriken eingesetzt. Im Gegensatz dazu wird in der vorliegenden Arbeit die Konfigurationsraummetrik als euklidische Distanz verwendet. Da verschiedene Konfigurationen die gleiche Pose im Arbeitsraum beschreiben, ist eine einfache Arbeitsraummetrik basierend auf TCP-Positionen ungeeignet. Entsprechend können Arbeitsraummetriken entwickelt werden, die eine Menge von Referenzpunkten auf der Roboteroberfläche zusätzlich zum TCP betrachten. Dieser Ansatz wird von Kunz et al. sowie Leven und Hutchinson verfolgt. Jedoch wird ein Pfad im Konfigurationsraum direkter abgebildet, da kürzere Winkeldistanzen in effizienteren Pfaden resultieren. Somit wird die euklidische Metrik direkt auf den Konfigurationen angewendet, wodurch zusätzliche Berechnungen vermieden werden.

Die Risikobeurteilung für eine gegebene Konfiguration und Situation kann ausgenutzt werden, um die Verfahrensgeschwindigkeit zu adaptieren. Der Risikowert  $R$  lässt sich als Sicherheitsbeiwert  $s$  umformulieren:

$$s = 1 - R. \quad (6.2)$$

Somit gilt:

$$s \in [0, 1]. \quad (6.3)$$

Folglich kann  $s$  direkt als Multiplikator für die einzustellende Geschwindigkeit  $v_{soll}$  genutzt werden:

$$v_{soll} = v_{max} * s. \quad (6.4)$$

Da eine Zielgeschwindigkeit von 0 nicht zum Erreichen des Zieles beiträgt, wird  $s$  auf das Intervall  $[\epsilon, 1]$  angepasst:

$$s = \begin{cases} 1 - R, & \text{falls } R < 1 - \epsilon \\ \epsilon, & \text{sonst} \end{cases}. \quad (6.5)$$

Somit stellt  $\epsilon$  den minimalen Multiplikator dar, so dass die resultierende minimale Geschwindigkeit den Sicherheitsbedürfnissen entspricht. Da die Risikobeurteilung ebenfalls Bezug zur Robotergeschwindigkeit herstellt, kann während der Suche die Geschwindigkeit bei der Risikobeurteilung als Parameter  $v_{risiko}$  angegeben werden. Liegt der resultierende Risikowert über dem Kollisionsschwellwert, wird der Geschwindigkeitsparameter inkrementell erniedrigt, bis die Konfiguration als ausreichend sicher ein-

gestuft wird, d.h. der Wert unter dem Schwellwert liegt. Die tatsächlich resultierende einzustellende Geschwindigkeit  $\hat{v}_{soll}$  ergibt sich somit aus:

$$\hat{v}_{soll} = \min(v_{soll}, v_{risiko}). \quad (6.6)$$

Bei der Graphensuche mittels A\*-Algorithmus wird die Geschwindigkeit in die Knotenevaluationsfunktion  $f$

$$f(n_i) = g(n_i) + h(n_i) \quad (6.7)$$

eingebunden. Dabei gibt  $n_i$  den untersuchten Knoten im Graphen an. Die Funktion  $h$  stellt die Heuristik dar, die mittels euklidischer Distanzbestimmung eine optimistische Schätzung des Weges bis zum Ziel angibt. Die Funktion  $g$  gibt die bisherige Wegstrecke vom Startknoten  $n_0$  an und wird angepasst zu:

$$g(n_i) = \sum_{k=0, \dots, i-1} (\bar{v}_{k+1} \cdot \|n_k - n_{k+1}\|) \quad (6.8)$$

Durch  $\bar{v}$  soll die Strecke so gewichtet werden, dass diese bei langsamer Geschwindigkeit länger wird. Somit geht  $\hat{v}_{soll}$  indirekt in  $g$  ein. Für  $\bar{v}$  ergibt sich:

$$\bar{v} = \frac{v_{max}}{\hat{v}_{soll}}. \quad (6.9)$$

Die Heuristik  $h$  muss nicht angepasst werden, da diese weiterhin ein optimistisches Orakel darstellt. Für die Anwendbarkeit des A\*-Algorithmus ist eine optimistische Heuristik notwendig. Falls sich kein Mensch im Arbeitsraum befindet und die resultierende Sollgeschwindigkeit maximal ist, wird die bestimmte Distanz nicht verändert. Bei Knoten, die direkt mit dem Ziel verknüpft sind, stimmt somit  $h$  mit dem tatsächlichen Wert überein. Falls ein Risiko größer 0 vorliegt, wird die betrachtete Distanz durch Gewichtung mit verlangsamter Geschwindigkeit größer. Die Heuristik  $h$  ist somit kleiner als die gewichtete Distanz. Somit ist  $h$  weiterhin optimistisch.

Auf Grundlage dieser Anpassung der Funktion  $g$  werden schnellere Pfade bevorzugt. Dennoch sind weniger sichere Passagen möglich, da eine sichere Geschwindigkeit eingehalten wird.

### 6.3.3 Reaktivität durch Bewegungsplanung

Die Bewegungsplanung erhält als Eingabe einen geplanten Pfad des Roboters. Der Pfad besteht aus Knoten aus dem Suchgraphen, die in der Bewegungsplanung als *Stützstellen* genutzt werden. Mittels dieser Stützstellen kann ein Weg zwischen Start- und Zielkonfiguration interpoliert werden, so dass Interpolationsschritte als Zwischenziele an den Roboter übertragen werden können. Dabei werden die Geschwindigkeitsvorgaben, die mit den Stützstellen verknüpft sind, berücksichtigt. Auf diese Weise kann ein geplanter Pfad direkt durch den Roboter verfahren werden, jedoch wird Reaktivität nicht erreicht.

Um auf den sich bewegenden Menschen zu reagieren, ist es notwendig, dass auf mögliche bevorstehende Kollisionen geprüft wird. Es ist nicht effizient, Kollisionsbetrachtungen für den gesamten Pfad zu berechnen, da möglicherweise kollisionsbehaftete Konfigurationen am Ende des Pfades nicht mehr notwendigerweise zu Kollisionen führen sobald der Roboter diese Konfiguration erreicht. Somit werden mittels einer begrenzten Vorausschau mögliche Kollisionen entlang des Pfades bestimmt. Durch Pfadinterpolation für eine festgelegte Wegstrecke werden anhand der Interpolationsschritte Distanz zum Menschen sowie Objekten und die Risikobeurteilung berechnet. Auf Grund der Diskretisierung durch die Interpolation können Lücken bei der Überprüfung auftreten [Graf et al., 2009]. Entsprechend wird bei der vorausschauenden Distanzbestimmung ein Sicherheitsabstand eingeführt.

Falls durch die Vorausschau eine mögliche Kollision erkannt wird, wird der Roboter zum Stillstand abgebremst und eine neue Bahnplanung wird angestoßen. Die neue Planung erhält als Startkonfiguration die aktuelle Roboterkonfiguration, wobei das Ziel unverändert bleibt. Sobald die Bahnplanung einen neuen Pfad gefunden hat, wird dieser interpoliert und abgefahren.

Während der Suche werden die Kernberechnungen innerhalb des Rahmenwerks mit hoher Priorität ausgeführt. Dadurch können Sensordaten zeitnah verarbeitet werden und der Roboter sicher gestoppt werden.

### 6.3.4 Evaluation und Diskussion

Zur Evaluation wurden zwei Varianten der Knotenevaluation während der Online-Phase gegenübergestellt:

**Variante 1** Funktion  $g$  mit reiner Auswertung der euklidischen Distanz,

**Variante 2** Die vorgestellte Funktion  $g$  mit Berücksichtigung der Geschwindigkeit.

Für die Analyse wurde eine Sequenz eines sich durch den Arbeitsraum bewegenden Menschen aufgezeichnet und zyklisch wiedergegeben. Während der Sequenz, die aus 2000 Bildern besteht, hat sich die Person frei im Sensorbereich bewegt. Die Bewegung enthält lineare und zirkuläre Pfade mit gewöhnlichen und unterschiedlichen Bewegungsgeschwindigkeiten. Insgesamt wurden je Variante 9000 Bilder zur Analyse wiedergegeben.

Während der Evaluation wurden verschiedene Parameter aufgezeichnet, die in Tabelle 6.4 aufgeführt sind. Dabei wurden Planungen auf den ersten drei Achsen des Roboters vorgenommen und die letzten drei Achsen entsprechend ohne Planung zwischen Start- und Zielkonfiguration interpoliert. Um die Effektivität der Umplanung abzubilden, sind speziell die Anzahl an beendeten und abgebrochenen Pfaden von Interesse. Abgebrochene Pfade sind solche, die auf Grund von Umplanungsauffufen unterbrochen wurden, bevor der Roboter sein Ziel erreicht hat. Die große Diskrepanz zwischen Variante 1 und 2 bezüglich abgebrochener Pfade wurde hauptsächlich dadurch verursacht, dass gefundene Pfade direkt im nächsten Zyklus bereits wieder durch einen Umplanungsauffufen abgebrochen wurden. Somit wurden Pfade gefunden, die im Grenzbereich zum risikobasierten Kollisionsausschluss lagen. Durch geringe Menschbewegungen wurden diese dann bereits wieder als zu riskant eingestuft. Somit wirkt sich zukünftige Risikoerhöhung drastisch aus. Dieser Aspekt wird durch die Anzahl der erfolgreichen Suchen bestätigt. Auf der anderen Seite erlaubt die Variante 2 einen größeren Sicherheitsabstand oder ein verlangsamtes Verfahren. Menschliche Bewegungen im Arbeitsraum wirken entsprechend mit größerem Spielraum.

