

Felix Lütteke

*Vielseitiges autonomes Transportsystem basierend auf Weltmodellerstellung mittels Datenfusion von Deckenkameras und Fahrzeugsensoren*





Felix Lütteke

*Vielseitiges autonomes Transportsystem basierend auf Weltmodellerstellung mittels Datenfusion von Deckenkameras und Fahrzeugsensoren*

Bericht aus dem Lehrstuhl für  
Fertigungsautomatisierung und Produktionssystematik  
Prof. Dr.-Ing. Jörg Franke

**FAPS**



Meisenbach  
G m b H V e r l a g

Als Dissertation genehmigt von der Technischen Fakultät  
der Friedrich-Alexander-Universität Erlangen-Nürnberg

Tag der mündlichen Prüfung: 29. September 2014

Vorsitzende des Promotionsorgans: Prof. Dr.-Ing. habil. Marion Merklein  
Gutachter: Prof. Dr.-Ing. Jörg Franke  
Prof. Dr.-Ing. Thomas Moor

### **Bibliografische Information Der Deutschen Bibliothek**

Die Deutsche Bibliothek verzeichnet diese Publikation in der  
Deutschen Nationalbibliografie; detaillierte bibliografische Daten  
sind im Internet über <http://dnb.ddb.de> abrufbar.

ISSN 1431-6226

ISBN 978-3-87525-376-4

Dieses Werk ist urheberrechtlich geschützt.

Alle Rechte, auch die der Übersetzung, des Nachdrucks  
und der Vervielfältigung des Buches oder Teilen daraus,  
vorbehalten.

Kein Teil des Werkes darf ohne schriftliche Genehmigung des  
Verlages in irgendeiner Form (Fotokopie, Mikrofilm oder ein  
anderes Verfahren), auch nicht für Zwecke der Unterrichts-  
gestaltung - mit Ausnahme der in den §§ 53, 54 URG ausdrücklich  
genannten Sonderfällen -, reproduziert oder unter Verwendung  
elektronischer Systeme verarbeitet, vervielfältigt oder  
verbreitet werden.

© Meisenbach Verlag Bamberg 2014

Herstellung: inprint GmbH, Erlangen

Printed in Germany

# Vielseitiges autonomes Transportsystem basierend auf Weltmo- dellerstellung mittels Datenfusion von Deckenkameras und Fahrzeugsensoren

Der Technischen Fakultät  
der Friedrich-Alexander-Universität  
Erlangen-Nürnberg

zur  
Erlangung des Doktorgrades Dr.-Ing.

vorgelegt von  
Felix Lütteke  
aus Darmstadt



## Vorwort und Danksagung

Die vorliegende Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik (FAPS) der Friedrich-Alexander-Universität Erlangen-Nürnberg.

Mein herzlicher Dank gilt dem ehemaligen Lehrstuhlinhaber Prof. Dr.-Ing. Klaus Feldmann, der mir die Möglichkeit zur Promotion gegeben hat. Zusätzlich hat er mir das Themenfeld der Fahrerlosen Transportfahrzeuge ans Herz gelegt und somit den Grundstein für meine erfüllte Zeit am Lehrstuhl FAPS mit technisch hoch interessanten Aufgabenstellungen gelegt. Seinem Nachfolger, Prof. Dr.-Ing. Jörg Franke, möchte ich ebenfalls für die langjährige Unterstützung bei der Bearbeitung meines Promotionsthemas danken. Zum einen für die fachlichen Diskussionen und Anregungen, sowie den Freiraum, um eigene Ideen selbstständig verfolgen zu können. Zum anderen auch für die Förderung eines besonders harmonischen und kollegialen Arbeitsumfelds am Lehrstuhl, welches die Arbeit dort sehr abwechslungsreich, diskussionsintensiv und angenehm gemacht hat.

Im Rahmen meiner Lehrstuhltätigkeit und den damit verbundenen Aufgaben in der Lehre bestand auch intensiver Kontakt zu Wissenschaftlern anderer Fachrichtungen. Hier ist insbesondere Prof. Dr.-Ing. Thomas Moor des Lehrstuhls für Regelungstechnik der Friedrich-Alexander-Universität Erlangen-Nürnberg zu nennen, der mir viele interessante Einblicke in aktuelle Forschungsgebiete der Regelungstechnik ermöglicht hat. Bei ihm möchte ich mich für die Übernahme der Rolle des Zweitgutachters bedanken. Ebenso hat mir Prof. Dr.-Ing. Ingo Hahn des Lehrstuhls für Elektrische Antriebe und Maschinen der Friedrich-Alexander-Universität Erlangen-Nürnberg immer selbstlos weitergeholfen, wenn Fragen bezüglich elektrischer Antriebslösungen zu diskutieren waren. Hierfür und für die Übernahme des Amts als fachfremder Prüfer bei meiner Promotion danke ich ihm herzlich.

Die bereits erwähnte positive Atmosphäre am Lehrstuhl und bei der täglichen Arbeit wäre ohne meine Kollegen nicht möglich gewesen. Daher bedanke ich mich bei all jenen, die mich bei der Arbeit durch die Diskussion technischer und organisatorischer Themen sowie die Programmierung einzelner Softwaremodule, aber auch durch Motivation und die Organisation von Gemeinschaftsveranstaltungen unterstützt haben. Hervorheben möchte ich an dieser Stelle meinen langjährigen Bürokollegen Dr.-Ing. Christian Ziegler, meinen ehemaligen Mitbewohner Dr.-Ing. Markus Michl und meinen unermüdlich mich unterstützenden Kollegen Xu Zhang.

Schließlich bedanke ich mich bei meiner Frau Sabine für Ihre Unterstützung bei meiner Arbeit und meinem kleinen Sohn Max für sein positives Wesen. Ebenso danke ich ganz herzlich meinen Eltern für ihre immerwährende Unterstützung auf meinem Lebensweg, der mich bisher immerhin zum Dokortitel geführt hat.

Fürth, im Oktober 2014

Felix Lütteke



**Inhaltsverzeichnis**

**Abkürzungsverzeichnis .....v**

**Begriffsverzeichnis .....vii**

**1 Einleitung..... 1**

**2 Stand der Technik Fahrerloser Transportfahrzeuge .....5**

2.1 Fertigungsorganisation und deren Optimierungspotentiale .....6

2.2 Sicherheitsanforderungen..... 7

2.3 Einrichtungen zur Standortbestimmung und Lageerfassung .....8

2.4 Sicherheitseinrichtungen ..... 11

2.5 Mobile Systeme in Forschung und Wirtschaft..... 13

2.6 Fahrzeugkinematiken mobiler Roboter ..... 19

**3 Moderne Intralogistik erfordert einen effizienten Transport von Gütern  
geringer Größe .....23**

3.1 Fräsbearbeitung Drosselvorrichtungsgehäuse .....23

3.1.1 Aufnahme des Istzustands .....23

3.1.2 Spezielle Anforderungen an das Transportsystem .....24

3.2 Leiterplattenfertigung für elektrische Antriebslösungen .....25

3.2.1 Aufnahme des Istzustands .....25

3.2.2 Spezielle Anforderungen an das Transportsystem .....26

3.3 Halbleiterherstellung .....26

3.3.1 Aufnahme des Istzustands .....26

3.3.2 Spezielle Anforderungen an das Transportsystem .....28

3.4 Hauspostabwicklung im Klinikum.....28

3.4.1 Aufnahme des Istzustands .....28

3.4.2 Spezielle Anforderungen an das Transportsystem .....29

**4 Stationäre Sensorik ermöglicht maximale Flexibilität des VTS .....31**

4.1 Das neuartige VTS stellt den VTF ein Weltmodell zur Verfügung .....33

4.2 Anforderungen an Soft- und Hardwarekomponenten werden formuliert.....35

**5 VTF sind kleinskalig und agieren autonom .....39**

5.1 Konstruktion.....39

5.1.1	Rechnergestützte Konstruktion.....	41
5.1.2	Aufbau eines Prototypen .....	42
5.1.3	Rampe zur neigungslosen Überwindung von Höhendifferenzen .....	43
5.2	Softwarekomponenten .....	44
5.2.1	Mikrocontroller zur Regelung der Antriebsmotoren.....	45
5.2.2	Mikrocontroller zur Kommunikation mit dem Steuerungsrechner .....	46
5.2.3	Pfadplaner .....	52
5.2.4	Tracker .....	57
5.2.5	Ermittlung der Fahrzeugposition im Weltmodell.....	59
5.2.6	Erhöhung der Navigationsgenauigkeit bei Andockvorgängen .....	65
<b>6</b>	<b>Ein zentrales Weltmodell befähigt VTF zum autonomen Betrieb.....</b>	<b>79</b>
6.1	Mathematische Modellierung von Weltobjekten.....	79
6.1.1	Zielobjekte .....	79
6.1.2	Hindernisobjekte .....	81
6.1.3	Fahrzeugobjekte .....	82
6.2	Optische Erkennung von Vordergrundobjekten und Zielmarkern.....	86
6.2.1	Hintergrundsubtraktion zur Detektion von Vordergrundobjekten .....	86
6.2.2	Mustererkennung zur Detektion von Zielmarkern .....	95
6.3	Zuweisung von Visionobjekten zu Fahrzeug- und Hindernisobjekten .....	96
6.4	Zustandsschätzung von Objekten bei totzeitbehaftetem Messsignal .....	99
6.4.1	Das Kalmanfilter .....	99
6.4.2	Kalmanfilter für Hindernisobjekte .....	101
6.4.3	Kalmanfilter für Fahrzeugobjekte .....	104
6.5	Graphische Benutzerschnittstelle.....	110
<b>7</b>	<b>Die prototypische Realisierung des VTS zeigt dessen volle Funktionsfähigkeit .....</b>	<b>113</b>
7.1	Detaillierte Konstruktion des VTF.....	113
7.2	Prototypische Realisierung des VTS.....	116
7.3	Im Rahmen der Entwicklung eingesetzte Software .....	118
7.3.1	Betriebssysteme .....	118
7.3.2	Python, Numpy, Cython, PyQt4 .....	118
7.3.3	OpenCV .....	120
7.3.4	Middleware .....	121



7.4	Erprobung des Transportsystems im Laborbetrieb .....	122
<b>8</b>	<b>Zusammenfassung und Ausblick .....</b>	<b>129</b>
<b>9</b>	<b>Summary.....</b>	<b>133</b>
<b>10</b>	<b>Literaturverzeichnis .....</b>	<b>137</b>



## Abkürzungsverzeichnis

AMR	Autonomous Mobile Robot
APT	Advanced Package Tools
AWT	Automatischer Warentransport
BSD	Berkeley Software Distribution
BZ	Bearbeitungszentrum
CAD	rechnergestützte Konstruktion (computer-aided design)
DARPA	Defense Advanced Research Projects Agency
DOF	Freiheitsgrad (degree of freedom)
DTP	Drehbare Transportplattform
ESD	Elektrostatische Entladung (electrostatic discharge)
FAPS	Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik
FTF	Fahrerloses Transportfahrzeug
FTS	Fahrerloses Transportsystem
GUI	Grafische Benutzerschnittstelle (graphical user interface)
ID	Identifikationsnummer
IFR	International Federation of Robotics
IFT	Institut für Fördertechnik und Logistik
IML	Fraunhofer-Institut für Materialfluss und Logistik
IPA	Fraunhofer-Institut für Produktionstechnik und Automatisierung
IR	Infrarot
IREL	Infrarot-emittierende Diode (infrared-emitting diode)
KARIS	Kleinskaliges Autonomes Redundantes Intralogistik System
KATE	Kleine Automatische Transporteinheit
KIT	Karlsruher Institut für Technologie
KLT	Kleinladungsträger
LiPo	Lithium-Polymer (-Akkumulator)
LTI	Linear, zeitinvariant (linear time-invariant)
MID	spritzgegossene Schaltungsträger (molded interconnect devices)
MSB	Höchstwertiges Bit (most significant bit)

MSM	Multishuttle Move
OMPL	Open Motion Planning Library
PL	Pufferlager
PR2	Personal Robot 2
PSD	Optischer Positionssensor (position-sensitive detector)
PWM	Pulsweitenmodulation
PXE	Preboot Execution Environment
RFID	radio-frequency identification
RGB	Rot, Grün, Blau
ROS	Robot Operating System
RRT	Rapidly Exploring Random Tree
SIFT	skaleninvariante Merkmalstransformation (scale-invariant feature transform)
SMC	Sequenzielle Monte-Carlo-(Methode)
SMD	oberflächenmontiertes Bauelement (surface-mounted device)
THT	Durchsteckmontage (through-hole technology)
TOF	Lichtlaufzeit (time of flight)
UML	vereinheitlichte Modellierungssprache (unified modeling language)
VTF	Vielseitiges Transportfahrzeug
VTs	Vielseitiges Transportsystem; Versatile Transportation System
VTv	Versatile Transportation Vehicle
WST	Werkstückträger
YUV	Y (Luminanz (Lichtstärke pro Fläche)), Chrominanz (Farbanteil U und V)

## Begriffsverzeichnis

Deckenkamera	An der Decke über dem Operationsgebiet der Fahrzeuge angebrachte Kamera. Die Kamera ist senkrecht Richtung Boden ausgerichtet.
DTP	Um die Hochachse drehbare Transportplattform des VTF.
Fahrzeugobjekt	Im Weltmodell hinterlegte Modellierung eines VTF.
Hindernisobjekt	Im Weltmodell hinterlegte Modellierung eines Hindernisses.
Markermittellinie	Linie auf optischem Marker senkrecht zur Orientierungslinie. Verläuft mittig durch das Schachbrettmuster, die Orientierungslinie und das ID-Feld.
Orientierungslinie	Linie auf optischem Marker, damit Transportfahrzeuge sich relativ zum Marker besser orientieren können.
Pose	Räumliche Lage eines Objektes bestehend aus Position und Orientierung.
RGB	Modell zur Darstellung einer Farbe durch additives Mischen der Farben Rot, Grün und Blau.
Schachbrettfeld	Quadratische, ausschließlich schwarze bzw. weiße Teilfläche eines Schachbretts.
Stationärer Rechner	Unbeweglicher Rechner, auf dem die von den Deckenkameras aufgenommenen Bilder verarbeitet, das Weltmodell errechnet und alle Daten per WLAN für alle VTF zur Verfügung gestellt werden.
Unterbodensensor	Optischer IR-Reflektionssensor zur Erfassung der Untergrundhelligkeit (schwarz/weiß).
Visionobjekt	Alle Vordergrundobjekte außer Zielmarkern sind Visionobjekte. Sie werden von der Bildverarbeitungsroutine im stationären Rechner detektiert.
Vordergrundobjekt	Durch Hintergrundsubtraktion in von Deckenkameras aufgenommenen Daten erkannte Objekte, die nicht zum Hintergrund (im Allg. Boden) gehören. Werden in der Bildverarbeitungsroutine des stationären Rechners als Zwischenwerte ermittelt.
VTF	Velseitiges Transportfahrzeug des in dieser Arbeit vorgestellten VTS. Es ist in der Lage, auf Basis des Weltmodells autonom Transportaufgaben zu erledigen.
VTS	Velseitiges Transportsystem als Gesamtheit aller Komponenten wie Deckenkameras, stationärem Rechner inklusive Weltmodell

	und allen VTF.
Weltkoordinatensystem	Kartesisches Rechts-Koordinatensystem in dem die Position von Weltobjekten dargestellt wird. Die Orientierung $\phi$ wird mathematisch positiv relativ zur positiven x-Achse notiert.
Weltmodell	Gesamtheit aller Weltobjekte. Repräsentiert den aktuellen Zustand der Betriebsumgebung der VTF sowie den Zustand aller VTF.
Weltobjekt	Abstrakte Modellierung eines real existenten Objekts, das im Weltmodell abgespeichert ist. Es gibt im Weltmodell 3 verschiedene Typen von Weltobjekten: Fahrzeugobjekte, Hindernisobjekte und Zielobjekte.
YUV	Modell zur Darstellung einer Farbe mit den Komponenten Luminanz Y (Lichtstärke pro Fläche) und Chrominanz (Farbanteil U und V).
Zielmarker	Optischer Marker, der hauptsächlich durch ein 4*4 Felder großes Schachbrettmuster und eine in 1*4 Feldern binär codierte Identifikationsnummer (ID) charakterisiert ist. Ein Zielmarker kann durch die Deckenkameras in Position und Orientierung sowie ID erkannt werden.
Zielobjekt	Im Weltmodell hinterlegte Modellierung eine Zielposition.
Zielposition	Durch optischen Zielmarker definierte Position mit eindeutiger ID. Wird im Weltmodell als Objekt modelliert.

# 1 Einleitung

Die Entwicklung Fahrerloser Transportfahrzeuge (FTF) als innerbetriebliche Logistiklösung erstreckt sich über die vergangenen 60 Jahre. Dabei werden bei der Entwicklung von Fahrerlosen Transportsystemen (FTS) als Kombination mehrerer FTF, zentraler Leitsteuerung und infrastruktureller Installationen insbesondere folgende Ziele hinsichtlich innerbetrieblichem Materialfluss verfolgt:

- kurze Durchlaufzeiten durch hohe Geschwindigkeiten, optimales Routing und kurze Wartezeiten
- geringe Lagerbestände durch Unterstützung des One-piece-flow
- hohe Flexibilität durch beliebige Vernetzung von Arbeitsstationen

Mit diesen Eigenschaften tragen moderne FTS heute weltweit dazu bei, logistische Aufgaben zu erledigen. Eingesetzt in Krankenhäusern zum Transport von Wäsche, Essen und Medikamenten, in Hafenanlagen zum Transport von Containern, in der fertigenden Industrie zum Transport von Automobilteilen und Flugzeugteilen sowie in der Lebensmittelindustrie, stellen FTS einen zentralen Baustein in der Logistikkette unserer täglichen Versorgung mit Gütern dar.

Einteilen lässt sich die Entwicklungsgeschichte dieser wichtigen Hilfsmittel für unsere tägliche Versorgung in folgende vier Epochen [111]:

I) Idee und Umsetzung (ab 1953 bis Ende 1960er Jahre): In der Anfangszeit der FTS basieren die Systeme auf einfachsten Spurfolgetechniken mit Hilfe in den Boden eingelassener oder auf den Boden aufgebrachter Leitlinien und taktilen Sensoren zur Kollisionsvermeidung. Die Systeme sind unflexibel und prinzipiell nur für Vorwärtsfahrt und zur Überbrückung größerer Strecken geeignet.

II) Automatisierungseuphorie (1970er und 1980er Jahre): Mittels aktiver, induktiver Spurführung über einen Draht im Boden per Infrarot (IR) oder Funk erfolgt eine Datenübertragung an die Fahrzeuge. Somit können Abzweige- oder Haltevorgänge an einer weiterhin starr verlegten Leitlinie flexibel gestaltet und große Anlagen mit vielen FTF realisiert werden. Die Automatisierungseuphorie kommt jedoch Ende der 1980er Jahre zum Erliegen. Das liegt insbesondere daran, dass in Niedriglohnländern bei geringerem Automatisierungsgrad kostengünstiger und qualitativ hochwertiger als in der westlichen Welt produziert wird. Die in Japan perfektionierte schlanke Produktion (lean production) hat gegenüber der so genannten gepufferten Produktion (buffered production) entscheidende Vorteile hinsichtlich Flexibilität, Fehler- und Verschwendungsreduktion sowie kontinuierlicher Optimierung des Produktionsprozesses. Diese Vorteile werden nicht durch starre und unflexible Automatisierungslösungen erreicht. Vielmehr liegt in Japan das Augenmerk auf der Minimierung von verschiedenen Verschwendungsarten, von denen hier „Transport“ und „Lange Wege“ genannt seien. Dieser grundsätzliche Unterschied zwischen japanischer und westlicher Produktionsphilosophie wird sehr deutlich in einer 1991 veröffentlichten Studie des MIT [126] beschrieben. Als Konsequenz daraus erfolgt in der westlichen Welt eine Umorganisati-

on. Große FTS werden durch mehrere kleine Systeme, die hauptsächlich dem Pull-Prinzip gehorchen (z. B. mittels Kanban), ersetzt. Somit können Puffer zwischen den einzelnen Fertigungsschritten klein gehalten bzw. vollständig eliminiert und die Flexibilität der gesamten Intralogistik erhöht werden.

III) Gestandene Technik für die Intralogistik (1990er Jahre bis heute): In den letzten Jahren entwickeln sich insbesondere die Spurführungssysteme von FTS weiter. Während in Epoche II hauptsächlich induktive und optische Spurführungen zum Einsatz kommen, sind jetzt hochflexible virtuelle Leitlinien der Stand der Technik. Hier sind die Leitlinien ausschließlich in digitaler Form im Rechner hinterlegt. Das ermöglicht eine schnelle Adaption der Leitlinien an sich ändernde Umgebungsbedingungen. Um die Position eines FTF an jedem Ort der Betriebsumgebung exakt bestimmen zu können, sind jetzt neue Techniken nötig. Durchgesetzt haben sich insbesondere die Lasernavigation und die Magnetstrasternavigation. Beide sind in Kapitel 2.3 detailliert erläutert.