Die Anzahl der beendeten Pfade gibt folglich die Häufigkeit an, mit der der Roboter sein Ziel erreichte.

**Tabelle 6.4:** Auszug der Evaluationsergebnisse zur reaktiven Bahnplanung aus [Puls et al., 2012a].

Parameter	Variante 1	Variante 2
# beendete Pfade	44	46
# abgebrochene Pfade	109	24
# Suchen gesamt	433	316
# erfolgreiche Suchen	151	70
# erfolglose Suchen	282	246
⊙ Pfadlänge	17,7	19,1
⊙ Zeit für erfolgreiche Suchen [ms]	27,33	233,68
⊙ Zeit für erfolglose Suchen [ms]	294,89	371,44
⊙ Zeit für Suchen gesamt [ms]	201,58	340,92

Insgesamt erreicht die Bahnplanung mit angepasster Knotenevaluierung durch Berücksichtigung maximal erlaubter Geschwindigkeiten (Variante 2) eine etwas höhere Effektivität, da mehr Pfade beendet wurden. Die Anzahl der abgebrochenen Pfade ist etwa ein viertel im Vergleich zu Variante 1, bei der nur Abstandsbetrachtungen bei der *kürzesten Weg*-Suche berücksichtigt werden. Im Gegensatz dazu benötigt Variante 2 etwa acht mal mehr Zeit für erfolgreiche Suchen, jedoch werden nur etwas über zwei drittel so viele Suchen insgesamt angestoßen. Somit kann trotz erhöhter Rechenleistung eine bessere Effektivität erzielt werden.

Im Vergleich zu [Kunz et al., 2010] sowie [Leven and Hutchinson, 2002], kann festgestellt werden, dass langsamere Planung erreicht wird. Das ist auf die zusätzlichen Berechnungen zurückzuführen, die für die Bestimmung der direkten Kinematik des Roboters notwendig sind, um Abstände und Risiko bestimmen zu können. Somit ist die Reaktivität etwas langsamer, jedoch kann eine erhöhte Sicherheit erreicht werden. Darüber hinaus wurde in [Pan and Manocha, 2011] demonstriert, dass GPU-basierte Berechnungen im Bereich der Bahnplanung Laufzeiten verbessern können. Dieser Ansatz

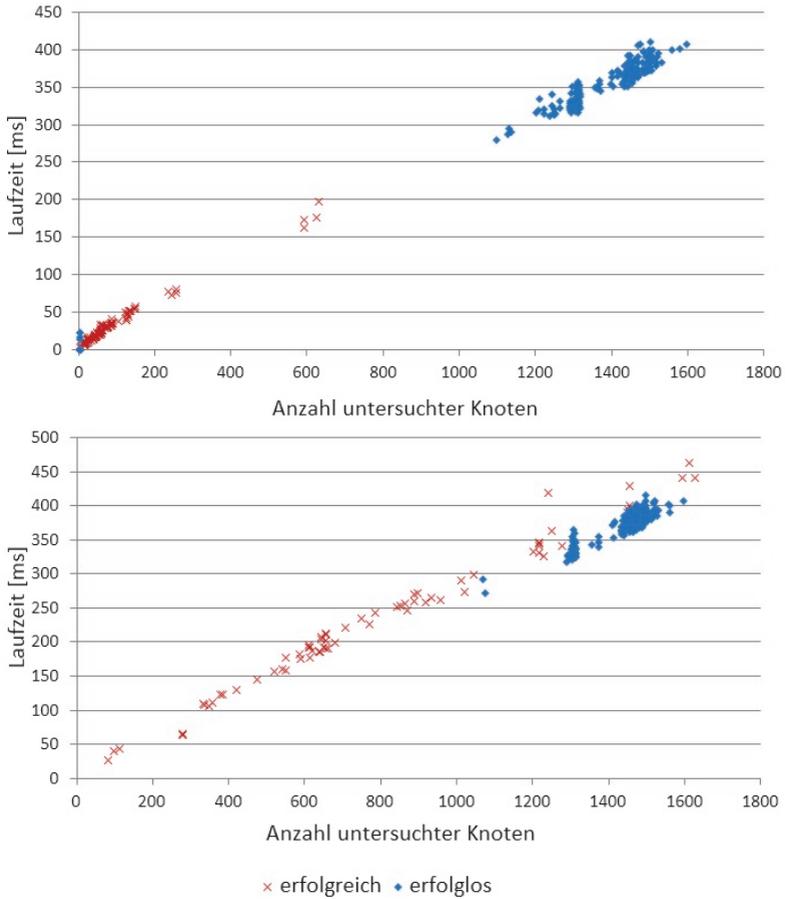
kann ebenfalls für die in dieser Arbeit aufgeführten Verfahren anwendbar sein.

In Abbildung 6.8 ist die Korrelation zwischen Anzahl der expandierten Knoten während der Suche und der Laufzeit bezüglich erfolgreicher und erfolgloser Suchen aufgeführt. Die lineare Beziehung wird offensichtlich. Für Variante 1 ergibt sich eine klare Trennung und Häufung. Suchen, die mehr als 1000 Knoten betrachteten führten zu erfolglosen Ergebnissen. Eine ähnliche Häufung bezüglich erfolgloser Suchen kann für Variante 2 erkannt werden. Jedoch decken erfolgreiche Suchen den gesamten Bereich ab, so dass auch eine hohe Anzahl an betrachteten Knoten zu einem gefundenen Pfad führen kann. Dieses Verhalten ist auf die Bevorzugung schnellerer Pfade im Vergleich zu kürzeren zurückzuführen. Somit wird für gefährliche und folglich langsame Passagen ein schnellerer Umweg gesucht.

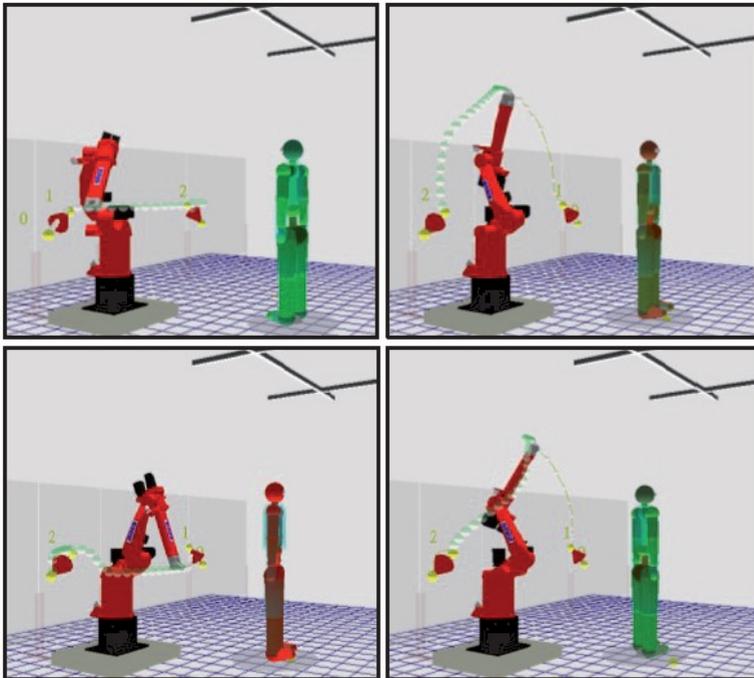
Verschiedene geplante Pfade sind in Abbildung 6.9 aufgeführt. Die Bahnen sind durch die TCP-Trajektorien symbolisiert. Das obere linke Bild zeigt die original geplante Bahn mit dem Menschen in großer Distanz dem Roboter zuschauend. Die menschliche Posen im oberen rechten und unteren linken Bild sind sehr ähnlich, jedoch unterscheiden sich die resultierenden umgeplanten Pfade wesentlich. Das obere rechte Bild stellt einen Pfad der Variante 2 dar, wobei die untere Reihe Pfade der Variante 1 veranschaulichen. Im unteren rechten Bild wird erkenntlich, dass der Mensch wesentlich an den original Pfad herantreten muss, damit ähnliche Ergebnisse wie für Variante 2 erzielt werden.

Wie in Abbildung 6.10 dargestellt, wird die Geschwindigkeit entsprechend angepasst, wenn der Roboter nahe dem Menschen verfährt. Die Geschwindigkeitsänderungen an den Stellen mit geringer Distanz zwischen Roboter und Mensch unterstreichen deren Notwendigkeit. Es zeigt auch, dass die Planung entsprechende Pfade als Ergebnis ausgibt.

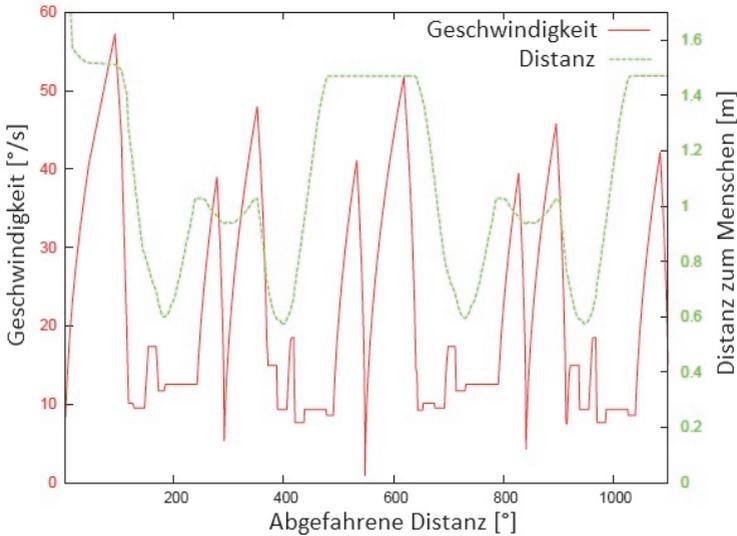
Die Ergebnisse haben die Notwendigkeit für eine reaktive Bahnplanung unterstrichen und die Möglichkeiten des untersuchten Ansatzes verdeutlicht. Risikoreiche Pfade werden gemieden, falls möglich. Andernfalls werden risikobehaftete Abschnitte mit entsprechend reduzierter Geschwindigkeit passiert.



**Abbildung 6.8:** Beziehung zwischen Laufzeit und Anzahl untersuchter Knoten bezüglich erfolgreicher und erfolgloser Suchen. Oben: Variante 1. Unten: Variante 2.



**Abbildung 6.9:** Beispiele für geplante Bahnen. Oben links: Original Plan. Oben rechts: Umplanung der Variante 2. Untere Reihe: Umplanungen der Variante 1 für verschiedene menschliche Positionen.



**Abbildung 6.10:** Robotergeschwindigkeit und Distanz zum Menschen.

Die untersuchten Laufzeitaspekte der Bahnplanung sind vielversprechend, jedoch ist eine Performanzsteigerung notwendig, um für Nutzer und Werker Reaktivität gewährleisten zu können. Davon unabhängig ist die Gewährleistung der Sicherheit, da das System den Roboter zu einem Stopp veranlasst, sobald eine Umplanung angestoßen wird. Abgesehen von Leistungssteigerungen von Rechnersystemen sind weitere Optimierungen möglich. GPU-basierte Verfahren erlauben beschleunigte Berechnungen [Pan and Manocha, 2011]. Ebenfalls kann der Ansatz von [Yoshida and Kanehiro, 2011] im Kontext der Risikobetrachtung weiter verfolgt werden, indem Planungsaufrufe durch Pfadanpassung vermieden werden. So wäre eine Möglichkeit, durch eine reine menschenzentrierte Risikobetrachtung Potentialfeldmethoden einzusetzen, die reaktive Pfadanpassungen erlauben. Im Falle von zu großen Abweichungen kann auf die komplexe Roboter-Mensch-Betrachtung als Teil der Bahnplanung übergegangen werden.



# 7 Zusammenfassung und abschließende Diskussion

Im Rahmen der vorliegenden Arbeit wurden Verfahren und Ansätze entwickelt, die, orientiert am kognitiven Zyklus, durch Wahrnehmung, Schlussfolgerung sowie Ausführung ein Situationsverständnis verwirklichen und sichere Mensch-Roboter Kooperation ermöglichen. Im Folgenden werden die wesentlichen Ergebnisse der Arbeit zusammengefasst und diskutiert. Dabei wurden die Lösungen nicht in reiner Theorie oder Simulation untersucht, sondern in realer Umgebung verschiedenen nationalen und internationalen Forschergruppen sowie bei industrienahe Führungen an einem Demonstrator vorgestellt.

## 7.1 Zusammenfassung der Ergebnisse

Es wurde zu folgenden wesentlichen Aspekten entsprechende Lösungsansätze entworfen und untersucht.

**Wahrnehmung** Zur Erkennung von Szeneobjekten wurden autoassoziative neuronale Netze in Form von erweiterten Hopfield Netzen untersucht. Im Vergleich zur Erkennungsmethode basierend auf globalen Gabor-Merkmalen wurden bessere Ergebnisse erzielt. Dabei werden Erkennungsraten zwischen 73 % und 100 % erreicht, wobei speziell ähnliche Objekte untersucht wurden, die die Erkennung erschweren.

Zur semantischen Positionsabbildung des Menschen in der Arbeitszelle wurde mittels Clusteranalyse anhand von aufgezeichneten Trajektorien wesentliche Orte im Arbeitsraum identifiziert. Durch Abbildung der Cluster in Potentialfunktionen kann zusätzlich zur Positionsklassifizierung eine Ortsprädiktion durchgeführt werden, die angibt zu

welchem Ort im Arbeitsraum der Mensch sich bewegt. Bei der Positionsklassifizierung werden Erkennungsraten bis zu 100 % erreicht. Die Prädiktion erreicht Werte von 90,9 %.

Die Erkennung von menschlichen Aktionen wird zum einen durch den Einsatz von mehrschichtigen Hidden Markov Modellen (HMMs) realisiert. Zum anderen werden kinematische Zustandsänderungen des Menschen beim Schlussfolgern ausgewertet. Die Erkennung mittels mehrschichtiger HMMs erreicht Erkennungsraten von über 83 % und dient im wesentlichen dem Erzeugen von weiterem Kontext, in dem die Situation eingebettet ist.

**Schlussfolgern** Zur Situationsanalyse wurde eine Wissensbasis entwickelt, die zur beschreibungslogischen Verarbeitung der Ergebnisse der Wahrnehmung und Menschenkennung dient. Auf diese Weise können Rückschlüsse auf die vorliegende Situation gezogen werden. Darüber hinaus werden Zielsetzungen für das Robotersystem generiert, die durch eine Handlungsplanung basierend auf dem Situationskalkül weiterverarbeitet werden. Somit können die Handlungen des Roboters direkt beeinflusst und vorgegeben werden. Das beschreibungslogische System erreicht Verarbeitungsraten von durchschnittlich 4,2 Hz. Die reaktive Handlungsplanung erzielt Verarbeitungszeiten zwischen 0,5 ms und 1,3 ms.

**Ausführung und Sicherheit** Ein wesentlicher Aspekt der vorliegenden Arbeit ist die situationsabhängige Risikobeurteilung. Diese baut auf den Ergebnissen der Wahrnehmung und Schlussfolgerung auf. Als Grundlage dient eine Regelbasis für unscharfe Logik, die durch ein hemmendes System mit Regeln zur Risikoadaption auf Basis der Situationsanalyse erweitert wurde. Somit wird die räumlich enge Kooperation zwischen Mensch und Roboter ermöglicht, wenn die Situation es zulässt.

Darüber hinaus werden Erkennungsergebnisse der Wahrnehmung im Zusammenhang untersucht und deren Plausibilität überprüft. In Fällen von Unstimmigkeiten wird die Risikobeurteilung angepasst, so dass mögliche riskante Situationen vermieden werden.

Eine reaktive Bahnplanung ermöglicht angepasstes Roboterverhalten, indem die Resultate der Risikobeurteilung während der Roboterbewe-

gung überprüft werden und, falls nötig, eine Umplanung angestoßen wird. Die Umplanung minimiert das Risiko und gibt angepasste Geschwindigkeiten vor, so dass ein Verfahren des Roboters auch in ansonsten risikobehafteten Situationen möglich ist. Auf diese Weise wird eine höhere Effizienz des Prozesses erzielt. Die Berechnungszeiten für die Planung liegen durchschnittlich bei etwa 340 ms. Während der Planung wird der Roboter angehalten, damit mögliche, drohende Kollisionen vermieden werden.

**Prozessanalyse** Um Wissen aus den produktiven Prozessen, die mit dem vorgestellten System beobachtet werden, abzuleiten, dient die Prozessanalyse. Mittels der Integration und Adaption des *Heuristic Miner* können Prozessgraphen extrahiert werden, die bspw. für eine Prozessoptimierung oder -fehleruntersuchung genutzt werden können. Zudem wurde eine Methode entworfen und untersucht, die Prozessfolgen ermittelt. Am Beispiel der beobachtbare Ortswechsel des Menschen wurde die Prozessfolgenermittlung untersucht. Die extrahierten Abläufe können direkt in der semantischen Ortsprädiktion eingesetzt werden. Zur Evaluation wurden verschiedene aufeinander aufbauende Szenarien untersucht und entsprechende Qualitätskriterien entwickelt. Anhand dessen wurden Aussagen zu Rauschverhalten und notwendiger Datenmenge getroffen.

## 7.2 Abschließende Diskussion

Die erzielten Ergebnisse ermöglichen Situationsverständnis und daraus resultierend ein angepasstes Roboterverhalten. Das System analysiert die beobachtete Szene und extrahiert notwendiges Wissen, um Schlussfolgerungen ziehen zu können. Es hat sich gezeigt, dass dafür ein gesamtheitlicher Ansatz notwendig ist, da erst die Betrachtung von Zusammenhängen und Einordnung der Beobachtungen in ihren Kontext die vorhandene Komplexität der realen Umgebung abbilden kann. Im Rahmen der vorliegenden Arbeit wurde hierzu ein Beitrag geleistet, indem ein System entworfen und untersucht wurde, das die identifizierten, wesentlichen Aspekte des Kontextes betrachtet, wozu menschliche Handlungen, deren Ausführungsort,

Objekte in der Szene sowie die menschliche Pose mit der Beziehung zum Robotersystem gehören.