IV) Intelligente Systeme (nach Ullrich „Die nächste Epoche“ [111]): Zukünftig ist damit zu rechnen, dass preiswerte und intelligente Sensorsysteme auch bei FTS Einzug halten und damit einen zunehmend autonomen Betrieb ermöglichen. Während auf diesem Gebiet häufig die Automobilindustrie mit den verschiedensten technologischen Entwicklungen wie Spurhalteassistent, Parkassistent oder automatische Distanzregelung genannt werden, lohnt an dieser Stelle zusätzlich ein Blick auf eines der anspruchsvollsten Themen der Automation, die Servicerobotik. Denn während beim Automobil die letzte Verantwortung für die Aktionen des Fahrzeugs beim Fahrer liegt, müssen Serviceroboter in der für den Menschen geschaffenen Welt selbstständig Objekte erkennen, Entscheidungen treffen, Aktionen durchführen und für die Sicherheit aller beteiligten Personen und Objekte sorgen. Aus diesem Grund können aktuelle Entwicklungen in der Servicerobotik als Wegweiser für die Zukunft der FTS dienen.

Die vorliegende Arbeit hat zum Ziel, Fertigungsabläufe durch eine neuartige Transportlösung effizienter zu gestalten. Aktuell sind diese Abläufe im Allgemeinen verrichtungsorientiert oder flussorientiert organisiert. Zur Nutzung der jeweiligen Vorteile, die in 2.1 genannt sind, soll eine Kombination dieser Organisationsstrukturen ermöglicht werden. Dies erfordert jedoch eine Materialflusslösung, welche durch die aktuell verfügbaren Transportsysteme nicht realisiert werden kann. Die im Stand der Technik beschriebenen Lösungen sind je nach verwendeter Spurführungstechnologie entweder zu unflexibel oder extrem kostenintensiv. Um konkrete Anforderungen hinsichtlich Nutzlast, Transportgeschwindigkeit und Betriebszeiten zu ermitteln, wurden verschiedene Unternehmen, bei denen Güter mit Massen von wenigen kg zu transportieren sind, analysiert.

Der resultierende Lösungsansatz sieht den Einsatz kleinskaliger Fahrzeuge vor, die mit Hilfe bereitgestellter Umgebungsdaten autonom Transportaufträge abarbeiten können. Somit sind am Gesamtsystem keine bzw. minimale Veränderungen vorzunehmen, wenn die Anzahl an Fahrzeugen modifiziert werden soll, was die Skalierbarkeit hinsichtlich Transportleistung sicherstellt. Die Flexibilität hinsichtlich maximal transportierbarer Nutzlast wird erreicht, indem mehrere Fahrzeuge im Schwarmbetrieb gemeinsam eine Last bewegen. Zusätzlich ist vorgesehen, dass die den Fahrzeugen



bereitgestellten Umgebungsdaten neben den Fahrzeugpositionen auch die Positionen aller Hindernisse und Zielpositionen umfassen. Daraus resultiert eine maximale Flexibilität hinsichtlich Veränderungen im Betriebsumfeld. Um die hierzu benötigte Überwachung der Betriebsumgebung zu realisieren, ist die Verwendung herkömmlicher 2D Kameras vorgesehen.

Zur Umsetzung dieses Lösungsansatzes werden zunächst kleinskalige Fahrzeuge entwickelt. Hierzu erfolgt deren Konstruktion, die Auswahl geeigneter Hardwarekomponenten und die Entwicklung von Softwarekomponenten, wie in 5 beschrieben. Darüber hinaus erfordert die Bereitstellung der Umgebungsdaten eine effektive Bildverarbeitung und eine mathematische Repräsentation aller Weltobjekte. Insbesondere der Aspekt der intelligenten Sensorik, wie er in Epoche IV der FTS-Entwicklung genannt ist, wird mit dieser Arbeit adressiert. Die hierzu benötigten Softwarekomponenten sind in 6 erläutert.

Zur praktischen Umsetzung des Gesamtsystems werden aktuelle Entwicklungen aus dem Gebiet der (Service-) Robotik, wie ROS, die Bildverarbeitungssoftware OpenCV oder die Open Motion Planning Library (OMPL) untersucht und eingesetzt. Diese sind in 7 erläutert. Ebenso erfolgen in diesem Abschnitt eine Beschreibung der Erprobung des Prototypen und die Ableitung weiterer Forschungsthemen.



## 2 Stand der Technik Fahrerloser Transportfahrzeuge

FTS sind aus der modernen Industrie nicht mehr wegzudenken. Insbesondere in der Automobilindustrie aber auch in anderen Branchen wie der Papierindustrie, in automatisierten Containerterminals oder in Krankenhäusern wird die Intralogistik zunehmend durch FTS automatisiert. Um im Folgenden den Terminus FTS korrekt zu verwenden, soll hier zunächst die Definitionen nach VDI Richtlinie 2510 [116] zitiert werden.

### ***Fahrerlose Transportsysteme Definition***

„Fahrerlose Transportsysteme (FTS) sind innerbetriebliche, flurgebundene Fördersysteme mit automatisch gesteuerten Fahrzeugen, deren primäre Aufgabe der Materialtransport, nicht aber der Personentransport ist. Sie werden innerhalb und außerhalb von Gebäuden eingesetzt und bestehen [...] im Wesentlichen aus

- einem oder mehreren Fahrerlosen Transportfahrzeugen,
- einer Leitsteuerung,
- Einrichtungen zur Standortbestimmung und Lageerfassung,
- Einrichtungen zur Datenübertragung sowie
- Infrastruktur und peripheren Einrichtungen.“

Des Weiteren erfolgt die Unterteilung eines FTF in folgende Baugruppen:

- Bedienelemente
- Datenübertragung
- Warneinrichtungen
- Fahrzeugsteuerung
- Energieversorgung
- Lastaufnahmemittel
- Fahrzeugrahmen
- Sicherheitseinrichtungen

Um den Stand der Technik zu umreißen, wird im Folgenden zunächst Optimierungspotentiale in der Organisation fertiger Betriebe eingegangen. Anschließend kommen allgemeine Sicherheitsanforderungen an FTF zur Sprache. Dabei gibt Diskussion der zugrundeliegenden Normen Hinweise, welche Richtlinien und damit welche Technologien bei der Entwicklung des in dieser Arbeit beschriebenen VTS (Vielseitiges Transportsystem bzw. Versatile Transportation System) Verwendung finden können.

Im Rahmen der Entwicklung des VTS hat sich gezeigt, dass die Themen „Einrichtungen zur Standortbestimmung und Lageerfassung“ und „Sicherheitseinrichtungen“ besonders viel Entwicklungsaufwand erfordern. Daher wird im Folgenden insbesondere der Stand der Technik dieser beiden Themengebiete dargestellt.

Abschließend richtet sich der Blick auf einige moderne Systeme aus Forschung und Wirtschaft. Dabei liegt das Augenmerk nicht ausschließlich auf Fahrerlosen Transport-

fahrzeugen sondern allgemein auf mobilen Robotern. Hier werden die Techniken entwickelt und erprobt, die in den FTS der Zukunft zum Einsatz kommen werden.

Die nach VDI Richtlinie 2510 weiteren Baugruppen von FTF (Bedienelemente, Datenübertragung, Warneinrichtungen, Fahrzeugsteuerung, Energieversorgung, Lastaufnahmemittel und Fahrzeugrahmen) sind für die Entwicklung des in dieser Arbeit beschriebenen Vielseitigen Transportfahrzeugs (VTF) bzw. Versatile Transportation Vehicle (VTV) von untergeordneter Bedeutung, sodass deren Stand der Technik hier nicht gesondert betrachtet wird.

### 2.1 Fertigungsorganisation und deren Optimierungspotentiale

Die geeignete Wahl der Fertigungsorganisation eines produzierenden Unternehmens stellt die Grundlage für einen wirtschaftlichen Betrieb dar. Da die Umorganisation einer Fertigungsumgebung prinzipiell mit großem Arbeitsaufwand und mit Fertigungsausfällen verbunden ist, muss diese möglichst vermieden werden. Hierzu ist ein geeigneter Kompromiss zu finden, um sowohl flexibel als auch effizient zu fertigen. Prinzipiell lässt sich die Fertigungsorganisation nach dem Verrichtungs- und Objektprinzip unterscheiden. Während beim Verrichtungsprinzip der Produktionsbereich entsprechend der durchführbaren Fertigungsprozesse unterteilt ist, orientiert sich dessen Gestaltung beim Objektprinzip an den zu fertigenden Werkstücken. Die typischen Vertreter des Verrichtungsprinzips und des Objektprinzips, Werkstattfertigung und Fließfertigung, werden hier kurz erläutert, deren Vor- und Nachteile genannt und schließlich mögliche Optimierungen durch den Einsatz eines VTS diskutiert.

Die Werkstattfertigung wird für kundenindividuelle Auftragsfertigung eingesetzt. Sie erfordert maximale Flexibilität, da die Arbeitsgangfolge bei jedem Auftrag verschieden ist. Eine räumliche Anordnung der Arbeitsplätze entsprechend der durchzuführenden Arbeitsgänge ist daher nicht möglich. Stattdessen sind funktionsgleiche oder ähnliche Maschinen in Gruppen angeordnet.

Nachteile dieser Organisation liegen nach [15] in:

- der zeitlich-organisatorischen Unübersichtlichkeit
- den langen, uneinheitlichen Transportwegen
- den vielen Zwischenlagern mit der dadurch bedingten hohen Kapitalbindung
- den langen Verweildauern im Fertigungsbereich in Relation zur reinen Bearbeitungszeit
- dem ständig neuen Planungsbedarf bei jedem neuen Auftrag

Vorteile der Werkstattfertigung bestehen hingegen in

- einem großen Maß an Flexibilität und Anpassungsfähigkeit
- geringen Rüstzeiten und -kosten
- einem großen Entscheidungsspielraum für die meist qualitativ vielseitig ausgebildeten Arbeitskräfte
- einer schnellen und effizienten Reaktionsfähigkeit auf Störungen wie z. B. den Ausfall einer Maschine

- der Möglichkeit, durch den Einsatz von Universalmaschinen stark variierende Produktarten fertigen zu können

Im Gegensatz zur Werkstattfertigung sind bei der Fließfertigung die Maschinen entsprechend der Arbeitsgangfolge für den Produktentstehungsprozess organisiert. Durch die Vorgabe eines Zeittakts der einzelnen Arbeitsstationen entfällt der Bedarf für Zwischenlager. Bereits vor Produktionsbeginn wird die gesamte Fertigungsline konstruktiv auf eine optimale Austaktung ausgelegt, sodass jede Arbeitsstation möglichst exakt die vorgegebene Taktzeit einhält und gleichzeitig die Anzahl der benötigten Arbeitsstationen minimiert wird.

Die Nachteile der Fließfertigung liegen nach [15] in

- der geringen Flexibilität der Produktion, sodass hier nur gleichartige Güter mit großer Auflage wirtschaftlich gefertigt werden können
- der Abhängigkeit der einzelnen Stationen voneinander; Fehler in der zeitlichen Abstimmung im Rahmen der Materialbereitstellung führen bei dieser Organisationsform schnell zum Stillstand des gesamten Produktionsablaufs
- dem hohen Kapitalbedarf für die Fertigungseinrichtungen

Vorteile liegen hingegen in

- den geringen Transportkosten
- den geringen Durchlaufzeiten
- den geringen Anforderungen an die Fertigungssteuerung,

Aus dieser Gegenüberstellung lässt sich ableiten, dass die Unterschiede beider Fertigungsorganisationstypen insbesondere durch Materialflusslösungen reduziert werden können. Sollte ein Transportsystem in der Lage sein, Arbeitsplätze in einer nach dem Verrichtungsprinzip orientierten Fertigungseinrichtung schnell, kostengünstig und individuell miteinander zu vernetzen, ließen sich die Vorteile beider Organisationsarten kombinieren.

Ein hierzu benötigtes vielseitiges Transportsystem wird im Rahmen dieser Arbeit erforscht.

## 2.2 Sicherheitsanforderungen

Die Europäische Maschinenrichtlinie [32] stellt hohe Ansprüche an die Sicherheit moderner FTF. Dies ist nicht zuletzt der Tendenz geschuldet, dass FTF zunehmend autonom Transportaufgaben im Innen- und Außenbereich übernehmen.

In Tabelle 1 werden einige relevante Normen und Richtlinien benannt, die für FTF gelten. Dabei ist anzumerken, dass die Normen bindend sind, während die VDI Richtlinien lediglich Empfehlungen darstellen.

Im Rahmen der Entwicklung des VTS wurde untersucht, inwieweit geltende Normen auf kleinskalige FTF mit einer Masse von ca. 20 kg anwendbar sind. Da die DIN EN 1525 [29] eine von der europäischen Maschinenrichtlinie abgeleitete Norm für den deutschen Markt darstellt, sollen hier beispielhaft folgende Inhalte wiedergegeben werden:

- Punkt 5.2 der DIN EN 1525 fordert ein mechanisches Bremssystem, das in den verschiedensten Situationen das Fahrzeug zum Stillstand bringen kann.
- Entsprechend Punkt 5.3 ist es notwendig, dass eine Steuereinrichtung für Notfälle oder für die Wartung mit dem FTF verbunden ist.
- Unter Punkt 5.9 sind verschiedene Schutzeinrichtungen genannt, die bei jedem FTF zu installieren sind. Dazu zählen eine gut sichtbare Warneinrichtung (Blinklicht) wenn es fährt sowie Blinker zur Richtungsanzeige. Not-Aus-Schalter müssen an beiden Seiten des FTF gut zugänglich sein.

Tabelle 1: Ausgewählte Normen und Richtlinien mit Relevanz für FTS

Bezeichnung	Beschreibung
DIN EN 1525	Sicherheit von Flurförderzeugen, fahrerlose Flurförderzeuge und ihre Systeme
DIN EN ISO 3691-4	Flurförderzeuge – Sicherheitstechnische Anforderungen und Verifizierung –Teil 4: Fahrerlose Flurförderzeuge und ihre Systeme
DIN EN 954-1	Sicherheit von Maschinen, sicherheitsbezogene Teile von Steuerungen
DIN EN ISO 14121	Sicherheit von Maschinen, Leitsätze zur Risikobeurteilung (ehem. 1050)
DIN EN 1175-1	Sicherheit von Flurförderzeugen, elektrische Anforderungen
VDI 2510	Fahrerlose Transportsysteme (FTS)
VDI 2710	Ganzheitliche Planung von Fahrerlosen Transportsystemen (FTS): Grundlagen

Diese genannten Punkte sind Beispiele dafür, dass die geltenden Normen nicht ohne Weiteres für kleinskalige Fahrzeuge mit geringer Masse anwendbar sind. Um ein kleines, kostengünstiges Fahrzeug mit geringer Masse realisieren zu können, muss auf Not-Aus-Schalter oder ein mechanisches Bremssystem verzichtet werden, wobei die Sinnhaftigkeit solcher Installationen im beschriebenen Fall ohnehin fraglich ist.

Im Gegensatz zu den oben genannten pauschalen Forderungen nach Einrichtungen bei FTF wird unter Punkt 5.9.5.2 der betrachteten Norm definiert, welche Kräfte bei Kollisionen eines FTF mit Prüfkörpern auftreten dürfen. Hier sind erlaubte Betätigungskräfte von 750 N bzw. 250 N je nach verwendetem Prüfkörper festgelegt. Die Endkraft, die vom Bumper auf den Prüfkörper ausgeübt werden darf, ist mit maximal 400 N angegeben.


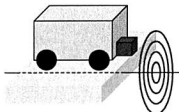
Diese Forderungen werden bei der Konstruktion des VTF (vgl. 5.1) berücksichtigt. Dabei wird das Fahrzeug so ausgelegt und betrieben, dass bei einer Kollision mit maximaler Geschwindigkeit eine Kraft von 400 N zu keinem Zeitpunkt überschritten wird.

### 2.3 Einrichtungen zur Standortbestimmung und Lageerfassung

Um seine Transportaufgabe selbstständig und sicher ausführen zu können, muss jedes FTF in der Lage sein, seinen Standort zu bestimmen. Hierfür existieren verschiedene Technologien, die im Laufe der FTF-Entwicklungsgeschichte bereits vielfach eingesetzt worden sind.

Klassische Spurführungstechnologien mittels physikalisch existenter Leitlinien waren insbesondere in den ersten beiden Epochen der FTS-Entwicklung maßgeblich. Je nach Funktionsprinzip existieren optische und induktive Verfahren mit verschiedenen Vor- und Nachteilen. Dabei ist den Verfahren gemein, dass die Leitlinie physisch existent ist. Sie muss also auf den Boden aufgebracht oder in den Boden eingelassen werden. In Tabelle 2 sind die zwei wesentlichen Kategorien physischer Leitlinien dargestellt, mit denen ein Rückschluss auf die Absolutlage des FTF möglich ist. Daher wird dieser Vorgang auch als Lagepeilung bezeichnet.

Tabelle 2: Einrichtungen zur Lagepeilung leitliniengeführter FTF angelehnt an [116]. Vor- und Nachteile angelehnt an [111].

	optische Leitlinie (bzw. passive Metallfolie)	induktive Leitlinie (aktiv)
Vorteile	<ul style="list-style-type: none"> <li>• preiswerte Technik</li> <li>• einfaches Layout ist schnell in Betrieb genommen</li> <li>• einfachste Systemsteuerung: stopp, wenn Leitspur unterbrochen oder bei zusätzlichen Bodenmarkierungen</li> </ul>	<ul style="list-style-type: none"> <li>• bewährte Technik</li> <li>• einfache Fahrzeugsteuerung</li> </ul>
Nachteile	<ul style="list-style-type: none"> <li>• keine Leitsteuerung</li> <li>• schwer erweiterbar</li> <li>• störanfällig bei Beschädigung des Farbstrichs oder Metallbands</li> </ul>	<ul style="list-style-type: none"> <li>• aufwändige Bodeninstallationen</li> <li>• Layoutänderungen extrem teuer</li> <li>• störanfällig bei Leitdrahtbruch</li> </ul>
Prinzipskizze		

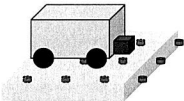
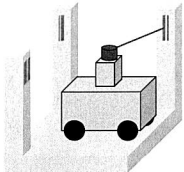
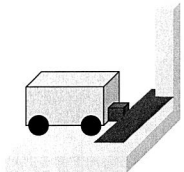
Diese Verfahren sind bewährt und werden vielfach auch heute noch eingesetzt. Während hinsichtlich Haltbarkeit, Installationsaufwand, möglicher Energieübertragung und Genauigkeit bei den dargestellten Systemen Unterschiede bestehen, haben sie einen gemeinsamen Nachteil. Alle Systeme, die auf einer physischen Leitlinie zur Spurführung basieren, sind hochgradig unflexibel bezüglich Fahrwegänderungen. Soll ein FTF einen neuen Weg nutzen, muss die Leitlinie neu verlegt werden.

Daher hat sich in der aktuellen dritten Epoche aus Flexibilitätsgründen die virtuelle Leitlinie etabliert. Sie ist in einem Rechner abgelegt und beschreibt die zu befahrenden Wege. Dieses Verfahren wird auch als freie Navigation bezeichnet, weil ein Fahrzeug nicht an eine physikalische Leitlinie gebunden ist. Hierfür muss ein FTF jedoch seine Position bestimmen können, ohne auf eine physikalisch existente Leitlinie zurück zu greifen. Zur indoor-Positionsbestimmung werden hauptsächlich die in Tabelle 3 dargestellten Einrichtungen verwendet.

Die verschiedenen Einrichtungen zur Lagepeilung von FTF mittels virtueller Leitlinie müssen eine Bestimmung der Pose (Position und Orientierung) im gesamten möglichen

Operationsgebiet erlauben. Das macht diese Systeme technisch aufwändiger und kostenintensiver als Systeme mit physikalischer Leitlinie.

Tabelle 3: Einrichtungen zur Lagepeilung virtuell geführter FTF, angelehnt an [116], Vor- und Nachteile, angelehnt an [111].

	Bodenmarkierungen (Raster; magnetisch, optisch)	bodenfreie Peilmarken (Reflektoren)	bodenfreie Peilmarken (natürliche Umgebungso- bjekte)
Vorteile	<ul style="list-style-type: none"> <li>flexibel innerhalb des Rasterbereichs</li> <li>Layout rein softwaremäßig anpassbar</li> </ul>	<ul style="list-style-type: none"> <li>hohe Genauigkeit der Lagepeilung möglich</li> </ul>	<ul style="list-style-type: none"> <li>Verwendung der ohnehin vorhandenen Personenschutzscanner zur Navigation</li> <li>keinerlei infrastrukturelle Installationen notwendig</li> </ul>
Nachteile	<ul style="list-style-type: none"> <li>Boden muss vorbereitet werden (Verlegung des Rasters)</li> <li>Einschränkungen bzgl. Bodenfreiheit und Bodenzustand</li> <li>Rasterverlegung bedeutet hohen Aufwand (insb. bei magnetischem Raster)</li> </ul>	<ul style="list-style-type: none"> <li>Reflektoren an den Wänden, Säulen, Maschinen erforderlich</li> <li>Laserkopf muss oberhalb der Last und den Mitarbeitern freie Rundumsicht haben</li> <li>wegen hohem Mast für den Laserkopf ist ebener Boden erforderlich</li> <li>Reflektoren können verschmutzen oder verdeckt werden</li> <li>Lichteinflüsse können System stören</li> </ul>	<ul style="list-style-type: none"> <li>erhöhter Softwareaufwand</li> <li>fehleranfällig bei Veränderungen der Umgebung</li> <li>nur geeignet für einfache Szenarien ohne viel Verkehr</li> </ul>
Prinzip- skizze			

Während die verschiedenen Einrichtungen zur Lagepeilung von FTF hier nicht im Detail erläutert werden sollen (hierzu sei auf [111] verwiesen), ist ein Kerngedanke festzuhalten: Virtuelle Leitlinien sind flexibel und im heutigen, wandelbaren Produktionsumfeld häufig von Vorteil. Die Bestimmung der Position ist jedoch kostenintensiv.

Zusätzlich zur Lagepeilung, die Absolutwerte für die Poseermittlung eines Fahrzeugs liefert, kann mittels Koppelnavigation die Genauigkeit der Posebestimmung erhöht werden. Dabei passiert das Koppeln mittels fahrzeugfester Sensoren wie Inkrementallencodern an den Antriebsmotoren oder Inertialsensoren (Beschleunigungs- und Drehratensensoren). Die Genauigkeit der Posebestimmung eines FTS lässt sich durch optimale Verrechnung von Lagepeilungs- und Koppelnavigationsdaten maximieren. Es existiert eine Vielzahl verschiedener Verfahren, um eine sogenannte Zustandsschät-



zung – in diesem Fall der Pose des FTF - durchzuführen. Ein populäres Beispiel hierfür stellt das Kalmanfilter (vgl. 6.4.1) dar.

## 2.4 Sicherheitseinrichtungen

Wie bereits in 2.2 beschrieben, werden hohe Anforderungen an die Sicherheit von FTS gestellt. Diese sind in einer Vielzahl von Normen definiert (vgl. Tabelle 1) und im „Leitfaden FTS-Sicherheit“ [117] vom Fachausschuss FTS der VDI-Gesellschaft „Produktion und Logistik“ zusammengefasst.

Als Sicherheitseinrichtungen werden in [116] unter anderem Not-Aus-Taster, optische und akustische Warneinrichtungen, Bremsen und Personenschutzsysteme genannt. Aus den in 2.1 genannten Gründen soll von diesen im Folgenden lediglich der Stand der Technik des Personenschutzsystems diskutiert werden. Es muss sicherstellen, dass Personen oder andere Hindernisse im Umfeld des FTF sicher erkannt und eine Kollision vermieden wird bzw. bei Kollisionen die in [29] genannten Kontaktkräfte nicht überschritten werden.

Prinzipiell lassen sich Personenschutzsysteme in taktil und berührungsfrei arbeitende Systeme unterteilen. Taktile Systeme wurden insbesondere in den ersten beiden FTS-Epochen eingesetzt. Durch Softschaum, einen Metall- oder Kunststoffbügel realisiert, erzeugen sie ein Signal zum Anhalten des Fahrzeugs, sobald der Bumper mit einem Objekt kollidiert. Dabei ist der Bumper so konstruiert, dass er einerseits die gesamte Breite des FTF abdeckt und andererseits eine starke Verformung erlaubt. Diese mögliche Verformung stellt den zur Verfügung stehenden Bremsweg des FTF dar und ermöglicht die Einhaltung der in [29] festgelegten Grenzen für erlaubte Kontaktkräfte bei Kollisionen. In Bild 1 sind zwei Ausführungen von taktilen Personenschutzsystemen dargestellt.

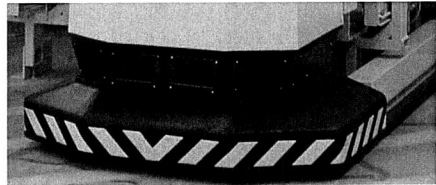
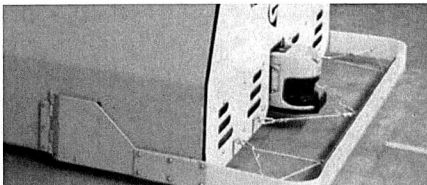


Bild 1: Einrichtungen für den taktilen Personenschutz. Kunststoffbügel (links) und Softschaum-Bumper (rechts). Quelle: [111]

Moderne FTF besitzen zumeist berührungslos arbeitende Personenschutzsysteme. Dabei ist es unumgänglich, dass diese eine berufsgenossenschaftliche Zulassung besitzen. Zu diesem Zweck kommen häufig Sicherheits-Laserscanner (siehe Tabelle 4) zum Einsatz. Die Überwachungsbereiche sind je nach Scannertyp in verschiedene Schutz- und Warnfelder einteilbar.

Die hohen Kosten und das hohe Gewicht eines Sick S3000 oder S300(mini) Laserscanners machen ihn zur Verwendung bei kleinskaligen VTS mit angestrebten Fahrzeugkosten von ca. 1000 € ungeeignet.

## 2 Stand der Technik Fahrerloser Transportfahrzeuge

Tabelle 4: Technische Daten von Laserscannern, wie sie als Personenschutzsysteme eingesetzt werden. Die Daten entsprechen jeweils der Standardausführung [92]. Bilder: Sick

	S300mini	S300	S3000
Schutzfeldreichweite maximal	3 m	3 m	7 m
Maximale Warnfeldreichweite	8 m	8 m	49 m
Auflösung bis zu	30 mm	30 mm	30 mm
Scanwinkel	270°	270°	190°
Ansprechzeit	80 ms	80 ms	60 ms
Gewicht	0,8 kg	1,2 kg	3,3 kg
Abmessungen (B x H x T)	102 x 116 x 105 mm <sup>3</sup>	102 x 152 x 106 mm <sup>3</sup>	155 x 185 x 160 mm <sup>3</sup>
Preis ca. (Netto, Stand 2013)	2500 €	3000 €	4500 €
Abbildung			

Alternative Sensoren basieren auf der Erfassung der Lichtlaufzeit (time of flight, TOF). Sogenannte 3D-TOF Kameras (z. B. [70]) besitzen verschiedene Vorteile gegenüber Laserscannern und Stereokamerasystemen. So besitzen sie eine sehr hohe Bildrate im Vergleich zu 3D-Scannern und erfordern im Vergleich zu Stereokamerasystemen keinerlei Rechenoperationen um Tiefeninformationen für jedes Pixel zu liefern. Allerdings besitzen sie eine relativ geringe Auflösung von ca. 200 x 200 Pixel. Während 3D-TOF Kameras bei aktuellen Forschungsprojekten zur Servicerobotik häufig eingesetzt werden (z. B. Care-O-Bot 3 [7]), ist dieser Sensor für das in dieser Arbeit beschriebene VTS als Personenschutzsystem nicht geeignet, weil sich die Kosten für diesen Sensor zwischen 1000 und 10000 € bewegen [41].

Auch die seit 2010 verfügbare Kinect Kamera liefert Tiefendaten. Ihre Auflösung von 640 x 480 Bildpunkten liegt dabei deutlich über der von aktuellen TOF Kameras. Dabei erzeugt ein IR-Projektor ein definiertes Punkteraster, welches mittels Infrarotkamera erfasst wird und die Errechnung des Tiefenprofils ermöglicht. Aufgrund des niedrigen Anschaffungspreises von unter 100 € wird sie bereits vielfach in Automatisierungsprojekten zur 3D-Objekterkennung eingesetzt. Im Bereich der Personensicherheit lässt sie sich jedoch unter anderem aufgrund der nicht vorhandenen Echtzeitfähigkeit nicht verwenden [121].

Die genannten alternativen Sensoren sind durchweg nicht berufsgenossenschaftlich als Personenschutzsystem zugelassen.

## 2.5 Mobile Systeme in Forschung und Wirtschaft

Mobile Systeme zum Transport von Gütern kommen in einer Vielzahl von Unternehmen in extrem großer Variantenvielfalt zum Einsatz. So werden beispielsweise in der KFZ-, Papier-, Lebensmittel- und Pharmaindustrie [114], in Krankenhäusern [112] und in Containerterminals [36] FTF eingesetzt.

Ein Großteil moderner Anlagen ist mit Magnetnavigation ausgestattet, jedoch kommen auch andere der in 2.3 beschriebenen Methoden zur Standortbestimmung zum Einsatz. Auf optischer Leitlinienführung basiert zum Beispiel das FTF bei MTU Aero Engines [113], welches zur Montage von Turbinenbaugruppen verwendet wird.

Neben herkömmlichen FTF mit großer Fahrzeugmasse werden zunehmend kleine FTF in der Industrie betrieben. Im Folgenden wird eine kleine Auswahl von FTF mit geringer Nutzlast in Forschung und industriellem Einsatz vorgestellt. Anschließend richtet sich der Blick auf aktuelle Arbeiten in der Servicerobotik, wo die Technologien erforscht werden, die in naher Zukunft auch in FTF zum Einsatz kommen werden. Zum Abschluss des Kapitels kommen zwei Systeme zur Sprache, die auf dem Einsatz von Deckenkameras basieren, da in 4 als Grundkonzept für das VTS der gleiche Ansatz favorisiert wird.

### *Kleinskalige Transportsysteme*

Aktuelle Forschungsprojekte zielen darauf ab, die Flexibilität von Logistiklösungen zu maximieren. Ein Ansatz hierzu ist das Konzept des zellularen Transportsystems, welches auf autonomen Transportfahrzeugen oder autonomen Fördersystemen basiert. Diese Entitäten kommunizieren über Softwareagenten. Somit ist eine vollständig dezentrale Logistiksteuerung realisierbar [39]. Das Transportsystem Multishuttle Move (MSM) [54] wird vom Fraunhofer Institut für Materialfluss und Logistik (IML) entwickelt. Es basiert auf autonomen Fahrzeugen, die sowohl als Regalbediengerät als auch als FTF außerhalb des Regals arbeiten können. Die Navigation eines Fahrzeugs basiert auf natürlichen Umgebungsobjekten, die mit Hilfe der vorhandenen Sicherheits-Laserscanner erfasst werden können. Außerdem werden zur Navigation Inertialsensoren, Odometrie und Funkortung eingesetzt. In Bild 2 ist ein entsprechendes Fahrzeug dargestellt. Es besitzt eine Nutzlast von 40 kg. Mit einem Eigengewicht von 134 kg, einer Länge von 1135 mm und einer Breite von 706 mm ist es jedoch verhältnismäßig schwer und groß. Das bringt neben hohen Fahrzeugkosten den Nachteil mit sich, dass der Transport kleiner Werkstücke innerhalb einer Fertigungslinie viel Platz benötigt. Zur Realisierung des Transports eines Werkstückes durch den gesamten Fertigungsprozess ist das MSM zu groß.

Ein weiteres Beispiel für zellulare Fördertechnik stellt das Kleinskalige Autonome Redundante Intralogistik System (KARIS) dar [34]. Es wurde in der Arbeitsgemeinschaft Technologie Intralogistik Baden-Württemberg entwickelt. Die mobilen Grundelemente des Systems (siehe Bild 2) sind modular aufgebaut und bestehen aus einer omnidirektional beweglichen Antriebsplattform mit einer Nutzlast von bis zu 250 kg, einer Sensorplattform, die neben weiteren Sensoren zwei Sicherheits-Laserscanner der Firma Sick (vgl. Tabelle 4) enthält und einem Fördermodul, um Lasten aufzunehmen

oder im Verbund mit anderen Grundelementen zu fördern. Die Navigation der Fahrzeuge basiert auf natürlichen Umgebungsobjekten. Mit einer Größe von 500 mm x 500 mm ist Karis bereits um einiges kleiner als das MSM. Aufgrund der verwendeten Lasersensoren, den resultierenden hohen Fahrzeugkosten und der Größe des Fahrzeugs stellt allerdings auch KARIS keine optimale Lösung für den kompletten innerbetrieblichen Materialfluss dar.

Die Kleine Automatische Transporteinheit (KATE) [27] der Firma Götting wird in Kooperation mit dem Institut für Fördertechnik und Logistik (IFT) der Universität Stuttgart entwickelt. Sie kann eine Nutzlast von 25 kg (optional 50 kg) transportieren. Die Grundfläche des Transportfahrzeugs beträgt dabei 40 x 60 cm<sup>2</sup> und ist für Kleinladungsträger (KLT) optimiert (siehe Bild 2). Im Gegensatz zu vielen anderen Projekten ist bei KATE eine optische oder induktive Leitlinie vorgesehen. Da der Personenauffahrtsschutz mittels taktiler Sensoren erfolgt, kann bei diesen Fahrzeugen auf den Einsatz von Sicherheits-Laserscannern verzichtet werden. Dies soll sehr geringe Fahrzeugkosten von unter 1000 € ermöglichen. Eine zentrale Leitsteuerung lässt den Betrieb von mehr als 100 Fahrzeugen zu. Hinsichtlich Fahrzeugkosten, Fahrzeugabmaßen und Zuladung stellt KATE eine Optimallösung für den innerbetrieblichen Transport von Gütern geringer Größe und Masse dar. Aufgrund der verwendeten leitlinienbasierten Spurführung hat KATE jedoch Mängel hinsichtlich Flexibilität. Zur Änderung der Transportwege muss die Leitlinie verlegt werden und im Schwarmbetrieb kann ausschließlich hintereinander gefahren werden.

Als Forschungsträger für Schwarmintelligenz und spritzgegossene Schaltungsträger (Molded Interconnect Devices, MID) wurde der BeBot [35] am Heinz Nixdorf Institut der Universität Paderborn entwickelt. Die ca. 9 x 9 cm<sup>2</sup> großen und 7 cm hohen Roboter sind mit zwölf IR-Sensoren und einer Kamera zur Umgebungserfassung ausgerüstet. Daneben existieren weitere Sensoren zur Navigation wie Beschleunigungssensoren und ein Drehratensensor. Aufgrund der geringen Abmaße und des resultierenden geringen Gewichts werden beim BeBot keine Sicherheits-Laserscanner eingesetzt. Somit resultieren verhältnismäßig niedrige Fahrzeugkosten von ca. 2200 €. BeBot ist nicht dafür ausgelegt, nennenswerte Nutzlasten zu transportieren. Als Forschungsträger zeigt er jedoch, dass der Schwarmbetrieb mehrerer Roboter mit sehr kostengünstigen und sensorarmen Robotern möglich ist.

Bereits im kommerziellen Einsatz befindet sich der Autonomous Mobile Robot (AMR) [100] (siehe Bild 2) der Firma Swisslog. Mit seiner Nutzlast von 25 kg stellt er eine effiziente Lösung für den Warentransport innerhalb von Krankenhäusern dar. Mittels Lasersensoren wird sowohl der Personenschutz als auch die Navigation, basierend auf natürlichen Umgebungsobjekten, realisiert. Zusätzlich ist es dem AMR möglich, sein Umfeld durch akustische Signale zu warnen. Um automatisierte Türen oder einen Fahrstuhl zu bedienen, ist ein Interface vorhanden.

Kommerziell verfügbar ist ebenso der mobile Roboter Lynx [2] der Firma Adept. Mit seiner Nutzlast von 60 kg stellt er ein flexibles und einfach einzusetzendes Transportmittel für Logistik-, Pharma-, Medizin-, Automobil- oder Krankenhausanwendungen dar.

Dabei erfolgt die Navigation anhand natürlicher Umgebungsobjekte, die mittels Sicherheits-Laserscanner abgetastet werden. In Bild 2 ist ein Adept Lynx abgebildet.

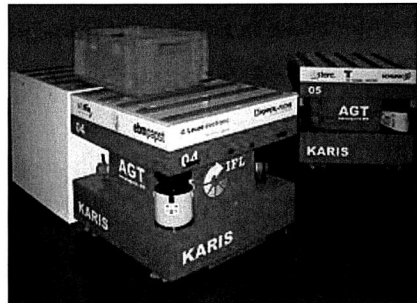
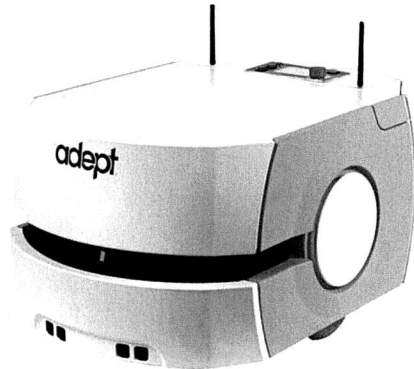
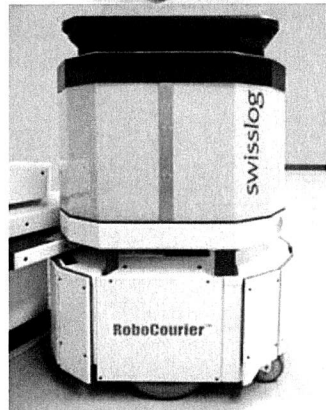
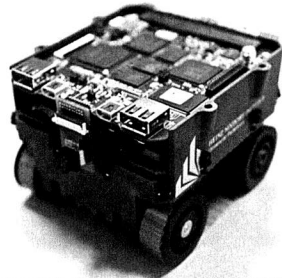
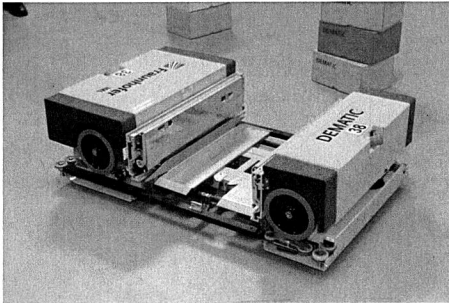


Bild 2: Kleinskalige FTF aus Industrie und Forschung (von oben links nach unten rechts): Multishuttle Move (Quelle: dematic.com), BeBot (Quelle: uni-paderborn.de), Kate (Quelle: goetting.de), AMR (Quelle: swisslog.com), Lynx (Quelle: handling.de), Karis (Quelle: kit.edu)

Von den genannten sechs Systemen sind aktuell lediglich der Swisslog AMR und der Adept Lynx im industriellen Einsatz. Die Investitionskosten für einen solchen Transportroboter liegen bei 125.000 € (inkl. Wartungsvertrag für 1 Jahr) bzw. ca. 100.000 €. Das

macht es unwirtschaftlich, diese Fahrzeuge im Schwarm bzw. in großer Menge für den flexiblen, innerbetrieblichen Materialfluss einzusetzen.

### **Servicerobotik**

Die Servicerobotik erfüllt nach der Definition der International Federation of Robotics (IFR) Dienste für das Wohlbefinden der Menschen, jedoch explizit keine Tätigkeiten in der Fertigungsindustrie [48]. Da die autonome Interaktion mit Menschen in der für den Menschen optimierten Umgebung (z. B. Haushalt) für Roboter extrem herausfordernd ist, werden auf diesem Gebiet momentan die anspruchsvollsten Forschungsarbeiten durchgeführt. Themen wie „visuelle Navigation“, „mobile Manipulation“ und „Greifen“ werden hierbei adressiert. Um diese Aufgaben erfüllen zu können, müssen umfangreiche Grundvoraussetzungen bezüglich Mechanik, Sensorik, Regelungstechnik, Pfadplanung, Computervision und maschinellem Lernen gegeben sein. In Bezug auf FTF sind diese hoch komplexen und weit entwickelten Roboter insofern interessant, als dass FTF in Zukunft zunehmend autonom und flexibel werden. Während FTF heute meistens fest vorgegebenen Wegen folgen und selten autonom agieren, ist zu erwarten, dass Lösungen der Servicerobotik bezüglich Umwelterkennung und (autonomer) Navigation in Zukunft verstärkt bei FTF eingesetzt werden. Genauso wie zum Thema kleinskalige Transportfahrzeuge kann an dieser Stelle lediglich eine kleine Auswahl von bekannten Projekten zur Servicerobotik vorgestellt werden.

Eines der populärsten Beispiele ist der laut dem Hersteller Honda weltweit am weitesten entwickelte humanoide Roboter Asimo [87]. Mit seiner Größe von 130 cm, seinem Gewicht von 54 kg und 34 Freiheitsgraden (degree of freedom, DOF) ist er in der Lage, im Haushalt oder im Büro Treppen zu steigen (siehe Bild 3), Lichtschalter zu bedienen, Türen zu öffnen, Objekte zu tragen oder Wagen zu schieben. Die Entwicklung von Asimo reicht zurück bis ins Jahr 1986, als Honda begann, humanoide Roboter zu erforschen. Ziel ist es, einen Roboter zu entwickeln, der die komplexe Bewegungsweise von Menschen beherrscht. Er soll als Haushaltshilfe, als Unterstützung für behinderte oder alte Menschen oder für gefährliche Aufgaben wie Brandbekämpfung oder Beseitigung toxischer Abfälle eingesetzt werden.