Bei der Erarbeitung der einzelnen Systemkomponenten wurde ein breites Spektrum verwandter Arbeiten hinzugezogen, woraufhin Algorithmen und Lösungsansätze übernommen, erweitert und angepasst wurden. Dabei wurde themenübergreifend vorgegangen, so dass interessante, neue Aspekte die Arbeit bereichern.

Als wesentlicher Teil der Arbeit ist die anhand der Situation angepasste Risikobeurteilung zu sehen. Am Beispiel industrieller Produktion wurde das Thema Sicherheit betrachtet und Konzepte zur Umsetzung erarbeitet. Dennoch sind die grundlegenden Ansätze und Untersuchungen auch für andere Robotersysteme, mit denen der Mensch engen Kontakt hat bzw. haben soll, von Bedeutung. Die Anwendung ist vor allem bei entsprechender Komplexität der Interaktion interessant, da einfache Systeme, die kaum mit ihrer Umwelt interagieren, häufig durch den Designprozess sicher gestaltet werden können.

Die wesentlichen Neuerungen in dieser Arbeit werden durch folgende Punkte charakterisiert:

- Gesamtheitliche Analyse der Situation und Folgerung von Zielsetzungen für das Robotersystem.
- Kontextuelle Plausibilitätsprüfung von beobachteten Handlungen.
- Situationsabhängige Risikobeurteilung.
- Reaktives Roboterverhalten durch risikoabhängige Bahn- und Bewegungsplanung.

### **Ausblick**

Mit dem vorgestellten System werden Grundlagen für eine zeitlich und räumlich nicht getrennte Mensch-Roboter Kooperation erarbeitet. Es können somit spezialisierte Bedarfsfälle dargestellt und analysiert werden. Weitere Arbeiten sind notwendig, um ein generalisiertes Gesamtsystem für die sichere Kooperation zu erreichen. Vor diesem Hintergrund sind folgende Aspekte mögliche Punkte, an denen weiterführende Arbeiten

ansetzen können, wobei diese bereits teilweise in den Betrachtungen der einzelnen Komponenten Erwähnung fanden.

**Sensorische Erfassung** Das vorgestellte System basiert auf der Verwendung eines einzelnen 3D Sensors, wodurch sich Einschränkungen für die Szenenanalyse ergeben. Die Erweiterung der sensorischen Erfassung stellt somit einen notwendigen Schritt dar. Hierbei sind vor allem die Aspekte der Sensorregistrierung bei verschiedenartigen Sensoren und Datenfusion wichtig. Damit einher geht auch die Erweiterung der Erkennung von Szeneobjekten, damit kleinere Handhabungsgegenstände erkannt, verfolgt und deren Benutzung registriert werden kann.

**Ortsbestimmung** Die weitere Betrachtung von kollinear angeordneten Arbeitsplätzen kann für die Verallgemeinerung des vorgestellten Ansatzes hilfreich sein.

**Handlungserkennung** Zusammen mit der Erweiterung der sensorischen Erfassung kann die Bestimmung der menschlichen Kinematik erweitert werden. Bei einer vollständigeren und genaueren Rekonstruktion können ebenfalls kleinere Handbewegungen und -gesten in die Erkennung eingebunden werden. Mögliche Ansätze hierzu wurden im entsprechenden Stand der Forschung aufgeführt.

**Schlussfolgerungssystem** Durch die Integration von unscharfer Logik in das Folgerungssystem können die Regeln zur Risikobeurteilung in die Situationsanalyse eingebettet werden. Jedoch ist dann auf die Laufzeitbetrachtung und weiterführende Optimierungen Wert zu legen, damit Sicherheitsaspekte mit entsprechender Geschwindigkeit berechnet werden. Eine andere Möglichkeit ist die Integration der Handlungsplanung durch Einführung eines Aktionsformalismus in das Folgerungssystem. Darüber hinaus ist die Wissensbasis in Hinblick auf einen Menschen und Roboter konzipiert. Die Erweiterung hinsichtlich mehrerer Kooperationspartner stellt einen wesentlichen Aspekt dar.

**Prozessanalyse** Eine weiterführende Möglichkeit besteht in der Identifizierung von kausalen Zusammenhängen zwischen Beobachtungen. Eine andere stellt die Analyse zur natürlich-sprachlichen Protokollgenerierung dar. Somit kann im Fehlerfalle oder zur Prozessoptimierung der

Mensch Verständnis für das Geschehene entwickeln, anstatt Zahlen- und Variablenkolonnen betrachten zu müssen.

**Risikobeurteilung** Die Anpassungsfähigkeit bezüglich fein-granularer Situationen kann weiter untersucht werden, um eventuell Prozesswissen in die Beurteilung einfließen zu lassen. So geben bspw. Prozessabweichungen oder sogar einzelne Prozessschritte Anlass die Beurteilung anzupassen.

**Plausibilitätsprüfung** Um die manuelle Regelerstellung von Aktion-Objekt-Beziehungen zu vermeiden, ist das Erlernen solcher Relationen notwendig. Dabei stellt das Lernen und Abstahieren unsachgemäßer Verwendung von Objekten eine besondere Hürde dar.

**Bahnplanung** Die Bahnplanung betrachtet zur Zeit einen Roboter im möglichen Beisein eines Menschen. Die Erweiterung durch die Berücksichtigung weiterer Kooperationsteilnehmer, sowohl Mensch als auch Roboter, kann für Anwendungen wichtig sein. Darüber hinaus können Konzepte entwickelt werden, die im Gegensatz zur reinen deliberativen Planung eine reaktive Komponente integrieren und somit beide Ansätze vereint. Auf diese Weise kann die benötigte Rechenzeit reduziert werden, indem weniger Umplanungen angestoßen werden müssen.

## Zur Hypothese

Es konnte in der Arbeit gezeigt werden, dass keine zeitliche und räumliche Trennung der Kooperation von Mensch und Roboter notwendig ist, da auf Grund der situationsabhängigen Risikobeurteilung ein entsprechend sicheres Roboterverhalten erzielt wird. Eine erhöhte Effizienz im Vergleich zum klassischen Systemstillstand bei Menscheneintritt in die Arbeitszelle kann erkannt werden. Die zukünftige Einbettung des Systems in industriennahe Anwendungsfälle und eine weiterführende, umfassende Analyse kann hierzu deutlichere Einblicke geben.

# Abbildungsverzeichnis

1.1	Überblick über die eigene Methode. . . . .	5
3.1	Layout der Roboterzelle, wie sie im MAROCO System verwendet wird. . . . .	20
3.2	Rekonstruktion der menschlichen Kinematik anhand von Tiefenbildern. . . . .	22
3.3	Übersicht der entwickelten Systemkomponenten. . . . .	24
4.1	Referenzpunkte und Trajektorie. . . . .	30
4.2	Roboter Reis RV6L mit Registriertkörper. . . . .	31
4.3	Sensordaten, links ungefiltert, rechts mit dem Amplitudenfilter gefiltert. . . . .	33
4.4	Fusionsergebnisse registrierter Punktwolken. . . . .	39
4.5	Links: Beistelltisch. Rechts: Schubladentisch. . . . .	40
4.6	Objektsegmentierung. . . . .	46
4.7	Tiefenabhängige Objektschichtung. . . . .	47
4.8	Beispiel eines Eingabemusters. . . . .	48
4.9	Zur Evaluation verwendete Objekte. Dargestellt sind Fotos und korrespondierende Tiefeninformation vom Sensor. . .	49
4.10	Konfusionsmatrix der Erkennungsergebnisse für Objekte in Bildmitte. . . . .	52
4.11	Konfusionsmatrix der Erkennungsergebnisse für Objekte am Bildrand. . . . .	53
5.1	Verfolgungsergebnis eines Menschen im Arbeitsraum (links). Arbeitsraumdarstellung von der Seite (rechts). . . . .	66
5.2	Normalisierte Gaußsche Funktionen als Repräsentanten der Cluster. . . . .	67
5.3	Datenfluss in der zweischichtigen HMM-Architektur (links). Orientierungsabbildung (rechts). . . . .	73

5.4	Modellierte Wissensbasis als ER-Modell. . . . .	79
5.5	Laufzeitanalyse. . . . .	90
5.6	Schematische Darstellung der Komponenten des Handlungsplanungs-systems. . . . .	93
5.7	Darstellung des Arbeitsraumes. Die verschiedenen symbolischen Orte sind durch Nummern markiert. . . . .	94
5.8	Kumulative Laufzeiten für Planungszyklen. . . . .	97
5.9	Beispiel eines Prozessmodells. . . . .	103
5.10	Schematische Darstellung verschiedener Handlungsabläufe. . . . .	108
5.11	Identifiziertes Prozessmodell. . . . .	109
6.1	System zur situationsabhängigen Hemmung der Risikobeurteilung. . . . .	116
6.2	Verhalten der Hemmung mit adaptiver Mittelung. . . . .	117
6.3	Risikoaussage und Betrachtung der Abstandseinhaltung. . . . .	120
6.4	Risikowertanalyse. . . . .	127
6.5	Direkter Vergleich des Einflusses von Zielvorgabe für den Kooperationsmodus. . . . .	129
6.6	Virtuelle Zäune zur Anpassung des Arbeitsraumes an den Produktionsprozess. . . . .	133
6.7	Geschwindigkeitsprofil und erreichbare Konfigurationen im Arbeitsraum. . . . .	134
6.8	Laufzeitanalyse. . . . .	141
6.9	Beispiele für geplante Bahnen. . . . .	142
6.10	Robotergeschwindigkeit und Distanz zum Menschen. . . . .	143