Ebenso ein humanoider Serviceroboter wird seit dem Jahr 2000 am Karlsruher Institut für Technologie (KIT) entwickelt. Die aktuelle Version ARMAR III [8] besitzt 43 DOF. Hier wurde insbesondere Wert auf die Nachempfindung der sensomotorischen Fähigkeiten des Menschen gelegt. Dazu besitzt der Kopf von ARMAR III zwei Farbkameras pro Auge, je eine für Nah- und Fernsicht sowie sechs Mikrophone, um Schallquellen akustisch lokalisieren zu können und einen 6D-Inertialsensor. Um seiner geplanten Aufgabe – Unterstützung im alltäglichen Haushalt – nachgehen zu können, besitzt ARMAR III zwei Arme mit je 7 DOF und zwei Hände mit je 8 DOF sowie eine holonome mobile Plattform. In Bild 3 ist der ARMAR III in den Versionen a (rechts) und b (links) zu sehen.

Nicht humanoid aber ebenfalls als Haushaltsassistent gedacht ist der seit 1998 am Fraunhofer-Institut für Produktionstechnik und Automatisierung (IPA) entwickelte Serviceroboter Care-O-Bot [37]. Bild 3 zeigt ihn in der aktuellen Version 3. Ausgestattet

mit einem 7-DOF Roboterarm und einem 7-DOF 3-Finger Greifer, einem in 1 DOF beweglichen Tablett, einer holonomen mobilen Plattform und einem Sensorkopf, ist Caro-O-Bot 3 in der Lage, typische Haushaltsgegenstände zu erkennen und zu handhaben. Neben den im Sensorkopf angebrachten Stereo-Kameras und der 3D-TOF Kamera besitzt der Roboter unter anderem zwei Sicherheits-Laserscanner. Damit kann der Care-O-Bot 3 auch in der Öffentlichkeit z. B. auf Messen oder im Altenheim eingesetzt werden.

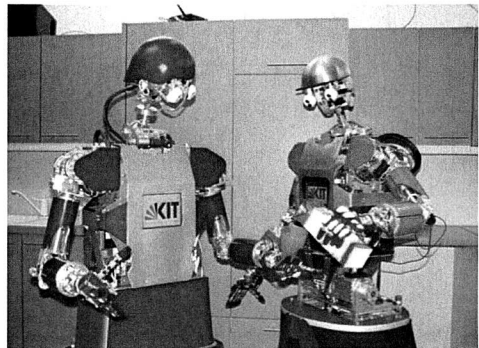
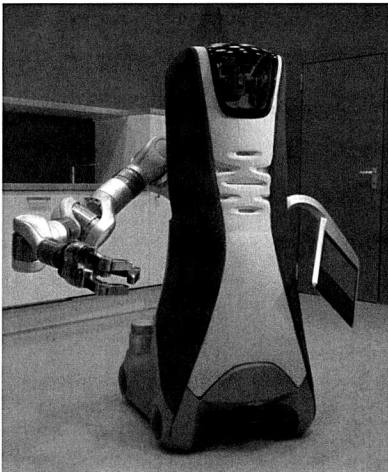
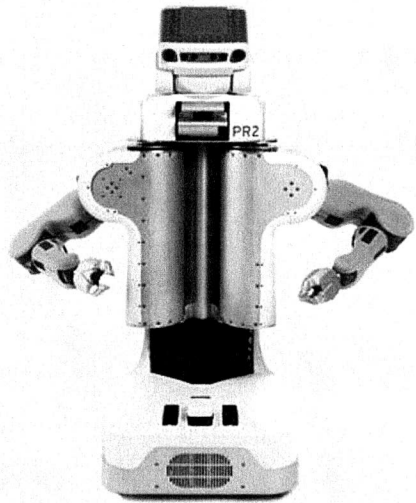
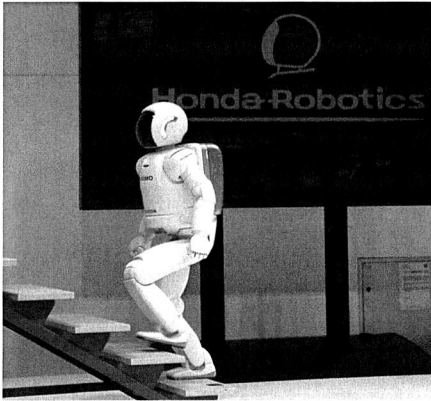


Bild 3: Autonome Serviceroboter (von oben links nach unten rechts): Asimo (Quelle: honda.com), PR2 (Quelle: willowgarage.com), Care-O-Bot 3 (Quelle: care-o-bot.de) und ARMAR III (Quelle: kit.edu)

Als letztes Beispiel eines Serviceroboters sei hier der von Willow Garage entwickelte Personal Robot 2 (PR2) [68] (siehe Bild 3) beschrieben. Dieser dient als Forschungsplattform mit dem Ziel, schnell und einfach neue Roboteranwendungen und Technologien zu testen und zu entwickeln. Ausgestattet mit zwei 7-DOF Roboterarmen und

jeweils einem Greifer mit 1 DOF sowie Sensoren wie einer Microsoft Kinect Kamera, einer Stereo-Kamera, Laserscannern und Inertialsensoren, sind eine Vielzahl von Anwendungen im Haushalt oder bei der Altenpflege denkbar. Willow Garage hat sich zum Ziel gesetzt, wiederverwendbare Software zu entwickeln, mit dem Ziel, „das Rad nicht mehrfach zu erfinden“ und damit Entwicklungszeit hinsichtlich Hardware- und insbesondere hinsichtlich Softwareentwicklung zu reduzieren. Hierzu wird seit 2007 das Robot Operating System (ROS) entwickelt, welches hardwareunabhängig verschiedene Gerätetreiber und Programmbibliotheken mittels effektiver Kommunikationsmechanismen zu einem leistungsfähigen Programm kombinieren kann. ROS wird beim PR2 aber auch bei vielen anderen Robotern wie z. B. dem Care-O-Bot III als Softwareinfrastruktur verwendet. Da davon ausgegangen werden kann, dass ROS in Zukunft eine große Rolle bei der Entwicklung von Robotikanwendungen einnehmen wird, ist es in 7.3.4 beschrieben und wird im Rahmen der vorliegenden Arbeit eingesetzt.

Die genannten Roboter werden zurzeit hauptsächlich zu Forschungs- und Entwicklungszwecken eingesetzt. Aufgrund der aufwändigen Sensorik und Aktorik sind sie momentan noch sehr teuer (z. B. Care-O-Bot: 250.000 €, PR2: 400.000 \$). Jedoch sinken die Preise rapide. Bereits heute sind autonome Haushaltshilfen in Form von Staubsaugerrobotern sehr populär. Beispielsweise sei hier der mit einem Laserscanner bestückte, 600 € teure (Stand 2013), Kobold VR100 [25] der Firma Vorwerk genannt.

### ***Deckenkamera-basierte Systeme***

Da in dieser Arbeit die Strategie verfolgt wird, VTF mittels Deckenkameras zu erfassen (vgl. 4), wird an dieser Stelle über den Stand der Technik bei Fahrerlosen Transportfahrzeugen und Servicerobotern hinaus die small size robot league des RoboCup vorgestellt. In dieser Liga treten jeweils zwei Teams aus maximal sechs Robotern gegeneinander an, um einen Ball in das gegnerische Tor zu befördern. Dabei darf jeder Roboter maximal 150 mm hoch sein, während seine Grundfläche in einen Kreis mit 180 mm Radius passen muss. Über dem Spielfeld sind zwei Kameras montiert. Deren Daten werden aufbereitet, sodass jedem Roboter permanent die Position aller Roboter und des Balls zur Verfügung gestellt werden kann. Die Verarbeitungszeit für einen Frame beträgt dabei ca. 6,2 ms [127]. Hierzu ist es notwendig, dass alle Roboter auf ihrer Oberseite eindeutig optisch markiert sind. Ebenso besitzt der Ball die Farbe Orange, die eine optische Erkennung vor dem grünen Hintergrund erleichtert.

Ein weiteres System, welches auf dem Einsatz von Deckenkameras beruht, ist das Konzept Monamove der TU Braunschweig [58]. Hier werden Kameras über den Fahrwegen der Fahrzeuge angebracht. Diese globale Bilderfassung liefert die Fahrzeugposition und gleichzeitig Umweltdaten, sodass mobile und stationäre Hindernisse in die Bewegungsplanung einbezogen werden können. Bildverarbeitung und Pfadplanung werden bei diesem System zentral durchgeführt. Die Fahrzeuge sind optisch markiert und können leicht mit Hilfe der Deckenkameras erkannt werden.



## 2.6 Fahrzeugkinematiken mobiler Roboter











Prinzipiell ist man bestrebt, FTF eine maximale Beweglichkeit zu ermöglichen. Maximale Beweglichkeit in der Ebene bedeutet dabei, dass alle 3 DOF, die die Bewegung in der Ebene bietet (Translation in zwei linear unabhängigen Bewegungsrichtungen und Rotation um die Hochachse) unabhängig voneinander erzeugt und überlagert werden können.

Dies ist jedoch bei Weitem nicht für alle Fahrzeuge notwendig. So genügt dem gewöhnlichen PKW eine Manövrierbarkeit in 2 DOF (Vortrieb und Lenkwinkel), um jede erdenkliche Pose in der Ebene – wenn auch über gewisse Rangierbewegungen – einzunehmen.

In [22] erfolgt die Einteilung von Kinematiken mobiler Roboter in fünf verschiedene Klassen. Dabei wird zunächst der Grad der Mobilität (degree of mobility)  $\delta_m$  eingeführt. Dieser gibt an, in wie vielen voneinander unabhängigen DOF eine ebene Bewegung ausgeführt werden kann, wobei hierfür ausschließlich die Geschwindigkeit der Traktionsantriebe verändert werden darf. Dabei ist die Anzahl der vorhandenen Traktionsantriebe beliebig, jedoch berechnet sich deren Geschwindigkeit jeweils aus den  $\delta_m$  Geschwindigkeitsparametern für die vorgegebene Bewegung.

Die Anzahl der unabhängig voneinander vorgebbaren Lenkparameter eines Roboters wird als Grad der Lenkbarkeit (degree of steerability)  $\delta_s$  definiert, welcher null, eins oder zwei betragen kann. Auch hier ist damit nicht die Anzahl der lenkbaren Räder festgelegt. Stattdessen errechnet sich der Lenkwinkel jedes lenkbaren Rads aus den  $\delta_s$  vorgegebenen Lenkwinkelparametern.

Tabelle 5: Die fünf möglichen Kinematiken mobiler Roboter in Abhängigkeit von  $\delta_m$  und  $\delta_s$  entsprechend [22].

	Typ (3,0)	Typ (2,0)	Typ (2,1)	Typ (1,1)	Typ (1,2)
$\delta_m$	3	2	2	1	1
$\delta_s$	0	0	1	1	2
$\delta_M$	3	2	3	2	3
Beispiel	Antrieb mit Mecanumrädern	zwei feste Fahrantriebe, zwei drehbare Stützräder	zwei feste Fahrantriebe, zwei drehbare Stützräder, drehbare Plattform	ein gelenktes Rad und zwei feste Stützräder	zwei Lenkantriebe, zwei drehbare Stützräder
Skizze					
Legende	 Mecanumrad	 festes Antriebsrad	 festes Stützrad	 drehbares Stützrad	 drehbares Antriebsrad

Für alle mobilen Roboter von praktischer Bedeutung gilt, dass die Summe von  $\delta_m$  und  $\delta_s$ , der Grad der Manövrierbarkeit (degree of maneuverability)  $\delta_M$ , mindestens zwei und höchstens drei betragen kann.

Daraus resultieren die in Tabelle 5 dargestellten möglichen Kinematiken.

Der Idealzustand ist der eines omnidirektionalen Roboters (Typ (3,0)) mit  $\delta_m = 3$ . Alle anderen Roboter mit  $\delta_M = 3$  aber  $\delta_s \neq 0$  besitzen lenkbare Räder, die zunächst den korrekten Lenkwinkel einnehmen müssen, um die gewünschte Bewegung durchzuführen. Sie werden zur Abgrenzung von den omnidirektionalen Kinematiken als quasi-omnidirektional bezeichnet. Die Kinematiken mit  $\delta_M = 3$  sind jedoch ausnahmslos holonom. Denn die Pose des Roboters lässt sich in allen DOF unabhängig voneinander verändern.

Trotz der angeführten Vorteile von omnidirektionalen Kinematiken kommen bei vielen aktuellen Forschungsprojekten an mobilen Robotern quasi-omnidirektionale Fahrwerke des Typs (1,2) zum Einsatz. Als Grund hierfür wird insbesondere die hohe Robustheit gegenüber unterschiedlichen Bodenbeschaffenheiten und unebenem Untergrund genannt [50]. Dabei sind diese Roboter nicht auf zwei gelenkte und angetriebene Räder beschränkt. Zumeist besitzen sie stattdessen vier gelenkte Antriebseinheiten. Konkret bedeutet ein Fahrwerktyp (1,2), dass zwei freie Lenkwinkelparameter existieren, um den Momentanpol der Roboterbewegung auf einen beliebigen Punkt der Ebene zu legen. Der Geschwindigkeitsparameter gibt die Rotationsgeschwindigkeit um diesen Momentanpol vor. Durch die Geometrie des Roboters vorgegebene Randbedingungen erlauben daraufhin die Ableitung der individuellen Lenkwinkel und Antriebsgeschwindigkeiten der vier gelenkten Antriebseinheiten.

Nachteilhaft bei diesen Fahrwerken ist insbesondere deren hoher technischer und damit finanzieller Aufwand. Der Anschaffungspreis für die mobile Plattform des Care-O-Bot (vgl. 2.5) von Neobotix (siehe Bild 4, links) wird in [41] mit mindestens 50.000 € angegeben.

Ein alternatives Fahrwerk mit  $\delta_M = 3$  stellt das omnidirektionale Fahrwerke mit sogenannten Mecanumrädern dar. Es wird in der Industrie (z. B. Kuka Omnimove) und in der Forschung (z. B. Neobotix MPO-500) (siehe Bild 4, rechts) eingesetzt. Obwohl es sehr hohe Mobilität ermöglicht, besitzt es im praktischen Einsatz verschiedene Nachteile. Zum einen wandert der Aufstandspunkt bei einer Umdrehung des Rads pro Rolle einmal quer über die Breite des Rads. Dies erzeugt oszillierende Biegemomente in der Radantriebswelle, die bei höheren Geschwindigkeiten zu störenden Vibrationen führen können. Zusätzlich hat die - meistens um  $45^\circ$  geneigte - Anordnung der Rollen auf den Rädern die Konsequenz, dass am Aufstandspunkt die gleiche Kraft senkrecht zur Fahrtrichtung in das Fahrzeug eingeleitet wird, wie in Fahrtrichtung. Das setzt gute Haftreibungskoeffizienten zwischen Rollen und Untergrund voraus, erfordert eine extrem steife Lagerung der Mecanumräder und führt durch Schlupf zu Energieverlusten. Als weiterer Nachteil von Mecanum Rädern gilt deren komplexes mechanisches Design [1]. Zusammenfassend kann gesagt werden, dass Mecanumräder für geringe Transportgeschwindigkeiten eine geeignete Lösung darstellen, bei höheren Geschwindigkeiten oder schlechten Untergrundbedingungen jedoch Defizite aufweisen.

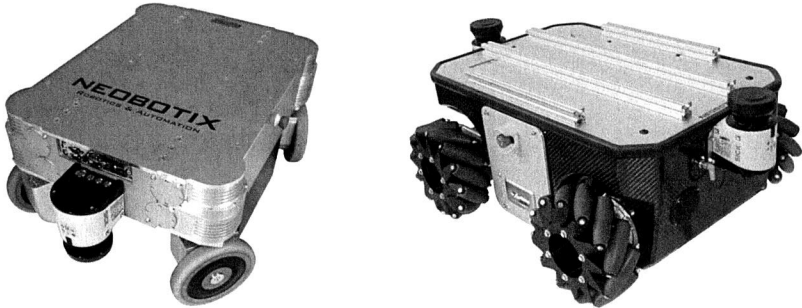


Bild 4: Quasi-omnidirektionaler Antrieb vom Typ (1,2) (links) und omnidirektionaler Antrieb (Typ (3,0)) am Beispiel mobiler Roboterplattformen. Quelle: neobotix-roboter.de.

Als letzte Alternative einer holonomen Plattform bleibt ein quasi-omnidirektionales Fahrwerk vom Typ (2,1), eine Kinematik, die über zwei voneinander unabhängige Geschwindigkeitsparameter und einen (Lenk-) Winkelparameter gesteuert wird. In Tabelle 5 ist die betreffende Kinematik als ein Differentialantrieb mit einer über der Drehachse des Differentialfahrwerks angebrachten drehbaren Transportplattform (DTP) dargestellt. Die Vorteile dieses Fahrwerks gegenüber einem omnidirektionalen Antrieb liegen insbesondere in der niedrigen mechanischen Komplexität dieser Lösung, die mit einfachen Rädern und einfacher Radlagerung auskommt. Zusätzlich zu den beiden Antriebsrädern ist mindestens ein Stützrad notwendig. Um die Standsicherheit des VTF zu maximieren, sind beliebig viele weitere Stützräder am VTF anbringbar. Diese sollten im Idealfall lediglich Kräfte normal zum Untergrund übertragen, also weder Dreh- noch Fahrbewegungen des VTF behindern. Als Nachteil unangetriebener Stützräder muss an dieser Stelle genannt werden, dass diese ungeeignet sind, um Hindernisse zu überwinden. Demnach ist auch für den Betrieb von Fahrwerkskinematiken vom Typ (2,1) (bzw. (2,0)) ein qualitativ hochwertiger, ebener Untergrund Voraussetzung.

Zusätzlich zu den beiden Antriebsmotoren wird lediglich ein weiterer Elektromotor zur Betätigung der DTP benötigt, um ein Fahrwerk des Typs (2,1) zu realisieren. Somit sind mit der minimal möglichen Anzahl an Elektromotoren ebene Bewegungen in allen 3 DOF unabhängig voneinander erzeugbar. Während dies auch mit einem Fahrwerk vom Typ (3,0) möglich ist, sind bei einem Fahrwerk des Typs (1,2) mindestens vier Elektromotoren zum Betätigen von zwei lenkbaren Antriebseinheiten notwendig. Die beiden in Bild 4 dargestellten Roboter benötigen hingegen acht bzw. vier Elektromotoren zum Betrieb.

In der VDI 2510 sind verschiedene Fahrwerkstypen Fahrerloser Transportfahrzeuge aufgeführt. Die dort als flächenbeweglich gekennzeichneten Fahrwerke entsprechen den diskutierten Fahrwerken der Typen (1,2), (2,1) und (3,0).



### **3 Moderne Intralogistik erfordert einen effizienten Transport von Gütern geringer Größe**

Für den Transport von Lasten mit einer Masse von mehreren Tonnen bieten aktuelle FTF eine effektive und wirtschaftliche Lösung [111], [36], [110]. Sind Lasten mit geringen Massen im Bereich weniger Kilogramm individuell zu bewegen, sinkt aufgrund der hohen Fahrzeugkosten die Wirtschaftlichkeit eines FTS. Dabei wird es zunehmend wichtiger, auch geringe Losgrößen von Produkten in hoher Variantenvielfalt effizient und schnell zu fertigen. „Daher wird es zukünftig nötig sein, mit Hilfe einer leistungsfähigen Logistik eine Vielzahl von Nischenprodukten effizient zu fertigen und zu verteilen. Bewährte Logistikkonzepte werden sich fortan nicht mehr in der Breite anwenden lassen; individualisierte Produkte erfordern vielmehr auch eine maßgeschneiderte Logistik.“ [39].

Eine maßgeschneiderte Logistikh Lösung für jedes Fertigungsszenario erfordert hohe Planungs- und Investitionskosten. Daher ist es wichtig, ein universell einsetzbares Transportsystem zur Verfügung zu haben, welches sich aufwandsarm auf unterschiedlichste Logistikprobleme adaptieren lässt. Dabei stellen nicht ausschließlich vielfältige Produkte und Fertigungsverfahren sondern auch verschiedene Fertigungsorganisationen wie verrichtungsorientierte oder flussorientierte Fertigung hohe Ansprüche an die universelle Einsetzbarkeit des VTS.

Daher wird in den folgenden Abschnitten untersucht, welche vielfältigen Ansprüche in modernen Industriebetrieben und Krankenhäusern an die jeweiligen innerbetrieblichen Materialflusslösungen für den Transport geringer Nutzlasten gestellt werden. Die dabei betrachteten, zu transportierenden Güter geringer Masse sind elektronische Drosselvorrichtungen, mittels Durchsteckmontage (through-hole technology, THT) gefertigte Steuerungsplatinen, Waferboxen für 200 mm Wafer und Aktenordner. Auf Basis dieser Untersuchungen lässt sich in 4 ein Konzept für ein VTS ableiten, welches allen identifizierten Ansprüchen genügt.

#### **3.1 Fräsbearbeitung Drosselvorrichtungsgehäuse**

Am Standort Nürnberg eines Automobilzulieferers werden elektronische Drosselvorrichtungen zur Regulierung der in den Brennraum eines Benzinmotors einströmenden Luftmenge hergestellt. Aus dem gesamten Produktionsprozess wird hier die Fräsbearbeitung der Gehäuse herausgegriffen. Diese ist hinsichtlich ihrer Logistikh Lösung interessant, da der Transport jedes Drosselklappengehäuses durch die Fertigungsanlage einzeln auf einem Werkstückträger (WST) erfolgt.

##### **3.1.1 Aufnahme des Istzustands**

Bei den zu transportierenden Gütern handelt es sich um Gehäuse für eine elektronische Drosselvorrichtung. Die Aluminiumdruckgussteile werden in Chargen von mehreren KLT auf einem Rollwagen an die Linie geliefert. Ein KLT beinhaltet neun Gehäuse und

besitzt eine Masse von ca. 20 kg. In der Fertigungslinie werden die Drosselvorrichtungsgehäuse einzeln auf Werkstückträgern transportiert.

Die Fertigungslinie zur spanenden Bearbeitung der Drosselklappengehäuse ist, wie in Bild 5 zu sehen, aufgebaut. Sie besitzt zwei parallel angeordnete Doppelgurtsysteme, auf deren WST die Drosselklappengehäuse manuell aufgesetzt werden müssen. Nach einer optische Prüfung der Werkstücke greift Roboter 1 zwei Gehäuse und legt diese in eines der zwei vorhandenen Bearbeitungszentren (BZ) ein. Beide BZ besitzen je eine Doppelspindel, sodass vier Gehäuse parallel bearbeitet werden können. Nach der spanenden Bearbeitung setzt der Roboter 1 die Gehäuse zurück auf die WST des Doppelgurtbands. Es folgt der Transport zur Pickposition von Roboter 2. Hier erfolgt zunächst eine taktile Kontrolle der Bohrungen im Drosselklappengehäuse, bevor die positiv geprüften Teile auf einem weiteren WST abgesetzt werden. Es schließt sich der Transport mittels Rollenförderer zur Waschstation an, bevor eine abschließende Sichtkontrolle und manuelle Entnahme der Gehäuse erfolgt. Die fertig bearbeiteten Werkstücke werden schließlich manuell in KLT eingelegt.

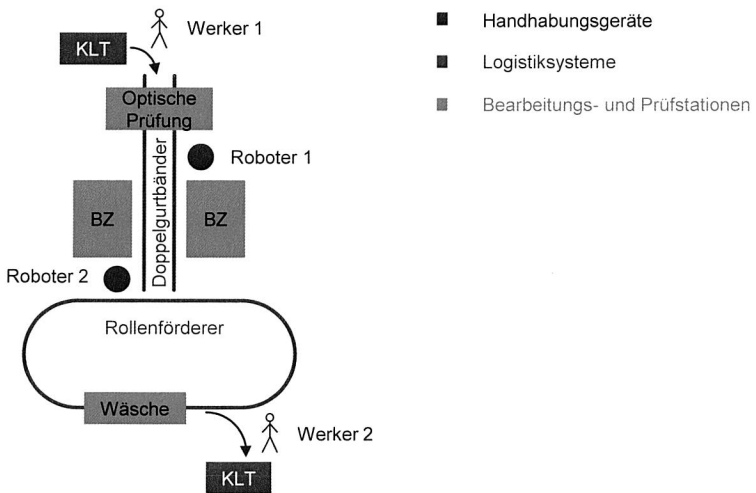


Bild 5: Materialflusssysteme in der spanenden Bearbeitung von Drosselklappengehäusen.

#### 3.1.2 Spezielle Anforderungen an das Transportsystem

An beide Transportsysteme innerhalb der Fertigungsinsel zur spanenden Bearbeitung der Drosselklappengehäuse werden ähnliche Ansprüche hinsichtlich Transportleistung gestellt. Unterschiedlich ist der Anspruch an die Systeme hinsichtlich der Transportumgebung (Feuchtigkeit). Soll der gesamte Transport innerhalb der Fertigungsinsel mit einem Transportsystem (VTS) erfolgen, muss dieses folgenden Ansprüchen genügen:

- Nutzlast ca. 1000 g
- Abmessungen der Nutzlast ca. 15 x 15 x 15 cm<sup>3</sup>

- Transportgeschwindigkeit ca. 12 m/min
- Fähigkeit zum Materialtransport in öligem und feuchtem Gebiet mit konstanter Beleuchtung sowie auf von oben teilweise nicht einsehbaren Fahrwegen
- Fähigkeit zum Materialtransport in feuchter Umgebung (Spritzwasserschutz)
- Möglichst enger Kurvenradius (wenig Platzbedarf)
- Möglichst kostengünstig
- Dreischichtbetrieb

### 3.2 Leiterplattenfertigung für elektrische Antriebslösungen

Die in Nürnberg ansässige Firma entwickelt und produziert elektrische Antriebs- und Automatisierungssysteme. In der Produktion sind ca. 120 verschiedene Leiterplattentypen nach aktueller Bedarfslage herzustellen. Diese große Variantenvielfalt stellt hohe Ansprüche an die verwendete innerbetriebliche Logistiklösung.

#### 3.2.1 Aufnahme des Istzustands

Auf ihrem Weg durch die Fertigung erfolgt zunächst die automatisierte Bestückung der Leiterplatten mit oberflächenmontierten Bauelementen (surface-mounted device, SMD) Bauteilen. Hieran schließt sich, sofern benötigt, die manuelle Montage von THT-Elementen an. Um die zwischen den Stationen zu transportierenden Leiterplatten vor elektrostatischer Entladung (electrostatic discharge, ESD) und mechanischer Beschädigung zu schützen, sind diese in sogenannten ESD-Boxen verpackt. Die Boxen besitzen eine Grundfläche von ca. 60 x 40 cm<sup>2</sup> und verschiedene Höhen zwischen 7,5 cm und 42 cm. Sie beinhalten zehn bis 50 Leiterplatten. Die Masse einer ESD-Box beträgt – je nach Leiterplattentyp – im Mittel ca. 10 kg.

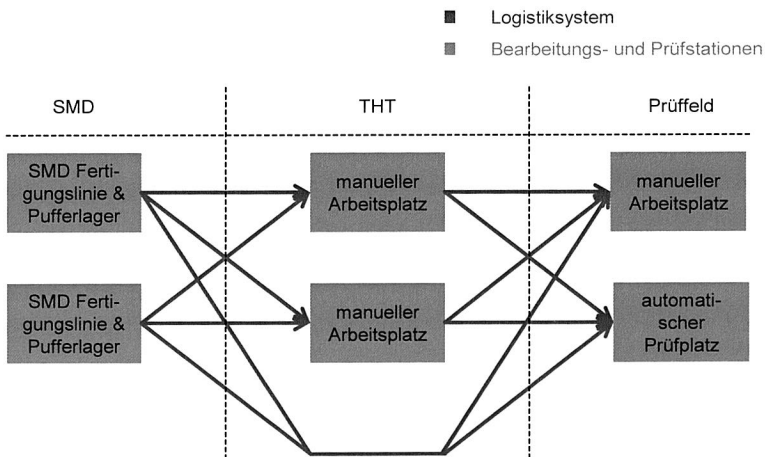


Bild 6: Schematische Darstellung der untersuchten Leiterplattenfertigung mit THT-Elementen.

Unterteilt werden kann die Leiterplattenfertigung in die drei Bereiche SMD-Fertigung, THT-Fertigung und Prüffeld. Eine schematische Darstellung der relativen Zusammenhänge ist Bild 6 zu entnehmen.

Die SMD-Fertigung wird im Dreischichtbetrieb an vollautomatischen Fertigungslinien durchgeführt. Die Verpackung der SMD-bestückten Leiterplatten in ESD-Boxen erfolgt durch einen Werker. Da an den manuellen Arbeitsplätzen zur THT-Montage lediglich im Einschichtbetrieb gefertigt wird, existieren an den SMD-Fertigungslinien Pufferlager für die ESD-Boxen mit den mit THT-Elementen zu bestückenden Leiterplatten.

Schließlich erfolgt im dritten Abschnitt der Leiterplattenfertigung die Prüfung der fertigen elektronischen Baugruppen. Während die dort vorhandenen manuellen Prüfplätze lediglich in der Hauptschicht nach Bedarf besetzt sind, läuft die automatische Prüfung ebenso wie die SMD-Fertigung im Dreischichtbetrieb ab.

Transportiert werden die ESD-Boxen mit Hilfe von speziellen, ESD-geschützten, manuell bewegten Rollwagen. Die damit befahrenen Transportwege besitzen eine Breite von ca. 1 m, sind ebenerdig und mit PVC-Belag ausgestattet.

#### 3.2.2 Spezielle Anforderungen an das Transportsystem

Um den Materialtransport in der betrachteten Leiterplattenfertigung mittels VTS zu realisieren, muss es folgenden Anforderungen genügen:

- Nutzlast ca. 10 kg
- Abmessungen der Nutzlast (B x T x H): maximal 60 x 40 x 42 cm<sup>3</sup>
- Transportgeschwindigkeit ca. 1 m/s
- Extreme Manövrierfähigkeit wegen teilweise engen Wegen
- Erfüllung ESD-Richtlinien
- Automatische Aufnahme von ESD-Boxen
- Dreischichtbetrieb
- Flexibler Betrieb im stark wandelbaren Umfeld (Anfahren verschiedener Zielpositionen je nach Leiterplattentyp, Umfahren von Hindernissen (Werker), einfaches „Teachen“ sich ändernder Zielpositionen, z. B. Verschieben von Handarbeitsplatz)

### 3.3 Halbleiterherstellung

In der Halbleiterfertigung des besuchten Unternehmens werden große Mengen von 8“-Wafern verarbeitet. Die hierbei abzuarbeitende Folge von verschiedenen Fertigungsschritten macht eine leistungsstarke Logistikköslung für den innerbetrieblichen Materialfluss nötig.

#### 3.3.1 Aufnahme des Istzustands

Um die hochgradig empfindlichen Wafer zu transportieren, kommen spezielle Waferboxen, zum Einsatz. Diese beinhalten maximal 25 8“-Wafer. Ihre Abmessungen betragen ca. (B x T x H) 29 x 28 x 25 cm<sup>3</sup> bei einem maximalen Gewicht von 2,5 kg. Das Transportaufkommen liegt bei mehreren tausend Boxen täglich.



In Bild 7 sind die in der betrachteten Halbleiterfertigung verwendeten Transportsysteme dargestellt. Die Fertigung erstreckt sich über drei Fertigungshallen, die eine Reinraumfläche von mehr als 10.000 m<sup>2</sup> besitzen. Während des gesamten Fertigungsprozesses befinden sich die Waferboxen innerhalb von Reinräumen. Somit ist es notwendig, auch die hallenübergreifenden Transportsysteme auf Reinraumniveau zu halten. Dies ist zum einen ein schienengebundenes Transportsystem, welches einen Bahnhof in jeder der drei Fertigungshallen anfahren kann. Dort nimmt jeweils ein Werker die angelieferten Waferboxen entgegen und lagert diese in einem Pufferlager ein. Zum anderen ist das ein Rollenförderer, der lediglich Halle 1 und 3 verbindet. Auch dieser muss manuell be- und entladen werden. Der Transport der Waferboxen von den Pufferlagern zu den jeweiligen Fertigungsstationen und zurück erfolgt ausschließlich manuell.

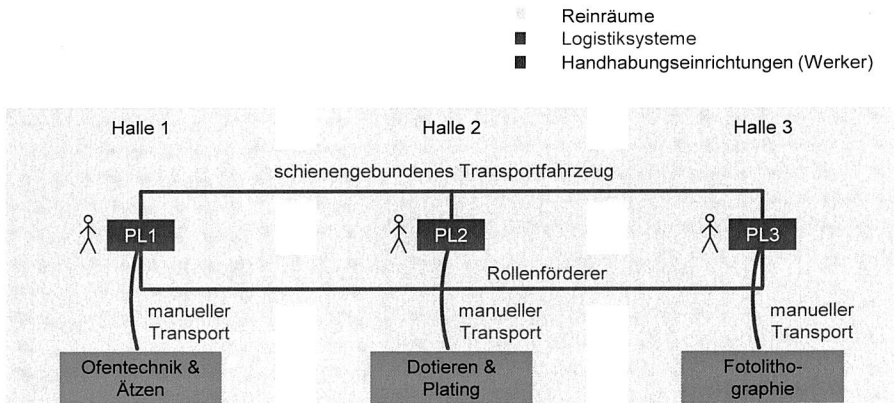


Bild 7: Schematische Darstellung der Logistiksysteme in der Fertigung des besuchten Halbleiterherstellers. Die von den Logistiksystemen angefahrenen Pufferlager sind mit PL1, PL2 und PL3 bezeichnet.

Die verschiedenen Schritte zur Silizium-Halbleiterherstellung [44] wie

- Lithographie (Auftrag von Fotolack und die Belichtung der zu bearbeitenden Gebiete des Wafers)
- Ätzen (Herauslösen bestimmter Bereiche aus dem Silizium oder abgeschiedenen Schichten)
- Dotieren (Einbringen von Fremdatomen in das Silizium-Kristallgitter um elektrische Eigenschaften zu verändern)
- Oxidation (im Ofenprozess durch thermische Oxidation von Silizium)
- Plating (Aufbringen von isolierenden oder leitenden Materialien)

werden in den Hallen 1 bis 3 in verschiedener Reihenfolge und mehrfachen Zyklen ausgeführt. Hieraus resultiert ein enormer Transportaufwand nicht nur innerhalb einer Fertigungshalle von den Pufferlagern zu den Bearbeitungsstationen, sondern auch zwischen den Hallen.

### 3.3.2 Spezielle Anforderungen an das Transportsystem

Die vorhandenen Transportsysteme besitzen unterschiedliche Leistungsdaten und Anforderungen. Es sind daher verschiedene Szenarien für den Einsatz eines VTS denkbar, indem entweder nur eines oder mehrere der Transportsysteme ersetzt werden. Die Anforderungen lassen sich folgendermaßen zusammenfassen:

- Nutzlast (eine Waferbox) ca. 2,5 kg
- Abmessungen der Nutzlast (B x T x H): 29 x 28 x 25 cm<sup>3</sup>
- Transportgeschwindigkeit ca. 1 m/s (von Pufferlager zu Bearbeitungsstation) (Transportgeschwindigkeit schienengebundenes System: 0,3 m/s, Transportgeschwindigkeit Rollenförderer: 0,3 m/s)
- Extreme Manövrierfähigkeit wegen teilweise enger Wege
- Erfüllung Reinraumkriterien
- Automatische Übergabe von Waferboxen an Bearbeitungsstationen
- Dreischichtbetrieb
- Betrieb in starrem Umfeld mit vielen verschiedenen Zielpositionen und langen Transportwegen (bis 700 m von Halle 1 zu Halle 3)
- Kostengünstige Fahrzeuge, da extrem hohe Anzahl von VTF benötigt

### 3.4 Hauspostabwicklung im Klinikum

Das untersuchte Klinikum zählt mit 13 Stockwerken zu den großen Kliniken in Deutschland. Jährlich werden rund 43.000 Patienten stationär behandelt. Hinzu kommen 120.000 Menschen aus dem ambulanten, teilstationären und rehabilitativen Bereich.

Im Rahmen des Tagesgeschäfts fallen enorme Transportaufgaben für Nutzlasten verschiedenster Masse und Volumina an. So ist es notwendig, unter anderem Blutproben, Operations-Abfälle, Wäsche, Medikamente, Essen und Dokumente innerhalb des Krankenhauses zu verteilen. Heute sind in einer Vielzahl von Krankenhäusern AWT-Systeme (Automatischer Warentransport) installiert. Mittels Unterfahrschleppern werden große Container zum Transport von Wäsche oder Essen automatisch zu ihrer jeweiligen Zielposition gebracht. Auch im betrachteten Klinikum existiert eine AWT-Anlage. Diese steht jedoch nicht im Fokus der Untersuchung, da deren Einsatz für Nutzlasten größerer Massen und Volumina bestimmt ist.

Anders als die AWT-Anlage befördert der Zusteller der Hauspost Transportgüter mit einer niedrigen individuellen Masse. Das zugrundeliegende Transportsystem für die Hauspostzustellung wird im Folgenden detailliert erläutert, um anschließend mögliche Anforderungen an ein VTS abzuleiten.

#### 3.4.1 Aufnahme des Istzustands

Von der Hauspost des Klinikums sind vorwiegend Briefumschläge mit einer Masse von weniger als 20 g zu befördern. Das Gewicht einer Hauspostsendung kann jedoch im Einzelfall bis zu 3 kg für einen Aktenordner betragen. Die Maße der Sendungen gehen dabei nicht über das DIN-A4-Format (bzw. Aktenordner für DIN-A4-Format) hinaus. Es

ist zu beachten, dass Briefsendungen nicht geknickt werden dürfen, da sie z. B. Röntgenbilder enthalten können.

Es existieren 69 einzelne Stationen im Klinikum, die an das Hauspostsystem angebunden sind. Diese sind über die 13 Stockwerke des Krankenhauses verteilt, wobei die einzelnen Stockwerke sowohl mittels Aufzug als auch mittels Treppe verbunden sind. Der Bodenbelag des Krankenhauses besteht aus PVC-Fliesen.

Aktuell erfolgt der Transport der Hauspost manuell. An jeder Station existiert ein Pufferlager für die ein- bzw. ausgehende Post. Eine Person fährt diese Pufferlager werktags von 7 Uhr bis 16 Uhr zyklisch mit einem Handwagen ab. Dort nimmt sie jeweils die ausgehende Post auf und legt die an diese Station adressierte Post ab. Dabei erfolgt das Abfahren der Stationen in einer fest vorgegebenen Reihenfolge. Ein kompletter Zyklus über alle Stationen dauert ca. drei Stunden. Somit werden alle Stationen pro Werktag dreimal angefahren. Bild 8 zeigt die Abwicklung der Hauspost am Beispiel von Station n.

Die aktuelle Transportlösung stellt sicher, dass ein versandtes Transportobjekt spätestens nach drei Stunden vom Pufferlager auf den Postwagen und nach sechs Stunden beim Adressaten angekommen ist.

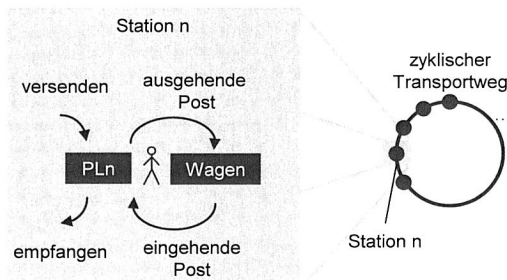


Bild 8: Schematische Darstellung der betrachteten Hauspostabwicklung. Der zyklische Transportweg erstreckt sich über 69 Stationen, wobei jede Station ein Pufferlager (PL) für zu versendende und empfangene Post besitzt.

### 3.4.2 Spezielle Anforderungen an das Transportsystem

Die Zustellung der Hauspost stellt hohe Anforderungen an eine automatisierte Transportlösung. Nicht nur die Vielzahl von verschiedenen Zielstationen ist herausfordernd. Auch die Tatsachen, dass diese Stationen über 13 Stockwerke verteilt, teilweise durch Feuerschutztüren getrennt und Aufzüge hauptsächlich für Patienten reserviert sind, stellen zu erwartende Schwierigkeiten dar.

Soll die Zustellung der Hauspost individuell mittels VTS erfolgen, muss es folgenden Anforderungen genügen:

- Nutzlast (ein DIN-A4-Ordner) ca. 3 kg
- Grundfläche der Nutzlast (liegend DIN-A4-Ordner): 32 x 29 cm<sup>2</sup>
- Transportgeschwindigkeit ca. 1 m/s

### 3 Moderne Intralogistik erfordert einen effizienten Transport von Gütern geringer Größe

- Extreme Steigleistung, um Rampen im Treppenhaus befahren zu können (Benutzung der Aufzüge soll vermieden werden)
- Selbstständiges Überwinden von Feuerschutztüren
- Automatische Übergabe von Post an Pufferlagern
- Einschichtbetrieb
- Betrieb in stark wandelbarem Umfeld mit vielen dynamischen Hindernissen (z. B. Patienten)

## 4 Stationäre Sensorik ermöglicht maximale Flexibilität des VTS

Die in 3 betrachteten Problemstellungen umfassen lediglich eine Diskussion der jeweiligen technischen Aspekte. Hinsichtlich wirtschaftlicher Anforderungen ist zusätzlich zu beachten, dass das VTS eine Transportlösung mit gegenüber herkömmlichen FTF wesentlich erhöhter Fahrzeugdichte (Anzahl Fahrzeuge pro Betriebsfläche) darstellt. Um den Einsatz eines solchen Transportsystems wirtschaftlich zu gestalten, müssen die Fahrzeugkosten gegenüber den in 2.5 beschriebenen (z. B. Adept Lynx 100.000 €) drastisch reduziert werden. In Bild 9 sind die Kosten für Sensortechnik in Abhängigkeit der Fahrzeugdichte eines VTS dargestellt. Die betrachteten Sensoren liefern dabei die Daten, um alle Fahrzeugpositionen zu bestimmen und Hindernisse zu erkennen. Falls das ausschließlich mit den fahrzeugeigenen Sensoren möglich ist (vgl. Swisslog AMR und Adept Lynx), sind keine weiteren Installationen notwendig. Können hingegen fest installierte Sensoren diese Daten bereitstellen, müssen die Fahrzeuge keinerlei Sensorik mit sich führen. Erkennbar ist, dass bei steigender Fahrzeugdichte die Installation stationärer Sensorik zunehmend sinnvoll ist.