# Tabellenverzeichnis

4.1	Registrierung mit 10 cm durchmessenden Registrierkörper und unterschiedlichen Startpositionen der Roboterbewegungen. . . . .	38
4.2	Erkennungsraten ohne Vorklassifikation in Abhängigkeit der Größe der Schichten. . . . .	50
4.3	Erkennungsraten ohne Vorklassifikation in Abhängigkeit der Anzahl der Schichten. . . . .	54
4.4	Vergleich der Erkennungsergebnisse für verschiedene Vorklassifizierungen. . . . .	55
4.5	Vergleich der Laufzeitanalyse in Bilder pro Sekunde ohne (links) und mit Vorklassifizierung (rechts) und für verschiedene Objektpositionen: im Bild zentriert (Z) oder nahe des Bildrandes (R). . . . .	55
4.6	Erkennungsergebnisse der Objekterkennung für ausgewählte Objekte mittels Gabor-Merkmalen. . . . .	56
5.1	Beispiel für erzeugte Ausgabe der mehrschichtigen HMM-basierten Handlungserkennung. . . . .	75
5.2	Ergebnisse zur Verarbeitungsdauer bei Verwendung der Merkmalsvorgabe. . . . .	88
5.3	Ergebnisse zur Verarbeitungsdauer bei Verwendung von Sensordaten. . . . .	88
5.4	Beispiele für Weltmodelle mit entsprechenden Plänen. . . . .	95
5.5	Auszug der Protokolldaten zur Verdeutlichung der Plananpassung auf Grund von verändertem Weltmodell. . . . .	96
5.6	Variationen von additivem Rauschen. Das Qualitätskriterium wurde für verschiedene Rauschwerte evaluiert. . . . .	107
5.7	Variationen von Sequenzlängen. Das Qualitätskriterium wurde für verschiedene Längen und Rauschwerte evaluiert. . . . .	107

5.8	Variationen von Sequenzlängen. Das Qualitätskriterium wurde für verschiedene Längen und Szenarien evaluiert. . . . .	110
5.9	Variationen von Koeffizientenschwellwert. . . . .	110
5.10	Betrachtung der Sequenzentwicklung. . . . .	111
6.1	Auszug aus der grundlegenden Regelbasis zur Risikobestimmung. . . . .	114
6.2	Zusätzliche Regelbasis für die Hemmung des Risikowertes. . . . .	119
6.3	Beispiele für definierte Klauseln von Handlungsvorbedingungen. . . . .	124
6.4	Auszug der Evaluationsergebnisse zur reaktiven Bahnplanung aus [Puls et al., 2012a]. . . . .	139

# Literaturverzeichnis

- [Alami et al., 2014] Alami, R., Clodic, A., Chatila, R., and Lemaignan, S. (2014). Reasoning about humans and its use in a cognitive control architecture for a collaborative robot. In *Cognitive Architectures for Human-Robot Interaction Workshop at HRI*.
- [Anderson et al., 1977] Anderson, J. A., Silverstein, J. W., Ritz, S. A., and Jones, R. S. (1977). Distinctive features, categorical perception, and probability learning: Some applications of a neural model. *Psychological Review*, 82(5):413.
- [Arica and Yarman-Vural, 2001] Arica, N. and Yarman-Vural, F. (2001). An overview of character recognition focused on off-line handwriting. In *IEEE Trans. on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, pages 216–233.
- [Arkin, 1998] Arkin, R. C. (1998). *Behaviour Based Robotics*. Cambridge Univ. Press.
- [Asfour et al., 2006] Asfour, T., Regenstein, K., Azad, P., Schroder, J., Bierbaum, A., Vahrenkamp, N., and Dillmann, R. (2006). ARMAR-III: An integrated humanoid platform for sensory-motor control. In *6th IEEE-RAS International Conference on Humanoid Robots*, pages 169–175.
- [Ashbrook and Starner, 2002] Ashbrook, D. and Starner, T. (2002). Learning significant locations and predicting user movement with gps. In *Proceedings of the Sixth IEEE International Symposium on Wearable Computers*, pages 101–108.
- [Axelrod, 1985] Axelrod, R. (1985). *The Evolution of Cooperation*. Basic Books Inc. New York.
- [Baader et al., 2010] Baader, F., Calvanese, D., McGuinness, D., Nardi, D., and Patel-Schneider, P. (2010). *The Description Logic Handbook*. Cambridge University Press, 2nd edition.

- [Bar-Shalom et al., 2001] Bar-Shalom, Y., Li, X. R., and Kirubarajan, T. (2001). *Estimation with Application to Tracking and Navigation: Theory, Algorithms, and Software*. Wiley.
- [Bay et al., 2006] Bay, H., Tuytelaars, T., and Gool, L. (2006). Surf: Speeded up robust features. In Leonardis, A., Bischof, H., and Pinz, A., editors, *Computer Vision - ECCV 2006*, volume 3951 of *Lecture Notes in Computer Science*, pages 404–417. Springer Berlin Heidelberg.
- [Beetz et al., 2011] Beetz, M., Klank, U., Maldonado, A., Pangercic, D., and Rühr, T. (2011). Robotic roommates making pancakes - look into perception-manipulation loop. In *11th IEEE-RAS International Conference on Intelligent Robots and Systems*.
- [Berberidis et al., 1968] Berberidis, C., Vlahavas, I., Aref, W., Atallah, M., and Elmagarmid, A. K. (1968). On the Discovery of Weak Periodicities in Large Time Series. In *Principles of Data Mining and Knowledge Discovery*, pages 51–61. Springer Berlin Heidelberg.
- [Blanc et al., 2009] Blanc, X., Mougénot, A., Mounier, I., and Mens, T. (2009). Incremental Detection of Model Inconsistencies Based on Model Operations. In *Advanced information systems engineering*, pages 32–46. Springer Berlin Heidelberg.
- [Blanc et al., 2008] Blanc, X., Mounier, I., Mougénot, A., and Mens, T. (2008). Detecting Model Inconsistency through Operation-Based Model Construction. In *ACM/IEEE 30th International Conference on Software Engineering*, pages 511–520.
- [Bobillo et al., 2008] Bobillo, F., Delgado, M., and Gomez-Romero, J. (2008). DeLorean: A Reasoner for Fuzzy OWL 1.1. In *URSW*.
- [Bond et al., 2010] Bond, S. D., Carlson, K. A., and Keeney, R. L. (2010). Improving the generation of decision objectives. In *Decision Analysis*, volume 7, pages 238–255.
- [Busch et al., 2012] Busch, F., Thomas, C., Deuse, J., and Kuhlenkötter, B. (2012). Development of a hybrid robot assistance system for welding of assemblies. In *Robotik 2012*.
- [Carpenter, 1989] Carpenter, G. A. (1989). Neural network models for pattern recognition and associative memory. *Neural Networks*, 2(4):243–257.

- [Chang and Lui, 2001] Chang, C. H. and Lui, S. C. (2001). IEPAD: Information Extraction Based on Pattern Discovery. In *Proceedings of 10th International Conference on World Wide Web*, pages 681–688.
- [Chua and Roska, 2002] Chua, L. O. and Roska, T. (2002). *Cellular neural Networks and visual computing*. Cambridge University Press.
- [Constantinti et al., 2006] Constantinti, G., Casali, D., and Perfetti, R. (2006). Associative memory design for 256 gray-level images using a multilayer neural network. *IEEE Trans. on Neural Networks*, 17(2).
- [Dasiopoulou et al., 2010] Dasiopoulou, S., Kompatsiaris, I., and Srinivasan, M. G. (2010). Investigating fuzzy DLs-based reasoning in semantic image analysis. In *Multimedia Tools and Applications*, volume 49(1), pages 167–194.
- [de Berg et al., 2008] de Berg, M., Cheong, O., van Kreveld, M., and Overmars, M. (2008). *Computational Geometry: Algorithms and Applications*. Springer.
- [Do Hoang et al., 2012] Do Hoang, Q. A., Guiochet, J., Powell, D., and Kaaniche, M. (2012). Human-robot interactions: Model-based risk analysis and safety case construction. In *6th European Congress on Embedded Real-Time Software and Systems*.
- [Doucet and Johansen, 2008] Doucet, A. and Johansen, A. (2008). A tutorial on particle filtering and smoothing: fifteen years later. Technical report, Department of Statistics, University of British Columbia.
- [Duden, 1990] Duden (1990). *Band 5 Fremdwörterbuch*. Dudenverlag.
- [Eagle et al., 2009] Eagle, N., Clauset, A., and Quinn, J. A. (2009). Location segmentation, inference and prediction for anticipatory computing. In *AAAI Spring Symposium: Technosocial Predictive Analytics*, pages 20–25.
- [Eaton, 1948] Eaton, J. W. (1948). A conceptual theory of co-operation. *American Journal of Sociology*, 54(2):126–134.
- [Elmezain et al., 2008] Elmezain, M., Al-Hamadi, A., Appenrodt, J., and Michaelis, B. (2008). A hidden markov modelbased continuous gesture recognition system for hand motion trajectory. In *Proceedings of the 19th IEEE International Conference on Pattern Recognition*.
- [Elmezain et al., 2009] Elmezain, M., Al-Hamadi, A., and Michaelis, B. (2009). Improving hand gesture recognition using 3d combined features. In *2nd International Conference on Machine Vision*, pages 128–132.

- [Endsley, 1995] Endsley, M. R. (1995). Toward a theory of situation awareness in dynamic systems. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 37(1):32–64.
- [Ester et al., 1996] Ester, M., Kriegel, H. P., Sander, J., and Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. *KDD*, 96:226–231.
- [Ferrein, 2010] Ferrein, A. (2010). Robot controllers for highly dynamic environments with real-time constraints. In *Künstliche Intelligenz*, volume 24(2), pages 175–178.
- [Ferrein et al., 2005] Ferrein, A., Fritz, C., and Lakemeyer, G. (2005). Using Golog for Deliberation and Team Coordination in Robotic Soccer. In *Künstliche Intelligenz*, volume 1, pages 24–43.
- [Ferrein and Lakemeyer, 2008] Ferrein, A. and Lakemeyer, G. (2008). Logic-based robot control in highly dynamic domains. In *Robotics and Autonomous Systems*, volume 56(11), pages 980–991.
- [Gardiner et al., 2006] Gardiner, T., Horrocks, I., and Tsarkov, D. (2006). Automated Benchmarking of Description Logic Reasoners. In *Proceedings of the 2006 International Workshop on Description Logics*, page 8.
- [Ghallab et al., 2004] Ghallab, M., Nau, D., and Traverso, P. (2004). *Automated Planning: Theory and Practice*. Morgan Kaufmann Publishers.
- [Gibson, 1977] Gibson, J. J. (1977). The theory of affordances. *Hilldale, USA*.
- [Graf, 2010] Graf, J. (2010). *Sichere Mensch-Roboter-Kooperation durch Auswertung von Bildfolgen*. PhD thesis, der Fakultät für Informatik des Karlsruher Instituts für Technologie (KIT).
- [Graf et al., 2010] Graf, J., Czapiewski, P., and Wörn, H. (2010). Evaluating risk estimation methods and path planning for safe human-robot cooperation. In *Proc. of the Joint 41th Int. Symp. on Robotics and 6th German Conf. on Robotics*, pages 579–585.
- [Graf et al., 2009] Graf, J., Puls, S., and Wörn, H. (2009). Incorporating novel path planning method into cognitive vision system for safe human-robot interaction. In *Computation World: Cognitive 2009*, pages 443–447.
- [Grau et al., 2010] Grau, B. C., Halaschek-Wiener, C., Kazakov, Y., and Suntisrivaraporn, B. (2010). Incremental classification of description

- logics ontologies. In *Journal of Automated Reasoning*, volume 44(4), pages 337–369.
- [Haarslev et al., 2007] Haarslev, V., Möller, R., and Wessel, M. (2007). Racerpro users guide and reference manual. Version 1.9.1.
- [Haddadin et al., 2008] Haddadin, S., Albu-Schäffer, A., De Luca, A., and Hirzinger, G. (2008). Evaluation of collision detection and reaction for a human-friendly robot on biological tissues. In *IARP International Workshop on Technical challenges and for dependable robots in Human environments*.
- [Haddadin et al., 2010] Haddadin, S., Albu-Schäffer, A., and Hirzinger, G. (2010). Safety analysis for a human-friendly manipulator. In *International Journal of Social Robotics*, volume 2, pages 235–252.
- [Haddadin et al., 2012] Haddadin, S., Haddadin, S., Khoury, A., Rokahr, T., Parusel, S., Burgkart, R., Bicchi, A., and Albu-Schäffer, A. (2012). On making robots understand safety: Embedding injury knowledge into control. In *The International Journal of Robotics Research*, volume 31, pages 1578–1602.
- [Halaschek-Wiener and Hendler, 2007] Halaschek-Wiener, C. and Hendler, J. (2007). Toward expressive syndication on the web. In *Proceedings of the 16th international conference on World Wide Web*, pages 727–736.
- [Heiligensetzer, 2003] Heiligensetzer, P. (2003). *Sichere Mensch-Roboter Kooperation durch Fusion haptischer und kapazitiver Sensorik*. PhD thesis, Universität Karlsruhe.
- [Hähnel, 1998] Hähnel, D. (1998). *Golex: Ein Laufzeitsystem für die Aktionsbeschreibungssprache GOLOG zur Steuerung des mobile Roboters RHINO*. PhD thesis, Institute for Informatics III, Rheinische Friedrich-Wilhelms-Universität.
- [Hinkelman and Allen, 1989] Hinkelman, E. A. and Allen, J. F. (1989). Two Constraints on Speech Act Ambiguity. In *Proceedings of the 27th annual meeting on Association for Computational Linguistics*, pages 212–219.
- [Hopfield, 1982] Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. In *Proceedings of the National Academy of Sciences of the United States of America*, volume 79, pages 2554–2558.

- [Horridge and Bechhofer, 2009] Horridge, M. and Bechhofer, S. (2009). The OWL API: A Java API for Working with OWL 2 Ontologies. In *Proceedings of OWL: Experiences and Directions*.
- [Hu, 1962] Hu, M.-K. (1962). Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, 8(2):179–187.
- [Hummel et al., 2007] Hummel, B., Thiemann, W., and Lulcheva, I. (2007). Description logics for vision-based intersection understanding. In *Proceedings of Cognitive Systems with Interactive Sensors*.
- [Jafar et al., 2014] Jafar, F. A., Abdullah, N., Muhammad, M. N., Zakaria, N. A., and Ali Mokhtar, M. N. (2014). Investigation of human emotional state in human-robot collaboration. In *Journal of Computers*, volume 9(3), pages 668–677.
- [Junxia et al., 2008] Junxia, G., Xiaoqing, D., Shengjin, W., and Wu Youshou, W. (2008). Full body tracking-based human action recognition. In *Proceedings of the 19th International Conference on Pattern Recognition*.
- [Kalal et al., 2010] Kalal, Z., Mikolajczyk, K., and Matas, J. (2010). Forward-Backward Error: Automatic Detection of Tracking Failures. In *Proceedings of International Conference on Pattern Recognition*, pages 2756–2759.
- [Kamarainen et al., 2007] Kamarainen, J.-K., Kyrki, V., and Kälviäinen, H. (2007). Local and global gabor features for object recognition. *Pattern Recognition and Image Analysis*, 17(1):93–105.
- [Karg et al., 2013] Karg, M., Seiber, W., Hoey, J., and Kulic, D. (2013). Human movement analysis: Extension of the f-statistic to time series using hmm. In *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on*, pages 3870–3875. IEEE.
- [Karg, 2012] Karg, M. E. (2012). *Pattern Recognition Algorithms for Gait Analysis with Application to Affective Computing*. PhD thesis, Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München.
- [Kendon, 2004] Kendon, A. (2004). *Gesture: Visible Action as Utterance*. Cambridge University Press.
- [Khatib, 1986] Khatib, O. (1986). Real-time obstacle avoidance for manipulators and mobile robots. *The International Journal of Robotics Research*, 5:90–98.

- [Krajcovic et al., 2011] Krajcovic, J., Hrncar, M., and Muzikarova, L. (2011). Hidden Markov Models in Speech Recognition. In *Advances in Electrical and Electronic Engineering*, volume 7(1-2).
- [Krüger et al., 2009] Krüger, N., Piater, J., Wörgötter, F., Geib, C., Petrick, R., Steedman, M., Ude, A., Asfour, T., Kraft, D., Omrcen, D., Hommel, B., Agostino, A., Kragic, D., Eklundh, J., Kruger, V., and Dillmann, R. (2009). A Formal Definition of Object Action Complexes and Examples at different Levels of the Process Hierarchy. Technical report, PACO-PLUS.
- [Krüger et al., 2007] Krüger, V., Kragic, D., Ude, A., and Geib, C. (2007). The meaning of action: A review on action recognition and mapping. In *Proceedings of Advanced Robotics*, volume 21, pages 1473–1501.
- [Kulić and Croft, 2006] Kulić, D. and Croft, E. A. (2006). Real-time safety for human–robot interaction. *Robotics and Autonomous Systems*, 54(1):1–12.
- [Kulić and Croft, 2007] Kulić, D. and Croft, E. A. (2007). Pre-collision safety strategies for human-robot interaction. *Autonomous Robots*, 22(2):149–164.
- [Kulić et al., 2012] Kulić, D., Ott, C., Lee, D., Ishikawa, J., and Nakamura, Y. (2012). Incremental learning of full body motion primitives and their sequencing through human motion observation. *The International Journal of Robotics Research*, 31(3):330–345.
- [Kulic et al., 2007] Kulic, D., Takano, W., and Nakamura, Y. (2007). Representability of human motions by factorial hidden markov models. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 2388–2393. IEEE.
- [Kunz et al., 2010] Kunz, T., Reiser, U., Stilman, M., and Verl, A. (2010). Real-time path planning for a robot arm in changing environments. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.
- [Lacevic and Rocco, 2010] Lacevic, B. and Rocco, P. (2010). Kinetostatic danger field - a novel safety assessment for human-robot interaction. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 2169–2174.
- [LaValle, 2006] LaValle, S. M. (2006). *Planning Algorithms*. Cambridge Univ. Press.