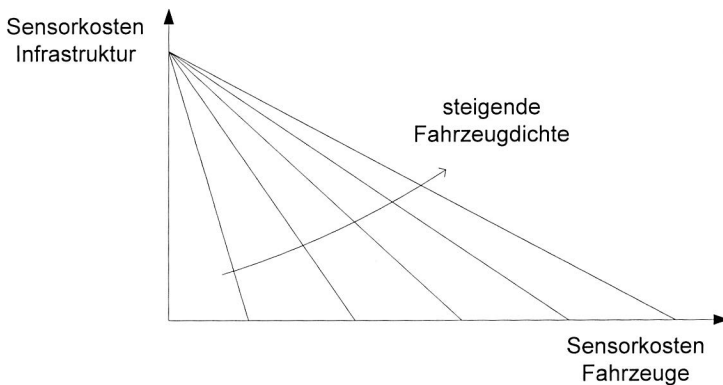


Bild 9: Qualitativer Zusammenhang zwischen Sensorkosten für die Infrastruktur und Sensorkosten für die Gesamtheit aller Fahrzeuge in Abhängigkeit der Fahrzeugdichte eines VTS.

Um ein System zum flexiblen, automatischen Transport kleiner Nutzlasten mit hoher Fahrzeugdichte kosteneffizient zu realisieren, sind daher folgende zwei Leitgedanken wichtig:

- Die Kosten des einzelnen Fahrzeugs müssen so niedrig wie möglich gehalten werden.
- Es ist vertretbar, infrastrukturelle Installationen vorzunehmen, die abhängig von der Größe der Betriebsumgebung des VTS, jedoch unabhängig von der Anzahl der betriebenen VTF sind.

In Abhängigkeit der Fahrzeugdichte  $d$  ergeben sich die vertretbaren Infrastrukturkosten pro Fläche  $k$  aus den Ersparnissen, die bei der Sensorik eines VTF  $s$  gemacht werden können:

$$k = d \cdot s. \quad (4-1)$$

Nach 4-1 sind beispielsweise bei Einsparungen von 1000 €/VTF und einer Fahrzeugdichte von 0,2 VTF/m<sup>2</sup> Infrastrukturkosten von 200 €/m<sup>2</sup> vertretbar.

Um maximale Flexibilität eines VTF zu erreichen, muss es folgende Kriterien erfüllen:

- Fähigkeit zur Standortbestimmung und Lageerfassung der VTF ohne Spurführung
- Fähigkeit zur Zielortbestimmung und Hinderniserfassung (im Idealfall global)
- autonomes Planen und Abfahren des Wegs zum Ziel (Umfahren von Hindernissen)

Diese Kriterien lassen sich durch eine Überwachung der gesamten Betriebsumgebung mittels Kameras erfüllen.

Tabelle 6: Anforderungen an ein VTS

Kriterium	Quantität	Kommentar
Transportkapazität	10 kg	
Transportfläche	32 x 32 cm <sup>2</sup>	Bosch WST (WT 2 – 320/320) [16].
Fahrgeschwindigkeit	1 m/s	Auch hinsichtlich Sicherheitsvorschriften (vgl. 2.2) akzeptabel.
Flexibilität	maximieren	Erfassen von Umgebungsveränderungen (Hindernisse, Zielpositionen) und automatische Reaktion im Sinne optimaler Transportauftragsabarbeitung. Skalierbarkeit hinsichtlich Transportleistung durch einfaches einbringen weiterer Fahrzeuge ins System
Automatische Lastaufnahme	möglich	Maximale Flexibilität in Verbindung mit WT 2 WST.
Einsatzzeit	8 h	3 Schicht Betrieb durch Akkutausch bzw. Nachladen.
Schwarmbetrieb	möglich	Transport größerer und schwererer Lasten durch mehrere VTF gemeinsam (maximale Flexibilität).
VTF Kosten	extrem niedrig, unter 2000 €	Zur Lösung der vorgestellten Transportprobleme bedarf es vieler VTF auf engem Raum. Daher stellen die Fahrzeugkosten einen Preistreiber für ein VTS dar.
Kosten für stationäre Sensorik	mehrere hundert €/m <sup>2</sup>	Je nach Fahrzeugdichte und Sensorkosten der Fahrzeuge (vgl. (4-1))
Sicherheitsanforderungen	gemäß DIN EN 1525 (vgl. 2.1)	Insb. passive Sicherheit maximieren.

Werden die Kameradaten aufbereitet und den VTF zur Verfügung gestellt, können diese autonom eine Transportaufgabe erledigen. Gleichzeitig sind dadurch Lasersensoren zur Lagepeilung eines VTF (vgl. Tabelle 3) nicht notwendig, was geringe Fahrzeugkosten ermöglicht. Um den Sicherheitsanforderungen gemäß DIN EN 1525 gerecht zu werden ist es wichtig, dass die in 2.2 beschriebenen Maximalkräfte bei Kollisionen eingehalten werden.

Aus diesen Überlegungen und den Ergebnissen der in 3.1 bis 3.4 vorgestellten Untersuchungen ergeben sich die Anforderungen, die an ein vielseitiges, automatisiertes Transportsystem gestellt werden. Diese sind in Tabelle 6 dargestellt.

#### **4.1 Das neuartige VTS stellt den VTF ein Weltmodell zur Verfügung**

Aus den vorangegangenen Überlegungen lässt sich das Grundkonzept für ein VTS ableiten. Unter dem Gesichtspunkt der formulierten Leitgedanken sind folgende Konzepte vielversprechend:

- Fahrwerk: Differentialantrieb mit drehbarer Transportplattform (vgl. 2.6)
- Globale Lageerfassung von VTF, Hindernissen und Zielpunkten: Stationäre 2D-Kameras (vgl. 4)

Daher wird im Folgenden das Ziel verfolgt, mit kostengünstigen 2D-Deckenkameras und damit moderaten Investitionen, eine Infrastruktur zu erstellen, die den Betrieb weitestgehend sensorloser VTF ermöglicht. Lediglich Inkrementalgeber in den Antriebsmotoren werden zu deren Drehzahlregelung benötigt.

Zusätzlich ermöglichen diese Sensoren den Fahrzeugen, über Koppelnavigation eine Schätzung der aktuellen Pose bzw. Poseänderung zu errechnen. Die sich durch Schlupf und Modellungenauigkeiten über die Zeit aufintegrierenden Fehler lassen sich durch Fusion mit den Kameradaten reduzieren. Weitere Sensoren zur optischen Spurführung oder Erhöhung der Navigationsgenauigkeit bei Andockvorgängen sowie Abstandssensoren zur Kollisionsvermeidung in abgeschatteten (mit den Deckenkameras nicht erfassbaren) Bereichen, sind installiert. Um die Fahrzeuge optimal klein und einfach zu gestalten, wird auf eine optische Markierung verzichtet. Dies hat einerseits minimale Kostenreduktionen der Fahrzeuge zur Folge. Wichtiger ist aber, dass somit ein Fahrzeug vollständig von der Nutzlast bedeckt sein kann. Somit können einerseits maximal große Werkstücke im Normalbetrieb transportiert werden. Andererseits ist damit die Grundlage geschaffen, mehrere VTF im Schwarmbetrieb gemeinsam eine große und schwere Last transportieren zu lassen.

Im Gegensatz zu den Fahrzeugen werden die Zielpositionen durch eindeutige, individuelle, optische Marker gekennzeichnet. Somit ist das VTS hochgradig flexibel. Soll eine Zielpose verändert werden, muss lediglich der optische Marker verschoben bzw. verdreht werden.

Wie in Bild 10 dargestellt, werden die Bilddaten, die von den Deckenkameras aufgenommen werden, in die drei Kategorien „Zielpositionen“, „Hindernisse“ und „Fahrzeuge“ unterteilt. Diese Informationen werden den VTF bereitgestellt, sodass diese autonom den Weg zu einer vorgegebenen Zielposition planen und abfahren

können. Hierzu benötigen die VTF verschiedene Softwarekomponenten wie Pfadplaner (vgl. 5.2.3) und Tracker (vgl. 5.2.4) oder Motorsteuerungen (vgl. 5.2.1). Weil die Fahrzeuge selbstständig ihre Transportaufträge bearbeiten – also keine Rechenlast an anderer Stelle auftritt – bleibt das VTS skalierbar. Ohne aufwändiges Umrüsten oder Umkonfigurieren können auf diese Weise bei Bedarf weitere VTF in das VTS eingebracht werden.

Die Entwicklung der VTF ist sowohl unter mechanischen als auch unter softwaretechnischen Aspekten in 5 beschrieben.

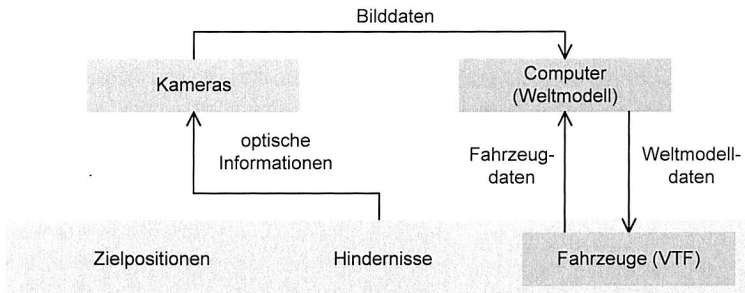


Bild 10: Grundkonzept für ein VTS. Dargestellt sind alle Komponenten und Datenflüssen des Gesamtsystems. Objekte im Sichtbereich der Kameras sind grau hinterlegt.

Um die Daten der Deckenkameras effizient an die Fahrzeuge zu übertragen und gleichzeitig alle anfallenden Sensordaten (von VTF und Kameras) zu einer optimalen Systemzustandsschätzung zu verrechnen, wird ein zentrales, abstraktes Modell der Umwelt erzeugt. In diesem Weltmodell werden alle von den Deckenkameras erkannten Objekte der drei Kategorien „Zielobjekt“, „Hindernisobjekt“ und „Fahrzeugobjekt“ (vgl. 6.1) hinterlegt. Zu jedem Objekt existiert ein Datensatz, der charakteristische Größen wie Orientierung, Abmaße, Position und Geschwindigkeit des Objekts beinhaltet. Dieser Datensatz verbleibt so lange im Weltmodell, wie das zugehörige Objekt von den Kameras detektiert werden kann. Da, wie oben beschrieben, VTF nicht optisch markiert sind, lassen sich diese zunächst nicht von Hindernissen unterscheiden. Die VTF vergleichen jedoch permanent die kürzlich zurückgelegte Wegstrecke mit den Trajektorien aller Hindernisobjekte und können somit feststellen, welches vermeintliche Hindernisobjekt das betreffende VTF repräsentiert (vgl. 5.2.5). Sobald ein VTF dies dem Weltmodell meldet, wird das Hindernisobjekt in ein Fahrzeugobjekt umgewandelt. Eine detailliertere Beschreibung des Weltmodells mit Modellierung der einzelnen Objekttypen, Bildverarbeitungsroutinen und verwendeten Zustandsschätzverfahren folgt in 6.

Das beschriebene Konzept erlaubt den Einsatz sehr kostengünstiger VTF. Im Rahmen des Projekts sind mehrere Prototypenfahrzeuge realisiert worden (vgl. Bild 51). Dabei liegt der Durchschnittspreis für ein VTF bei ca. 1000 € (vgl. 7), die Kosten für Installation einer USB-Deckenkamera und für einen ortsfesten PC bei ca. 600 €. Herkömmliche FTF benötigen sowohl zur Detektion umliegender Hindernisse als auch zur Ermittlung der eigenen Position eigene Sensoren. Zumeist sind das Lasersensoren entsprechend



Abschnitt 2.4. Alleine die Kosten für die genannte Lasersensorik belaufen sich bei zwei Sensoren zur Kollisionsvermeidung (vorne und hinten) und einem Sensor zur Positionsbestimmung auf ca. 13.000 € (Netto).

### **4.2 Anforderungen an Soft- und Hardwarekomponenten werden formuliert**

Entsprechend des Transportsystemkonzepts nach 4, werden in diesem Abschnitt die Anforderungen an die Fahrzeuge und die stationären Komponenten eines VTS abgeleitet, um die Anforderungen aus Tabelle 6 zu erfüllen.

Aufgrund der Überlegungen aus 2.6 kommt als Kinematik ausschließlich ein Fahrwerk vom Typ (2,1) in Frage. Die hierfür benötigte DTP kann optional mittels zweier Fördergurte Werkstückträger von Doppelgurtbändern übernehmen. Diese Option entspricht nicht zwingend dem Grundgedanken kostengünstiger Fahrzeuge. In Betracht zu ziehen wären als Alternative auch stationäre Handhabungsgeräte an den Übergabestationen, die die Fahrzeuge be- und entladen. Da die verschiedenen Möglichkeiten der Beladung und Entladung der VTF stark anwendungsspezifisch sind, werden sie nicht im Rahmen dieser Arbeit diskutiert.

Der Energieverbrauch des VTF beim Betrieb in der Ebene resultiert hauptsächlich aus dem Verbrauch der Traktionsantriebe und dem Verbrauch des Fahrzeug-PCs inklusive WLAN-Kommunikation. Die Beschleunigung auf die angestrebte Maximalgeschwindigkeit von 1 m/s benötigt verhältnismäßig wenig Energie (bei einer angenommenen Fahrzeugmasse von 20 kg 10 J). Bei einer Frequenz von ca. einem Beschleunigungsvorgang alle 30 s werden durchschnittlich lediglich 0,2 W mechanische Leistung benötigt. Jedoch muss aufgrund von Reibungsverlusten sowie Verlusten in der mechanischen und elektrischen Energiewandlung und einer Zuladung von bis zu 10 kg von ca. 10 W elektrischer Dauerleistung ausgegangen werden, um ein VTF zu bewegen. Zusätzlich benötigt der Fahrzeug-PC eine Leistung von ca. 6 W. Beim Befahren einer Rampe sollte von ca. 20 W an Bedarf für elektrische Durchschnittsleistung ausgegangen werden. Je nach zu erwartender Einsatzzeit und den Möglichkeiten der Zwischenladung ist somit der Energiegehalt des elektrischen Energiespeichers vorgegeben. Es wird gefordert, dass ein Betrieb von acht Stunden am Stück möglich sein muss. Somit ist ein Akku mit einer speicherbaren Energiemenge von ca. 160 Wh vorzusehen.

Der Konzeption des VTS liegen die beiden in 4 formulierten Leitgedanken zugrunde. Dabei wird prinzipiell von einer hohen Fahrzeugdichte im mit Kameras überwachten Bereich ausgegangen. Sind Bereiche zu befahren, die aus technischen Gründen nicht durch Kameras erfasst werden können (z. B. unter Handhabungsgeräten in der Fertigungsumgebung) oder nicht effektiv überwacht werden können, weil größere Strecken zu überwinden sind (z. B. lange Verbindungsgänge zwischen zwei Fertigungsinseln), ist es sinnvoll, auf herkömmliche optische Spurführung zurückzugreifen. Daher sind optische Sensoren an der Unterseite des Fahrzeugs anzubringen, mit deren Hilfe einer optischen Leitlinie gefolgt werden kann.

Da die VTF klein sind, muss eine Lösung erarbeitet werden, um Höhendifferenzen zu überwinden. Dies ist insbesondere bei der Interaktion mit Menschen notwendig.

Während sich automatisierte Fügeprozesse prinzipiell in beliebiger Höhe durchführen lassen, befinden sich manuelle Arbeitsplätze ca. 80 cm über dem Boden. Ebenso wie das Be- und Entladen der Fahrzeuge ist eine ggf. notwendige Überwindung von Höhendifferenzen jedoch auf verschiedenste Art und Weise möglich und stark anwendungsabhängig. Unabhängig von der Wirtschaftlichkeit der verschiedenen Lösungen, wird hier eine Möglichkeit dargestellt, mit der die Fahrzeuge selbstständig Höhendifferenzen überwinden können. Hierfür wird eine Rampe konstruiert, die befahren werden kann, ohne dass sich das VTF dabei neigt.

Zusammenfassend sind in Tabelle 7 die aus den Anforderungen an ein VTS abgeleiteten Folgerungen für die mechanische Konstruktion eines VTF dargestellt. In 5.1 wird detailliert auf die mechanische Konstruktion des VTF und einer Rampe zur Überwindung von Höhendifferenzen eingegangen.

Tabelle 7: Anforderungen an das zu konstruierende VTF und sich daraus ergebende Folgerungen für die mechanische Konstruktion

Anforderung	Folgerung
Nutzlast: 10 kg,      Fahrgeschwindigkeit    1 m/s; beweglich in 3 DOF	Fahrwerk vom Typ (2,1) (vgl. 2.6)
Automatische Lastaufnahme (TS2 WST) möglich (optional)	Anbindung an Doppelgurtsysteme. Lastaufnahme mittels Rollen oder Gurten (optional)
Vollständige Lastbedeckung erlaubt	Keine optische Markierung der Fahrzeuge
Robuster Energiespeicher für acht Stunden Betrieb	Akkumulator mit Energiegehalt von ca. 160 Wh
Kostengünstige Realisierung	Low-cost Inkrementalencoder für lokale Positionsschätzung Low-cost Abstandssensoren für Kollisionsvermeidung
Betrieb in abgeschatteten Bereichen (nicht von Kamera erfassbar)	Optische Sensoren zum Folgen einer optischen Leitlinie (Unterbodensensoren)
Autonomer Betrieb von VTF auf Basis des bereitgestellten Weltmodells	WLAN-Verbindung zum stationären Rechner
Einfache Überwindung von Höhendifferenzen	Befahren einer Rampe ohne Neigen des VTF (nur im Individualbetrieb)
Einsetzbarkeit im industriellen Umfeld	Beachten der in DIN EN 1525 genannten Maximalkräfte bei Kollisionen (vgl. 2.2)

Entsprechend den beschriebenen Anforderungen an die Konstruktion eines VTF lassen sich aus 4 ebenso Anforderungen an die Softwarekomponenten eines VTF herleiten. Diese betreffen zunächst die Antriebsmotoren. Sie müssen drehzahleregelt sein, um präzise Manöver fahren zu können (z. B. Kreisbogen mit vorgegebenem Radius). Zur Koppelnavigation ist es wichtig, die zurückgelegten Wegstrecken beider Antriebsräder exakt und möglichst ohne Zeitverzug zu registrieren und zu verrechnen. Nur so kann der Fehler bei der Erfassung von Poseänderungen klein gehalten werden. Um schnell auf Sensorsignale zu reagieren sowie effektive Regelungen zu implementieren, sind

Mikrocontroller die erste Wahl. Hier werden Regler zur Drehzahlregelung und Routinen zur Koppelnavigation implementiert.

Die Signale der IR-Abstandssensoren sowie der optischen Unterbodensensoren werden ebenfalls mittels Mikrocontroller erfasst und in digitale Werte umgesetzt. Hier sind Softwarekomponenten wichtig, die diese Daten optimal verwerten. Mit Hilfe der Abstandssensoren müssen Kollisionen verhindert werden, die Unterbodensensoren liefern Daten zur Spurführung oder zur Genauigkeitssteigerung der Positionserkennung bei Andockvorgängen (bei Navigation auf einem Zielmarker werden die Farbkanten des Schachbrettmusters detektiert).

Die Absolutposition eines VTF lässt sich zunächst nicht auf Basis der Daten der Deckenkameras ermitteln, da die Fahrzeuge nicht optisch von anderen Hindernissen unterschieden werden können. Jedoch können die VTF durch die von ihnen zurückgelegte Wegstrecke Hindernisobjekten zugeordnet werden. Hierfür ist permanent die mittels Inkrementalencoder aufgenommene Trajektorie der VTF mit den Trajektorien aller Hindernisobjekte zu vergleichen.

Geht man davon aus, dass das VTF sowohl seine Pose als auch die ihm zugewiesene Zielpose und die Posen und Abmaße aller Hindernisse kennt, kann es autonom den Weg zur Zielpose planen und abfahren. Hierzu sind Softwarekomponenten wie Pfadplaner und Tracker (zum Abfahren des Pfads) notwendig. Die für ein VTF benötigten Softwarekomponenten sind in Tabelle 8 dargestellt. In 5.2 werden die Softwarekomponenten im Detail beschrieben.

Tabelle 8: Anforderungen an die Softwarekomponenten eines VTF

Anforderung	Folgerung
Koppelnavigation und Motorregelung	Mikrocontroller mit schnellen Motorreglern und präziser Berechnung der Poseänderung des VTF
Betrieb in abgeschatteten Bereichen (nicht von Kamera erfassbar)	Softwareinstanz zur optischen Spurführung
Hohe Navigationsgenauigkeit bei Andockverfahren	Erkennen von Farbkanten der Zielmarker und Ableitung der relativen Position zum Marker. Zustandsschätzverfahren
Vollständige Lastbedeckung erlaubt	Vergleich von lokal aufgenommener Trajektorie des VTF mit allen Trajektorien von Hindernissen. Errechnung eines Ähnlichkeitsmaßes und Ermittlung des zugehörigen Hindernisobjekts.
Autonomer Betrieb von VTF auf Basis des bereitgestellten Weltmodells	Pfadplaner, Tracker

Um die beschriebenen VTF zu betreiben, ist entsprechend des in 4 formulierten Konzepts eine stationäre Umwelterkennung notwendig. Die hierzu eingesetzten Deckenkameras erfassen die Betriebsumgebung und ermöglichen die Erstellung eines Weltmodells. Es stellt, wie in Bild 10 angedeutet, eine mathematische 2D-Repräsentation aller Zielpositionen, Hindernisse und Fahrzeuge dar. Zugrunde liegt hier ein kartesisches Rechts-Koordinatensystem, wie es z. B. in Bild 25 dargestellt ist.

Die sich aus den Anforderungen an ein VTS ergebenden Folgerungen für die Gestaltung der stationären Komponenten des VTS sind in Tabelle 9 aufgeführt.

Tabelle 9: Aus den in Tabelle 6 formulierten Anforderungen an das VTS resultierende Anforderungen an dessen stationäre Komponenten

Anforderung	Folgerung
Sicherheit	zuverlässige Objekterkennung (mittels 2D-Kameras) Verrechnung aller zur Verfügung stehender Sensordaten zur Maximierung der Weltmodellgenauigkeit
Flexibilität maximieren	optische Erkennung von Zielmarkern
Kosten einige hundert €/m <sup>2</sup> (vgl. (4-1))	Verwendung von 2D-Kameras zur Objekterkennung zentrale Verarbeitung der Kameradaten, um Fahrzeugrechner zu entlasten

## 5 VTF sind kleinskalig und agieren autonom

Die VTF stellen die Transportelemente des Systems dar. Somit kommt ihrer anforderungsgerechten mechanischen Konstruktion eine zentrale Bedeutung zu. Um einen autonomen Betrieb der VTF zu ermöglichen, sind zusätzlich aufwändige Softwarekomponenten auf den fahrzeugfesten Rechnern zu installieren. Die hierzu formulierten Anforderungen aus 4.2 dienen als Grundlage für die in den folgenden Abschnitten beschriebenen Arbeiten hinsichtlich Konstruktion und Softwarekomponenten der VTF.

### 5.1 Konstruktion

Um die Konstruktion eines VTS durchzuführen, müssen zunächst die zum Einsatz kommenden Komponenten festgelegt werden. Dies erfolgt entsprechend den Anforderungen aus Tabelle 7. Daran schließen sich grundlegende Überlegungen zur Konstruktion eines VTF an. Eine detaillierte Betrachtung konkret ausgewählter Komponenten und der Konstruktion des VTF erfolgt in 7.1.

#### *Elektrische Antriebe*

Zum Betrieb des VTF werden zwei Traktionsantriebe und ein Antrieb zum Verdrehen der DTP benötigt. Des Weiteren müssen die Gurte zur Lastaufnahme mittels Elektromotor angesteuert werden. Die Berechnungen aus 7.1 zeigen, dass hierbei mit einer elektrischen Gesamtaufnahmeleistung von maximal 110 W auszugehen ist.

Zwecks Regelung der Antriebsmotoren und zur Erfassung der zurückgelegten Wegstrecke ist es notwendig, dass die Motoren mit Winkelsensoren ausgestattet sind. Die Daten dieser Encoder werden, wie in 5.2.1 beschrieben, mittels Mikrocontroller ausgelesen und verarbeitet.

#### *Elektrische Energiespeicher*

Bereits in 5 wurde überschlägig ein durchschnittlicher elektrischer Leistungsbedarf eines VTF von 20 W ermittelt. Somit ergibt sich über acht Stunden ein Energiebedarf von 160 Wh. Diese Energie muss in einem elektrischen Energiespeicher mitgeführt werden, falls keine Zwischenladungen an Andockpositionen vorgenommen werden sollen. Neben der maximal speicherbaren Energiemenge von 160 Wh ist insbesondere die maximale Leistungsabgabe von ca. 110 W ein wichtiges Selektionskriterium.

Leistungsdaten aktueller Akkumulatoren nach [56] und [65] sind in Tabelle 10 aufgeführt. Aufgrund der aufkommenden Elektromobilität stellt dieses Thema aktuell ein Gebiet intensiver Forschung dar.

Entsprechend Tabelle 10 und den o. g. Anforderungen muss ein Kompromiss bezüglich geringer Anschaffungskosten und hoher Energie- und Leistungsdichte der elektrischen Energiespeicher gefunden werden. Um das Fahrzeug möglichst klein und leicht realisieren zu können, wird ein Li-Ion-Akkumulator vorgesehen. Zu dieser Familie zählen auch die Lithium-Polymer-Akkumulatoren (LiPo).

Tabelle 10: Leistungsdaten aktueller Akkumulatoren nach [56] und [65]

	Blei Säure	NiCd	NiMH	Li-Ion
Energiedichte in Wh/kg	20-35 Wh/kg	30-50 Wh/kg	50-80	90-200
Leistungsdichte in W/kg	100	600	1000	300-4000
Zyklusfestigkeit	1200	2000	1000	500-4000
Kosten in €/kWh	100-250	250-500	180-220	300-1800
Kommentar	Anwendung in FTF und Gabelstaplern	umweltschädlich	Memory Effekt	teuer, sehr leistungsstark

### **Motorregler**

Zur Regelung der Traktionsantriebe sind Mikrocontroller basierte Schaltungen ideal geeignet. Im Allgemeinen wird eine PI-Regler gestützte Drehzahlregelung der Motoren realisiert. Auf diese Weise ist sichergestellt, dass geplante Manöver präzise ausgeführt werden können. Neben den beiden Motorreglern kommt ein dritter Mikrocontroller zum Einsatz. Dieser zentrale Controller hat die Aufgabe, mit dem Fahrzeug-PC zu kommunizieren und in Abhängigkeit der Fahrbefehle Sollgeschwindigkeiten für beide Motorregler zu generieren. Ebenso ermittelt der zentrale Mikrocontroller die Änderung der Fahrzeugposition auf Basis der von beiden Motorreglern zurückgemeldeten Winkelinkremente. Zusätzlich werden die Daten aller analogen Sensoren (Abstands- und Unterbodensensoren) auf diesem Mikrocontroller analog-digital gewandelt und an den Fahrzeug-PC weitergeleitet.

### **Fahrzeug-PC**

Um das gesamte Fahrzeug zu steuern und die Sollwerte für die Motorregler zu generieren, wird ein Fahrzeug-PC benötigt. Dieser muss rechenintensive Aufgaben wie Pfadplanung oder Pfadverfolgung erledigen (vgl. Tabelle 8). Zusätzlich ist es notwendig, dass er per WLAN mit dem Weltmodell kommuniziert. Im Sinne einer langen Betriebszeit eines VTF mit einer Akkuladung ist es von Vorteil, die elektrische Leistungsaufnahme des PC zu minimieren.

### **Sensoren**

Neben den Inkrementalencodern der Traktionsantriebe sind in Tabelle 7 weitere Sensortypen genannt, die zum Betrieb des VTF benötigt werden. Zum einen sind das kostengünstige Abstandssensoren, um Kollisionen verhindern zu können. In 7.1 sind die hierfür ausgewählten IR-Abstandssensoren erläutert.

Zum anderen werden Unterbodensensoren benötigt, um einer optischen Leitlinie folgen zu können und die Genauigkeit der Positionsschätzung bei Andockvorgängen zu erhöhen. Dazu ist es notwendig, die Farbkanten des Musters des aktuell überfahrenen Zielmarkers zu erkennen (vgl. 5.2.6). Auch hier ist der Einsatz von IR-Sensoren vorgesehen.

### Passive Sicherheit bei Kollisionen

In 2.1 sind die Sicherheitsanforderungen an moderne FTF beschrieben und diskutiert. Dabei wurde festgelegt, dass die Kontaktkraft, die bei Kollisionen vom VTF auf das Kollisionsobjekt ausgeübt wird, 400 N nicht übersteigen darf. In diesem Abschnitt erfolgt die überschlägige Dimensionierung der passiven Schutzeinrichtungen (Stoßfänger), um bei Kollisionen die geforderte Maximalkraft auf einen Prüfkörper nicht zu überschreiten. Dabei wird von einem energieabsorbierenden Schaumstoff ausgegangen, der über dem gesamten zur Verfügung stehenden Verformungsweg eine konstante Bremskraft erzeugt.

Entsprechend dieser idealisierten Annahme, lässt sich die minimale Dicke des Schaumstoffbelags errechnen, die notwendig ist, um ein VTF bei einer Kollision zum Stillstand zu bringen.

Die maximal erlaubte Verzögerung  $a_{\max}$ , die durch eine Kollision hervorgerufen werden kann, lässt sich in Anbetracht der gegebenen Randbedingungen  $m_{\text{vehicle}} = 30 \text{ kg}$  und  $f_{\max} = 400 \text{ N}$  folgendermaßen angeben

$$a_{\max} = \frac{f_{\max}}{m_{\text{vehicle}}} = \frac{400 \text{ N}}{30 \text{ kg}} \approx 13,3 \text{ m/s}^2. \quad (5-1)$$

Der bei dieser Maximalverzögerung notwendige Bremsweg  $s$ , um das VTF zum Stehen zu bringen, hängt quadratisch von der Geschwindigkeit  $v_{\max}$  des Fahrzeugs ab

$$s = \frac{1}{2} a_{\max} t^2 = \frac{v_{\max}^2}{2a_{\max}}. \quad (5-2)$$

Bei über dem gesamten Bremsweg konstanter Verzögerung ergibt sich nach (5-2) bei einer Geschwindigkeit von 1 m/s ein Bremsweg von 3,76 cm. Beträgt die Fahrzeuggeschwindigkeit 0,5 m/s, so steht das Fahrzeug bereits nach 0,94 cm Bremsweg. Daher wird ein Stoßfänger mit 4 cm starkem Schaumstoffbelag an Vorder- und Rückseite der DTP vorgesehen.

#### 5.1.1 Rechnergestützte Konstruktion

Die in 5.1 ausgewählten Komponenten werden in diesem Kapitel mittels rechnergestützter Konstruktion (computer-aided design, CAD) zu einem vollständigen VTF zusammengesetzt. Entsprechend den Anforderungen nach Tabelle 7 darf sich ein VTF nicht neigen, wenn es eine Rampe befährt. Das Fahrwerk vom Typ (2,1) muss dementsprechend ausgelegt werden.

Grundlegend hierfür ist der Gedanke, dass ein Fahrzeug auf einer schiefen Ebene dann horizontal ausgerichtet bleibt, wenn jedes Rad auf einer eigenen schiefen Ebene rollt. Diese Ebenen müssen gleiche Steigungen besitzen und in Fahrtrichtung um den Abstand der Räder versetzt sein. Um die notwendige Bodenfreiheit eines Fahrzeugs aufgrund der Führung der hinteren Stützräder gering zu halten, wird hier der Ansatz

verfolgt, die vorderen Stützräder mit einer Schiene oben und unten zu umschließen. Damit kann auf eine Führung der hinteren Stützrollen verzichtet werden. Gleichzeitig erlauben die außenstehend montierten Antriebsräder maximale Steigung der schiefen Ebene, da auf diese Weise eine Kollision zwischen Fahrzeugkörper und Fahrschiene verhindert wird. In Bild 11 ist schematisch dargestellt, welche konstruktiven Anforderungen an Fahrzeug und Rampe gestellt werden. Details zur CAD Konstruktion des VTF können in 7.1 nachgelesen werden.

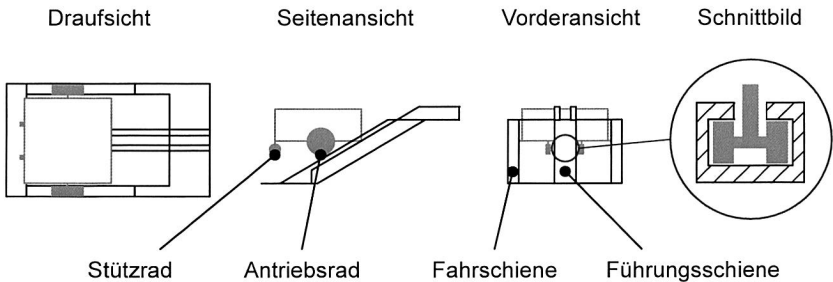


Bild 11: Schematische Darstellung des Fahrwerkkonzepts zum neigungslosen Überwinden von Höhendifferenzen. Grün dargestellt ist jeweils ein Fahrwerk vom Typ (2,0).

Das VTF kann sowohl im Individualbetrieb als auch im Schwarmbetrieb eingesetzt werden. In Bild 12 sind beispielhaft beide Möglichkeiten dargestellt. Als Nutzlast ist ein Werkstückträger WT2 [17] des Transfersystems TS 2plus der Firma Bosch Rexroth zu sehen. Der Werkstückträger hat eine Größe von 320 x 320 mm<sup>2</sup>. Ausschnittsweise zu sehen ist das obere Ende einer Rampe, die das VTF nutzt, um die Höhe des Transfersystems zu erreichen. In 5.1.3 ist die Konstruktion der Rampe detailliert erläutert.

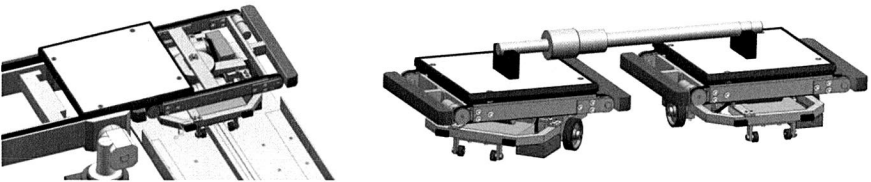


Bild 12: Mögliche Einsatzszenarien für ein bzw. zwei VTF. Individualbetrieb mit Anbindung an ein Transfersystem (links) und Schwarmbetrieb (rechts)

### 5.1.2 Aufbau eines Prototypen

In diesem Abschnitt erfolgt der Entwurf eines Fahrzeugs, mit dem alle Mechanismen und Softwarekomponenten zur autonomen Navigation des VTF entwickelt und erprobt werden können. Hauptsächlichster Unterschied zu der Konstruktion aus 5.1 ist, dass der Prototyp lediglich ein Fahrwerk vom Typ (2,0) besitzt. Dabei besteht der Unterschied zu dem in 5.1 beschriebenen Fahrwerk vom Typ (2,1) hauptsächlich darin, dass die DTP nicht realisiert ist. Aus Gründen der Einfachheit wird zusätzlich auf einen Metallrahmen für das Fahrzeug verzichtet. Stattdessen kommt eine einfache Plexiglakonstruktion,



wie in Bild 13 dargestellt, zum Einsatz. Die Größe des Fahrzeugs ist so gewählt, dass es einen WT2 Werkstückträger [17] transportieren kann.

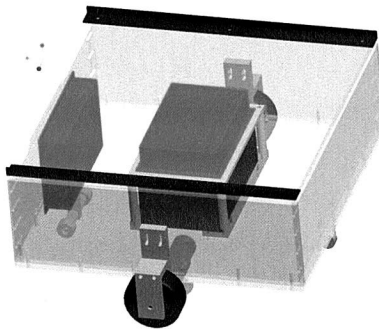


Bild 13: CAD-Modell des vereinfachten Prototyps eines VTF mit einem Fahrwerk vom Typ (2,0).

Neben kostengünstigeren Komponenten für den Antrieb wird für den Prototyp des VTF auch ein einfacherer Energiespeicher als ein LiPo-Akkumulator verwendet. Zum Einsatz kommt ein herkömmlicher Blei-Gel-Akkumulator mit einer Kapazität von 7,2 Ah. Nachteilhaft gegenüber den LiPo-Akkumulatoren ist insbesondere das Gewicht von ca. 2,5 kg (vgl. Tabelle 10).

### 5.1.3 Rampe zur neigungslosen Überwindung von Höhendifferenzen

Wird ein Fahrzeug im Individualbetrieb genutzt, ist es sinnvoll, bei der Befahrung von Steigungen darauf zu achten, dass die Transportplattform sich nicht neigt und somit das Transportgut optimal gelagert ist.

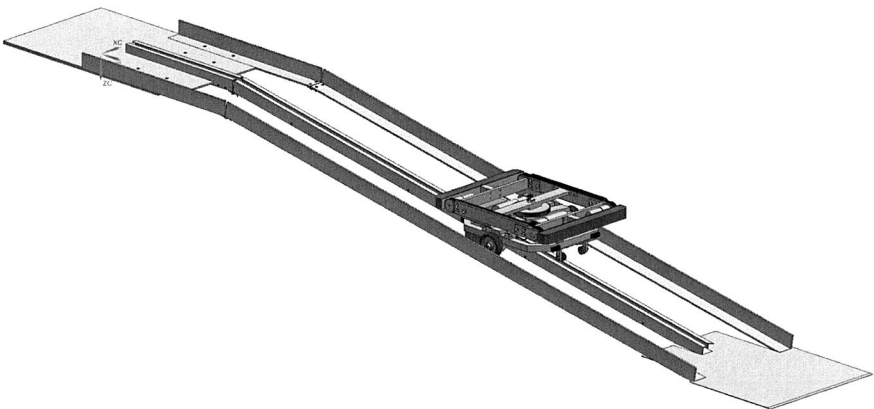


Bild 14: CAD-Modell der Rampe zum neigungslosen Überwinden von Höhendifferenzen mit VTF.

Wie in 5.1.1 erläutert, wird eine Rampe mit einer zu den Fahrschienen in der Tiefe versetzten Führungsschiene (vgl. Bild 11) benötigt. Die zu konstruierende Rampe soll flexibel für verschiedene Steigungen einsetzbar sein.

In Bild 14 ist die Rampenkonstruktion mit VTF dargestellt. Für die Fahrspuren der beiden Antriebsräder kommen L-Profile zum Einsatz. Diese ermöglichen eine seitliche Führung des VTF und bieten gleichzeitig eine ausreichend hohe Biege- und Torsionssteifigkeit. Am oberen Ende werden die Schienen mit Scharnieren gelagert. Am unteren Ende liegen sie auf dem Boden auf und werden seitlich durch die Bodenplatte fixiert. Die Führungsschiene ist um den Abstand zwischen Aufstandspunkt der Antriebsräder und vorderem Stützräderpaar in Fahrtrichtung versetzt. Sie besteht aus einer Grundplatte mit seitlich angeschraubten L-Profilen.

Da der Andockvorgang eine hohe Navigationsgenauigkeit erfordert, wird in 5.2.6 ein Verfahren beschrieben, das eine präzise Positionierung des VTF mit Hilfe optischer Marker ermöglicht.

## 5.2 Softwarekomponenten

Die in 5.1 beschriebene Konstruktion eines VTF besitzt eine Vielzahl elektronischer und elektrischer Komponenten wie Sensoren, Motoren, PC, Mikrocontroller, Akkumulator und Spannungsregler. In diesem Abschnitt werden die Softwarekomponenten beschrieben, die auf dem Fahrzeug-PC und den Mikrocontrollern installiert sind (siehe Bild 15). Dabei wird zunächst auf die Softwarekomponenten für die Mikrocontroller eingegangen. Dann folgen die Erläuterungen für die Softwarekomponenten des fahrzeugeigenen IBM-kompatiblen Fahrzeug-PC. Eine Diskussion der im Rahmen der Entwicklung des VTS verwendeten Softwareprodukte wie Betriebssysteme und Programmiersprachen erfolgt separat in Kapitel 7.3.

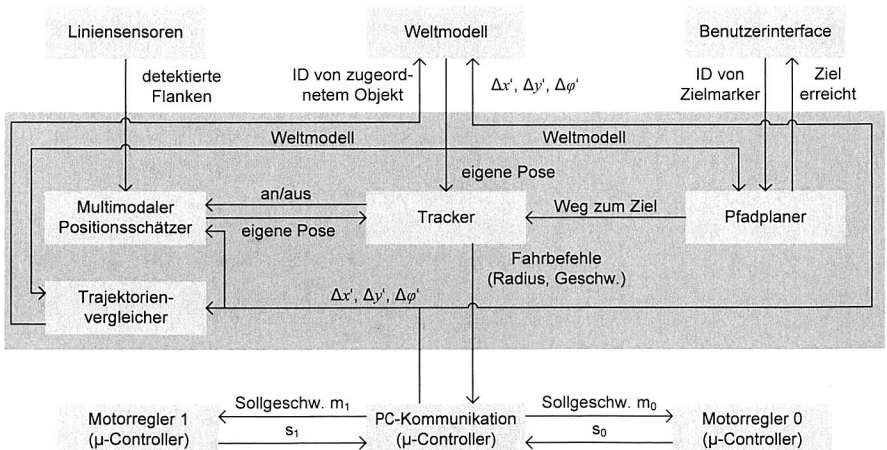


Bild 15: Datenfluss in einem VTF. Blau hinterlegt ist der IBM-kompatible Fahrzeug-PC. Jedes VTF steht mittels WLAN mit dem Weltmodell in Verbindung.

### 5.2.1 Mikrocontroller zur Regelung der Antriebsmotoren

Als Fahrantriebe kommen im Prototyp des VTF mechanisch kommutierte 12 V-Gleichstrommotoren mit einer Getriebeuntersetzung von 30:1 zum Einsatz [85]. Pro Umdrehung der Abtriebswelle liefern die Inkrementalencoder jeweils 180 Inkremente.