- [Lenz, 2011] Lenz, C. (2011). *Context-aware human-robot collaboration as a basis for future cognitive factories*. PhD thesis, Technische Universität München.
- [Leonard, 2009] Leonard, N. E. (2009). Integrating human and robot decision-making dynamics. In *2nd International Conference on Robot Communication and Coordination*, pages 1–2.
- [Leven and Hutchinson, 2002] Leven, P. and Hutchinson, S. (2002). A framework for real-time path planning in changing environments. *The International Journal of Robotics Research*, 21:999–1030.
- [Liu et al., 2002] Liu, W. Q., Easterbrook, S., and Mylopoulos, J. (2002). Rule-Based Detection of Inconsistency in UML Models. In *Workshop on Consistency Problems in UML-Based Software Development*, volume 5.
- [Loccoz et al., 2003] Loccoz, N. M., Bremond, F., and Thonnat, M. (2003). Recurrent bayesian network for the recognition of human behaviours from video. In *Proceedings of the 3rd International Conference on Computer Vision Systems (ICVS03)*.
- [Lowe, 1999] Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 1150–1157.
- [Lukasiewicz and Straccia, 2006] Lukasiewicz, T. and Straccia, U. (2006). An overview of uncertainty and vagueness in description logics for the semantic web. Technical report, Institut für Informationssysteme Technische Universität Wien.
- [Lukasiewicz and Straccia, 2008] Lukasiewicz, T. and Straccia, U. (2008). Managing uncertainty and vagueness in description logics for the semantic web. In *Web Semantics: Science, Services and Agents on the World Wide Web*, volume 6(4), pages 291–308.
- [Marmasse and Schmandt, 2002] Marmasse, N. and Schmandt, C. (2002). A usercentered location model. In *Personal and Ubiquitous Computing*, volume 6, pages 318–321.
- [Matthias et al., 2011] Matthias, B., Kock, S., Jerregard, H., Kallman, M., Lundberg, I., and Mellander, R. (2011). Safety of collaborative industrial robots: Certification possibilities for a collaborative assembly robot concept. In *IEEE International Symposium on Assembly and Manufacturing*, pages 1–6.

- [McCarthy, 1963] McCarthy, J. (1963). Situations, Actions and Causal Laws. Technical report, Stanford University.
- [Minnen et al., 2003] Minnen, D., Essa, I., and Starner, T. (2003). Expectation grammars: Leveraging high-level expectations for activity recognition. In *Proceedings of Computer Vision and Pattern Recognition*, number 2, pages 626–632.
- [Möller and Neumann, 2008] Möller, R. and Neumann, B. (2008). Ontology-based reasoning techniques for multimedia interpretation and retrieval. In *Semantic Multimedia and Ontologies*, pages 55–98. Springer.
- [Montibeller and Belton, 2006] Montibeller, G. and Belton, V. (2006). Causal maps and the evaluation of decision options - a review. In *Journal of the Operational Research Society*, volume 57, pages 779–791.
- [Morrison, 1968] Morrison, D. R. (1968). PATRICIA - Practical Algorithm to Retrieve Information Coded in Alphanumeric. In *Journal of the ACM*, volume 15(4), pages 514–534.
- [Naik, 2008] Naik, R. (2008). Blending the Logic Paradigm into C++. Online: <http://www.mpprogramming.com/resources/CastorDesign.pdf>.
- [Najmaei et al., 2010] Najmaei, N., Lele, S., Kermani, M. R., and Sobot, R. (2010). Human factors for robot safety assessment. In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 539–544.
- [Nasrabadi and Li, 1991] Nasrabadi, N. M. and Li, W. (1991). Object recognition by a hopfield neural network. In *IEEE Trans. on Systems, Man and Cybernetics*, volume 21.
- [Neumann and Möller, 2008] Neumann, B. and Möller, R. (2008). On scene interpretation with description logics. In *Image and Vision Computing*, volume 26, pages 81–101.
- [Niu and Abdel-Mottaleb, 2005] Niu, F. and Abdel-Mottaleb, M. (2005). HMM-Based Segmentation and Recognition of Human Activities from Video Sequences. In *IEEE International Conference on Multimedia and Expo*, pages 804–807.
- [Oberer-Treitz et al., 2010] Oberer-Treitz, S., Puzik, A., and Verl, A. (2010). Measuring the collision potential of industrial robots. In *41st International Symposium on Robotics (ISR) and 6th German Conference on Robotics (ROBOTIK)*, pages 1–7.

- [Padmanabhan and Tuzhilin, 1996] Padmanabhan, B. and Tuzhilin, A. (1996). Pattern Discovery in Temporal Databases: A Temporal Logic Approach. In *Proceedings of the International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 351–354.
- [Pan and Manocha, 2011] Pan, J. and Manocha, D. (2011). Gpu-based parallel collision detection for fast motion planning. *The International Journal of Robotics Research*.
- [Pan, 2005] Pan, Z. (2005). Benchmarking DL Reasoners Using Realistic Ontologies. In *Proceedings of the First OWL Experiences and Directions Workshop*.
- [Park and Aggarwal, 2004] Park, S. and Aggarwal, J. K. (2004). A hierarchical bayesian network for event recognition of human actions and interactions. In *Multimedia Systems*, volume 10, pages 164–179.
- [Parsia et al., 2006] Parsia, B., Halaschek-Wiener, C., and Sirin, E. (2006). Towards incremental reasoning through updates in owl-dl. In *Reasoning on the Web*.
- [Patel et al., 2012] Patel, R., Hedelind, M., and Lozan-Villegas, P. (2012). Enabling robots in small-part assembly lines: The rosetta approach - an industrial perspective. In *Proceedings of ROBOTIK and 7th German Conference on Robotics*, pages 1–5.
- [Pilz GmbH & Co. KG, 2013] Pilz GmbH & Co. KG (2013). Safety eye. Online: <http://www.safetyeye.de/>.
- [PMD Technologies, 2009] PMD Technologies (2009). Camcube 2.0 datasheet.
- [Poppe, 2010] Poppe, R. (2010). A survey on vision-based human action recognition. In *Image and Vision Computing*, volume 28, pages 979–990.
- [Proske, 2006] Proske, D. (2006). *Unbestimmte Welt*. Dirk Proske Verlag.
- [Puls et al., 2012a] Puls, S., Betz, P., Wyden, M., and Wörn, H. (2012a). Path planning for industrial robots in human-robot interaction. In *IEEE/RSJ International Conference in Intelligent Robots and Systems 2012 (IROS), WS Robot Motion Planning: Online, Reactive and in Real-time*.
- [Puls et al., 2013a] Puls, S., Giffhorn, C., and Wörn, H. (2013a). Identifying Objectives and Execution Planning for Safe Human-Robot Cooperation. In *XVI Portuguese Conference on Artificial Intelligence (EPIA)*, pages 252–263.

- [Puls et al., 2012b] Puls, S., Graf, J., and Wörn, H. (2012b). Design and Evaluation of Description Logics based Recognition and Understanding of Situations and Activities for Safe Human-Robot Cooperation. In *International Journal on Advances in Intelligent Systems*, volume 4, pages 218–227.
- [Puls et al., 2012c] Puls, S., Graf, J., and Wörn, H. (2012c). Cognitive robotics in industrial environments. In Inaki, M., editor, *Human Machine Interaction - Getting Closer*, pages 213–234. InTech.
- [Puls et al., 2013b] Puls, S., Lotspeich, B., and Wörn, H. (2013b). Incorporating Online Process Mining Based on Context Awareness into Human-Robot-Cooperation Framework. In *Proceedings of the 2nd International Conference on Intelligent Systems and Applications*, pages 89–94.
- [Puls et al., 2012d] Puls, S., Schnorr, N., and Wörn, H. (2012d). Object recognition based on depth information and associative memory. In *IEEE International Symposium on Robotic and Sensor Environments*, pages 186–191.
- [Puls and Wörn, 2012] Puls, S. and Wörn, H. (2012). Combining HMM-Based Continuous Human Action Recognition and Spatio-Temporal Reasoning for Augmented Situation Awareness. In *IADIS International Conference on Interfaces and Human Computer Interaction*, pages 133–140.
- [Puls and Wörn, 2013] Puls, S. and Wörn, H. (2013). Situation Dependent Risk Estimation for Workspace-Sharing Human-Robot Cooperation. In *Proceedings of IADIS International Conference on Intelligent Systems and Agents*, pages 51–58.
- [Puls and Wörn, 2014] Puls, S. and Wörn, H. (2014). Plausibility Verification for Situation Awareness in Safe Human-Robot Cooperation. In *The 23rd IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN) (to be published)*.
- [Puls and Wörn, 2013] Puls, S. and Wörn, H. (2013). Seamless Interfacing: Situation Awareness through Action Recognition and Spatio-Temporal Reasoning. In Blashki, K. and Isaias, P., editors, *Emerging Research and Trends in Interactivity and the Human-Computer Interface*, pages 144–159. IGI Global.

- [Rabiner, 1989] Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, volume 77, pages 257–286.
- [Reiser et al., 2009] Reiser, U., Connette, C., Fischer, J., Kubacki, J., Bubeck, A., Weishardt, F., Jacobs, T., Parlitz, C., Hägele, M., and Verl, A. (2009). Care-o-bot 3 - creating a product vision for service robot applications. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*.
- [Reiter, 2001] Reiter, R. (2001). *Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems*. MIT Press.
- [Santosh et al., 2009] Santosh, K., Lamiroy, B., and Ropers, J.-P. (2009). Inductive Logic Programming for Symbol Recognition. In *10th International Conference on Document Analysis and Recognition*, pages 1330–1334.
- [Shi et al., 2012] Shi, J., Jimmerson, G., Pearson, T., and Menassa, R. (2012). Levels of human and robot collaboration for automotive manufacturing. In *Proceedings of the Workshop on Performance Metrics for Intelligent Systems*, pages 95–100.
- [Shi et al., 2004] Shi, Y., Huang, Y., Minnen, D. and Bobick, A., and Essa, I. (2004). Propagation networks for recognition of partially ordered sequential action. In *Proceedings of Computer Vision and Pattern Recognition*, number 2, pages 862–869.
- [Shima et al., 2011] Shima, T., Ikuta, C., Uwate, Y., and Nishio, Y. (2011). Investigation of recall image by partitioned hopfield neural network. In *20th European Conference on Circuit Theory and Design*, pages 41–44.
- [Sirin, 2006] Sirin, E. (2006). *Combining Description Logics with AI Planning for Composition of Web Services*. PhD thesis, University of Maryland.
- [Sirin et al., 2007] Sirin, E., Parsia, B., Grau, B., Kalyanpur, A., and Katz, Y. (2007). Pellet: A practical owl-dl reasoner. In *Web Semantics: Science, Services and Agents on the World Wide Web*, volume 5, pages 51–53.
- [Springer et al., 2010] Springer, T., Wustmann, P., Braun, I., Dargie, W., and Berger, M. (2010). A comprehensive approach for situation-awareness based on sensing and reasoning about context. In *Lecture Notes in Computer Science*, volume 5061, pages 143–157. Springer.

- [Stoilos et al., 2008] Stoilos, G., Stamou, G., Pan, J. Z., Simou, N., and Tzouvaras, V. (2008). Reasoning with the fuzzy description logic *f-SHLN*: Theory, practice and applications. In *Uncertainty reasoning for the semantic web I*, pages 262–281.
- [Straeten et al., 2003] Straeten, R. V. D., Mens, T., Simmonds, J., and Jonckers, V. (2003). Using description logic to maintain consistency between UML models. In *UML 2003 - The Unified Modeling Language. Modeling Languages and Applications*, pages 326–340. Springer Berlin Heidelberg.
- [Sun and Giles, 2001] Sun, R. and Giles, C. L. (2001). Sequence Learning: From Recognition and Prediction to Sequential Decision Making. In *IEEE Intelligent Systems*, volume 16(4), pages 67–70.
- [Tadokoro et al., 1993] Tadokoro, S., Takebe, T., Ishikawa, Y., and Takamori, T. (1993). Control of human cooperative robots based on stochastic prediction of human motion. In *Proceedings of the 2nd IEEE International Workshop on Robot and Human Communication*, pages 387–392.
- [Tan et al., 2010a] Tan, J. T. C., Duan, F., Kato, R., and Arai, T. (2010a). Collaboration Planning by Task Analysis in Human-Robot Collaborative Manufacturing System. In *Advances in Robotic Manipulators*.
- [Tan et al., 2010b] Tan, J. T. C., Duan, F., Kato, R., and Arai, T. (2010b). Safety Strategy for Human-Robot Collaboration: Design and Development in Cellular Manufacturing. In *Advanced Robotics*, volume 24, pages 839–860.
- [Tan and Inamura, 2013] Tan, J. T. C. and Inamura, T. (2013). Integration of work sequence and embodied interaction for collaborative work based human-robot interaction. In *Proceedings of the 8th ACM IEEE international conference on Human-robot interaction*, pages 239–240.
- [Tenorth and Beetz, 2009] Tenorth, M. and Beetz, M. (2009). KNOWROB - Knowledge processing for autonomous personal robots. In *IEEE International Conference on Intelligent Robots and Systems (IROS)*.
- [Thiemermann, 2005] Thiemermann, S. (2005). *Direkte Mensch-Roboter-Kooperation in der Kleinteilmontage mit einem SCARA-Roboter*. PhD thesis, Fakultät für Maschinenbau der Universität Stuttgart.

- [Tsarkov and Horrocks, 2006] Tsarkov, D. and Horrocks, I. (2006). FaCT++ Description Logic Reasoner: System Description. In *Lecture Notes in Computer Science (LNCS)*, volume 4273, pages 654–667. Springer.
- [Tucnik, 2007] Tucnik, P. (2007). Multicriterial Decision-Making Control of Robot Soccer Team. In Lima, P., editor, *Robotic Soccer*, pages 421–446. InTech.
- [van der Aalst, 2011a] van der Aalst, W. M. P. (2011a). *Process Mining - Discovery, Conformance and Enhancement of Business Processes*. Springer.
- [van der Aalst, 2011b] van der Aalst, W. M. P. (2011b). Process Mining: Discovering and Improving Spaghetti and Lasagna Processes. In *IEEE Symposium on Computational Intelligence and Data Mining*.
- [van der Aalst et al., 2004] van der Aalst, W. M. P., Weijters, A., and Maruster, L. (2004). Workflow mining: discovering process models from event logs. In *IEEE Transactions on Knowledge and Data Engineering*, pages 1128–1142.
- [Venkatasubramanian et al., 2003] Venkatasubramanian, V., Rengaswamy, R., Yin, K., and Kavuri, S. N. (2003). A review of process fault detection and diagnosis Part I: Quantitative model-based methods. In *Computers and Chemical Engineering*, volume 27, pages 293–311.
- [Versmold and Saeger, 2006] Versmold, H. and Saeger, M. (2006). Plausibility Checking of Sensor Signals for Vehicle Dynamics Control Systems. In *8th International Symposium on Advanced Vehicle Control AVEC*, volume 136.
- [Vogel et al., 2012] Vogel, C., Walter, C., and Elkmann, N. (2012). Exploring the possibilities of supporting robot-assisted work places using a projection-based sensor system. In *IEEE International Symposium on Robotic and Sensors Environments (ROSE)*, pages 67–72.
- [Waimer, 2006] Waimer, M. (2006). Integration Adaptiver Prozess-Management-Technologie und Process Mining.
- [Weijters et al., 2006] Weijters, A., van der Aalst, W., and de Medeiros, A. (2006). Process mining with the Heuristics Miner algorithm. Technical report, Technische Universiteit Eindhoven.

- [Wen et al., 2009] Wen, L., Wang, J., van der Aalst, W., Huang, B., and Sun, J. (2009). A novel approach for process mining based on event types. In *Journal of Intelligent Information Systems*, pages 163–190.
- [Winkler, 2008] Winkler, B. (2008). Konzept zur sicheren mensch-roboterkooperation auf basis von schnellen 3-d time-of-flight sensoren. In *Robotik 2008*, pages 147–151.
- [Wittbrodt et al., 2009] Wittbrodt, N., Hillebrand, A., Drewitz, U., D., S.-R., and Thüring, M. (2009). Situation awareness: eine kognitionspsychologische erweiterung und präzisierung. In Liechtenstein, A., Stößel, C., and Clemens, C., editors, *Prospektive Gestaltung von Mensch-Technik-Interaktion*, pages 351–354. VDI Verlag.
- [Wu et al., 2008] Wu, Y. C., Chen, H. S., Tsai, W. J., Lee, S. Y., and Yu, J. Y. (2008). Human action recognition based on layered-hmm. In *Proceedings of the IEEE International Conference on Multimedia and Expo*.
- [Yamada et al., 2007] Yamada, N., Sakamoto, K., Kunito, G., Isoda, Y., Yamazaki, K., and Tanaka, S. (2007). Applying ontology and probabilistic model to human activity recognition from surrounding things. In *IPSJ Digital Courier*, volume 3.
- [Yoshida and Kanehiro, 2011] Yoshida, E. and Kanehiro, F. (2011). Reactive robot motion using path replanning and deformation. In *IEEE International Conference on Robotics and Automation (ICRA)*.
- [Young et al., 1997] Young, S., Scott, P., and Nasrabadi, N. (1997). Object recognition using multilayer hopfiled neural network. In *IEEE Transactions on Image Processing*, volume 6.

Industrieroboter werden heutzutage sehr verbreitet in der Produktion eingesetzt. Jedoch arbeiten diese Roboter meist isoliert und unter stark kontrollierten Bedingungen, damit eine Gefahr für den Menschen ausgeschlossen werden kann. Damit jedoch die kognitiven Fähigkeiten und Fingerfertigkeit von Menschen mit der Ausdauer, Stärke und Genauigkeit von Robotern in einer sicheren Kooperation zusammengeführt werden können, muss das Robotersystem seine Umgebung wahrnehmen und verstehen können. In dem vorgestellten System wird die Roboterumgebung mittels Algorithmen des maschinellen Lernens erfasst und Objekte sowie menschliche Handlungen bestimmt. Anhand semantischer Analyse kann auf vorliegende Situationen geschlossen werden, wodurch sich dynamisch Risikobewertungen und Handlungsvorgaben für den Roboter ableiten lassen. Diese stellen die Grundlage für ein reaktives Roboterverhalten dar, das eine zielgerichtete und sichere Mensch-Roboter-Kooperation ermöglicht.

ISBN 978-3-7315-0299-9



9 783731 502999 >