In diesem Kapitel wird die Programmierung der Mikrocontroller erläutert, die die Geschwindigkeitsregelung der Fahrantriebe übernehmen. Sie sind in Bild 15 als Motorregler 0 und Motorregler 1 bezeichnet und besitzen folgende Aufgaben:

- Errechnen des Stellsignals ( $pw$ ) für die Antriebsmotorenregelung
- Kommunikation mit dem zentralen Mikrocontroller (über I<sup>2</sup>C-Bus)
- Ansteuerung des Motortreiber-IC [4]
- Auslesen und Aufsummieren der Motorinkremente (zurückgelegte Wegstrecke)

Die Regelgröße  $v_o$  beschreibt in diesem Kapitel stellvertretend die Umfangsgeschwindigkeit des Antriebsrads von Motor 0 oder Motor 1. In Bild 16 ist der zugehörige Regelkreis dargestellt. Zur Erfassung von  $v_o$  kommt ein externer Interrupt des Mikrocontrollers zur Verwendung. Dabei wird die Zeit zwischen zwei detektierten Flanken des Ausgangssignals des Inkrementalencoders mittels Timer gemessen und daraus die aktuelle Geschwindigkeit ermittelt. Dieses Verfahren liefert Messwerte in einer Frequenz, die von der Drehzahl des Motors abhängig ist. Der Einfachheit halber wird hier davon ausgegangen, dass der gemessene Geschwindigkeitswert immer der tatsächlichen Geschwindigkeit  $v_o$  entspricht.

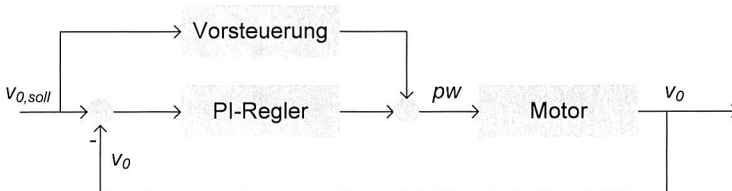


Bild 16: Blockschaftbild des Regelkreises zur Geschwindigkeitsregelung der beiden Antriebsmotoren am Beispiel von Motor 0.

Mittels PWM-Signal lässt sich die Ankerspannung des Motors einstellen. Die Pulsweite ( $pw$ ) ergibt sich aus der Summe von Vorsteuersignal und Reglerausgang. Dabei hat sich die Vorsteuerung zur Verbesserung des Führungsverhaltens im niedrigen Drehzahlbereich bewährt. Sie ist notwendig, weil der Zusammenhang zwischen Pulsweite  $pw$  und Geschwindigkeit  $v_o$  nicht linear ist. In Bild 17 ist die Hysterese dargestellt, die aufgrund von Reibungseinflüssen das Steuerverhalten des Motors negativ beeinflusst.

Um ein gutes Regelverhalten auch im niedrigen Drehzahlbereich zu erreichen, wird mittels Vorsteuerung die Pulsweite je nach Drehrichtung auf  $\pm pw_{stop}$  gesetzt. Das erlaubt eine schnelle Reaktion des Motors auch bei Drehrichtungsumkehrungen im niedrigen Drehzahlbereich.

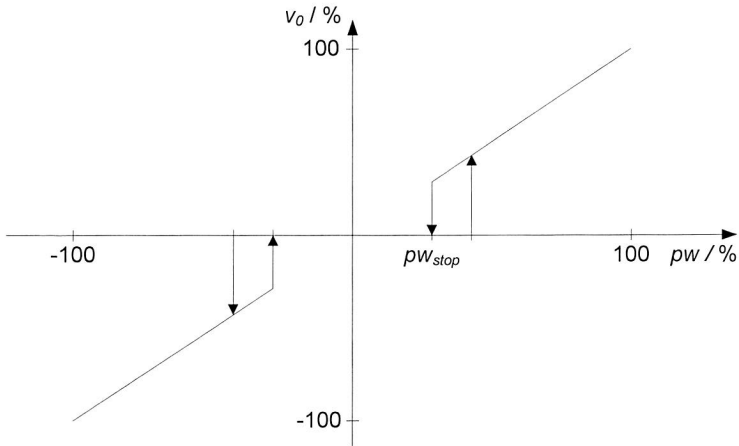


Bild 17: Qualitativer Zusammenhang zwischen Pulsweite (Ankerspannung) und Umfangsgeschwindigkeit des Antriebsrads im lastfreien Fall.

Der zusätzlich vorhandene Regler ist als PI-Regler ausgeführt. Er ist dafür verantwortlich, im stationären Fall die bleibende Regelabweichung auf 0 auszuregulieren. Die zeitdiskrete Übertragungsfunktion eines PI-Reglers ist in der Literatur [5] zu finden.

$$G_{PI}(z) = k_p \cdot \left( 1 + k_i \frac{dt}{1 - z^{-1}} \right) \quad (5-3)$$

Eine schnelle und nicht oszillatorische Ausregelung von Regelabweichungen lässt sich mit den Werten  $k_p = 2$  und  $k_i = 500/\text{s}$  bei einer Abtastzeit von  $dt = 0,005 \text{ s}$  erreichen.

### 5.2.2 Mikrocontroller zur Kommunikation mit dem Steuerungsrechner

In periodischen Abständen liefert der Fahrzeug-PC Sollwerte für Longitudinalgeschwindigkeit des Fahrzeugs und Sollradius der Trajektorie (vgl. 5.2.4) an den zentralen Mikrocontroller. Ebenso werden periodisch alle Sensorwerte (vgl. 5.1.2) mittels Mikrocontroller abgetastet und in digitaler Form an den Fahrzeug-PC übermittelt.

Dem zentralen Mikrocontroller, in Bild 15 mit „PC-Kommunikation“ bezeichnet, kommen somit folgende Aufgaben zu:

- Wandeln analoger Sensorsignale in digitale Werte
- Kommunikation mit den Motortreibern (vgl. 5.2.1) (per I<sup>2</sup>C-Bus werden sowohl die Sollgeschwindigkeiten bereitgestellt als auch die zurückgelegten Strecken  $s_0$  und  $s_1$  empfangen)
- Sollgeschwindigkeiten für Motor 0 und 1 berechnen
- Poseänderung des Fahrzeugs aus  $s_0$  und  $s_1$  errechnen (Koppelnavigation)

- Übertragen der Poseänderung des Fahrzeugs und der Sensorsignale an den Fahrzeug-PC

Diese Punkte werden im Folgenden im Detail erläutert.

### **Wandeln analoger Sensorsignale**

Der entsprechend 5.1 verwendete zentrale Mikrocontroller ATmega32 ist mit einem zehn Bit analog-digital Wandler ausgestattet. Zusätzlich besitzt er einen acht Kanal Multiplexer. Damit ist es möglich, alle acht Kanäle mit einer Samplerate von bis zu 153 kHz abzutasten. Somit kann aus Einfachheitsgründen davon ausgegangen werden, dass alle Sensordaten (vier Infrarot Abstandssensoren und vier Unterbodensensoren) dem Controller permanent zur Verfügung stehen. Obwohl der Wandler eine zehn Bit Auflösung der digitalisierten Sensorwerte erlaubt, werden hier lediglich acht Bit verwendet.

### **Kommunikation mit den Motortreibern**

Die Kommunikation mit den Mikrocontrollern zur Motorregelung basiert auf dem I<sup>2</sup>C-Bus [89]. Dieser erlaubt Datenraten von bis zu 3,4 Mbit/s und benötigt lediglich zwei Leitungen (Clock und Data).

### **Sollgeschwindigkeiten für Motor 0 und 1 berechnen**

Die beiden Motorregler bekommen die Sollwerte für den jeweiligen Motor mit einer Frequenz von 50 Hz übermittelt. Die Werte  $v_0$  und  $v_1$  entsprechen dabei der resultierenden Winkelgeschwindigkeit des jeweiligen Antriebsrades multipliziert mit dessen Durchmesser. Dabei ergeben sich die Sollwerte  $v_{0,soll}$  für den rechten Antrieb und  $v_{1,soll}$  für den linken Antrieb aus der vom Fahrzeug-PC geforderten Longitudinalgeschwindigkeit  $v_{soll}$  und dem zu befahrenden Kurvenradius  $r_{soll}$ . Positive Sollradien entsprechen einer Linkskurve (bei Vorwärtsfahrt), negative Sollradien einer Rechtskurve (bei Vorwärtsfahrt). Bei  $r_{soll} = 0$  rotiert das VTF auf der Stelle um seine Hochachse (mathematisch positiv bei  $v_{soll} > 0$ ).

$$v_{0,soll} = \begin{cases} v_{soll}, & r_{soll} = 0 \\ v_{soll}, & (r_{soll} > r_{\max}) \vee (r_{soll} < -r_{\max}) \\ v_{\max} \cdot \operatorname{sgn}(v_{soll}), & \left( \frac{2 \cdot r_{soll} + d}{2 \cdot r_{soll}} > 1 \right) \wedge \left( \left| v_{soll} \cdot \frac{2 \cdot r_{soll} + d}{2 \cdot r_{soll}} \right| > v_{\max} \right) \\ v_{\max} \cdot \operatorname{sgn}(v_{soll}) \cdot \left( \frac{2 \cdot r_{soll} + d}{2 \cdot r_{soll} - d} \right), & \left( \frac{2 \cdot r_{soll} - d}{2 \cdot r_{soll}} > 1 \right) \wedge \left( \left| v_{soll} \cdot \frac{2 \cdot r_{soll} - d}{2 \cdot r_{soll}} \right| > v_{\max} \right) \\ v_{soll} \cdot \left( 1 + \frac{d}{2 \cdot r_{soll}} \right), & \text{sonst} \end{cases} \quad (5-4)$$

$$v_{1,soll} = \begin{cases} -v_{soll}, & r_{soll} = 0 \\ v_{soll}, & (r_{soll} > r_{max}) \vee (r_{soll} < -r_{max}) \\ v_{max} \cdot \operatorname{sgn}(v_{soll}) \cdot \left( \frac{2 \cdot r_{soll} - d}{2 \cdot r_{soll} + d} \right), & \left( \frac{2 \cdot r_{soll} + d}{2 \cdot r_{soll}} > 1 \right) \wedge \left( \left| v_{soll} \cdot \frac{2 \cdot r_{soll} + d}{2 \cdot r_{soll}} \right| > v_{max} \right) \\ v_{max} \cdot \operatorname{sgn}(v_{soll}), & \left( \frac{2 \cdot r_{soll} - d}{2 \cdot r_{soll}} > 1 \right) \wedge \left( \left| v_{soll} \cdot \frac{2 \cdot r_{soll} - d}{2 \cdot r_{soll}} \right| > v_{max} \right) \\ v_{soll} \cdot \left( 1 - \frac{d}{2 \cdot r_{soll}} \right), & \text{sonst} \end{cases} \quad (5-5)$$

Hierbei wird davon ausgegangen, dass gilt  $-v_{max} < v_{soll} < v_{max}$ . Der Abstand der Antriebsräder (Fahrzeugbreite) entspricht  $d$ . In (5-4) und (5-5) wird zwischen folgenden fünf Fällen unterschieden:

- Wenn  $r_{soll} = 0$ : Reine Rotation um die Hochachse (Drehen auf der Stelle)
- Wenn  $r_{soll} > r_{max}$  und  $r_{soll} < -r_{max}$ : Geradeausfahrt (vorwärts oder rückwärts). Dabei gilt  $r_{max} = 32.000 \text{ mm}$
- Wenn ein Bogen mit positivem Radius (Linkskurve bei Vorwärtsfahrt) gefahren werden soll und gleichzeitig das kurvenäußere Rad ( $m_0$ ) die zulässige Höchstgeschwindigkeit  $v_{max}$  überschreiten würde: Reduziere die Geschwindigkeit  $v_{0,soll}$  auf  $v_{max}$  und stelle über  $v_{1,soll}$  sicher, dass der gefahrene Radius dem Sollradius entspricht. Die Sollgeschwindigkeit  $v_{soll}$  wird in diesem Falle vom VTF nicht erreicht
- Wenn ein Bogen mit negativem Radius (Rechtskurve bei Vorwärtsfahrt) gefahren werden soll und gleichzeitig das kurvenäußere Rad ( $m_1$ ) die zulässige Höchstgeschwindigkeit  $v_{max}$  überschreiten würde: Reduziere die Geschwindigkeit  $v_{1,soll}$  auf  $v_{max}$  und stelle über  $v_{0,soll}$  sicher, dass der gefahrene Radius dem Sollradius entspricht. Die Sollgeschwindigkeit  $v_{soll}$  wird in diesem Falle vom VTF nicht erreicht
- Wenn keiner der vier zuvor genannten Sonderfälle eintritt: Befahre Kreisbogen mit vorgegebenem Radius  $r_{soll}$  und vorgegebener Fahrzeuggeschwindigkeit  $v_{soll}$

Auf die mathematische Herleitung der in (5-4) und (5-5) angegebenen Relationen sei hier verzichtet.

### **Poseänderung des Fahrzeugs aus $s_0$ und $s_1$ errechnen (Koppelnavigation)**

Die Inkrementalencoder beider unter 5.2.1 beschriebenen Antriebsmotoren werden genutzt, um die zurückgelegte Wegstrecke  $s_0$  bzw.  $s_1$  jedes Antriebsrads zur ermitteln. Mit Hilfe dieser Daten, die mit einer Frequenz von 50 Hz bereitgestellt werden, kann auf dem zentralen Mikrocontroller die Poseänderung des Fahrzeugs errechnet werden. Entsprechend Bild 18 lässt sich diese in einem ortsfesten kartesischen  $x''y''$ -Koordinatensystem angeben. Dabei wird davon ausgegangen, dass sich das Fahrzeug während eines Abtastschrittes der Mikrocontroller auf einer Kreisbahn bewegt. Alle Koordinatensysteme, die nicht das Welt-Koordinatensystem darstellen, sind mit einem Apostroph (') gekennzeichnet. Ein Koordinatensystem, welches innerhalb des Weltkoordinatensystems liegt, wird mit  $x'y'$  bezeichnet. Ein Koordinatensystem, das in

einem  $x'y'$ -Koordinatensystem liegt, wird mit  $x''y''$  gekennzeichnet. Mit der Fahrzeugbreite  $d$  beträgt die Änderung der Orientierung

$$\Delta\varphi'' = \frac{s_0 - s_1}{d} \quad (5-6)$$

und die zurückgelegte Wegstrecke des Fahrzeugmittelpunktes

$$s = \frac{s_0 + s_1}{2} \quad (5-7)$$

Der Radius  $r$  des gefahrenen Kreisbogens (vgl. Bild 18) lässt sich aus

$$\Delta\varphi'' = \frac{s}{r} = \frac{s_1}{r - \frac{d}{2}} = \frac{s_0}{r + \frac{d}{2}} \quad (5-8)$$

durch Auflösen ermitteln

$$r = \frac{d}{2} \cdot \frac{s_0 + s_1}{s_0 - s_1} \quad (5-9)$$

Dabei beschreiben positive  $r$  einen Kreisbogen nach links, negative  $r$  einen Kreisbogen nach rechts. Somit lässt sich die Änderung der Position des Fahrzeugs während eines Abtastschrittes der Mikrocontroller mit

$$\Delta x'' = \sin(\Delta\varphi'') \cdot r = \begin{cases} \sin\left(\frac{s_0 - s_1}{d}\right) \cdot \frac{s_0 + s_1}{s_0 - s_1} \cdot \frac{d}{2}, & s_0 \neq s_1 \\ s_0, & s_0 = s_1 \end{cases} \quad (5-10)$$

bzw.

$$\Delta y'' = (1 - \cos(\Delta\varphi'')) \cdot r = \begin{cases} \left(1 - \cos\left(\frac{s_0 - s_1}{d}\right)\right) \cdot \frac{s_0 + s_1}{s_0 - s_1} \cdot \frac{d}{2}, & s_0 \neq s_1 \\ 0, & s_0 = s_1 \end{cases} \quad (5-11)$$

angeben.

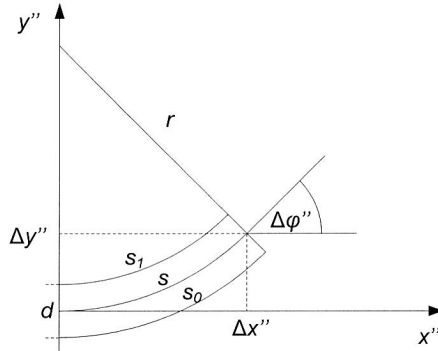


Bild 18: Änderung der Position und Orientierung im ortsfesten kartesischen Koordinatensystem aufgrund einer zurückgelegten Wegstrecke  $s_0$  des rechten Antriebsrads und  $s_1$  des linken Antriebsrads eines VTF.

Die Poseänderung eines Fahrzeugs über einen längeren Zeitraum (länger als  $1/50$  s) wird aus der Summe aller aneinandergehängten Kreisbögen errechnet.

Auf dem zentralen Mikrocontroller ergibt sich aus  $n$  aufeinanderfolgenden Poseänderungen im  $x''y''$ -Koordinatensystem die gesamte Poseänderung im  $x'y'$ -Koordinatensystem mit

$$\Delta\varphi'_n = \sum_{i=1}^n \Delta\varphi''_i \quad (5-12)$$

$$\Delta x'_n = \sum_{i=1}^n (\Delta x''_i \cdot \cos(\Delta\varphi'_{i-1}) - \Delta y''_i \cdot \sin(\Delta\varphi'_{i-1})) \quad (5-13)$$

$$\Delta y'_n = \sum_{i=1}^n (\Delta x''_i \cdot \sin(\Delta\varphi'_{i-1}) + \Delta y''_i \cdot \cos(\Delta\varphi'_{i-1})) \quad (5-14)$$

wobei gilt  $\Delta\varphi'_0 = 0$ .

Nach der Weiterleitung der Poseänderung  $\Delta\varphi'_n$ ,  $\Delta x'_n$  und  $\Delta y'_n$  eines Fahrzeugs an den Fahrzeug-PC wird das  $x'y'$ -Koordinatensystem an die Pose des Fahrzeugs zu diesem Zeitpunkt gelegt und  $\Delta\varphi'_n$ ,  $\Delta x'_n$  und  $\Delta y'_n$  auf 0 gesetzt. Daher bezieht sich die Poseänderung nach (5-12), (5-13) und (5-14) immer auf ein ortsfestes, kartesisches Koordinatensystem, das seinen Ursprung und seine Orientierung an der Pose des Fahrzeugs zum letzten Kommunikationszeitpunkt mit dem Fahrzeug-PC hat.

Zur Vereinfachung wird die Poseänderung des Fahrzeugs im Folgenden mit  $\Delta\varphi'_k$ ,  $\Delta x'_k$  und  $\Delta y'_k$  angegeben. Dabei stellt  $k$  das Zeitinkrement für die Kommunikation zwischen VFT und Weltmodell dar. Somit liefern  $\Delta\varphi'_k$ ,  $\Delta x'_k$  und  $\Delta y'_k$  die Poseänderung des Fahrzeugs zwischen dem Zeitpunkt  $k$  und  $k+1$ . Dieser Zusammenhang ist in Bild 19 dargestellt.



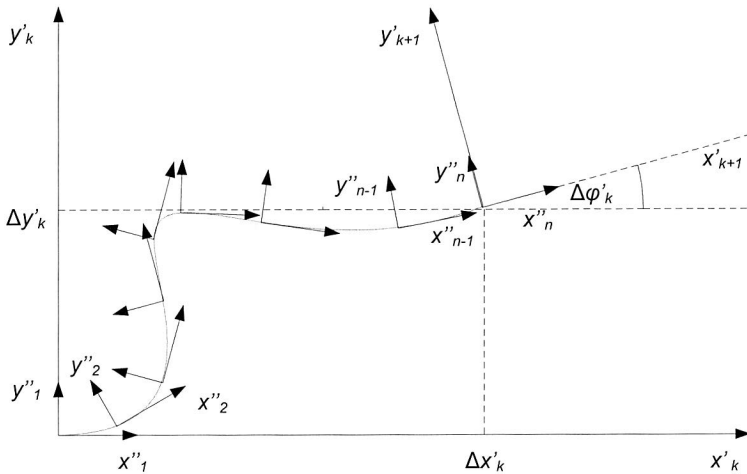


Bild 19: Poseänderung eines VTF über einen Zeitraum von  $n$  Zyklen der Mikrocontroller Kommunikation. Die einzelnen Poseänderungen (Kreispfeile) im  $x''y''$ -Koordinatensystem werden zur Poseänderung im  $x'y'$ -Koordinatensystem aufsummiert.

Die Änderung der Fahrzeugpose wird auf dem Fahrzeug-PC sowohl an den Trajektorienvergleich als auch an das Weltmodell weitergeleitet. Der Trajektorienvergleich (vgl. 5.2.5) summiert ab dem Systemstart die Poseänderungen  $\Delta \varphi'_k$ ,  $\Delta x'_k$  und  $\Delta y'_k$  der letzten  $l$  Kommunikationsschritte auf, sodass die Trajektorie im ersten lokalen Koordinatensystem,  $x'_0y'_0$  (die Pose, die das Fahrzeug bei dessen Systemstart innehatte), dargestellt ist.

Im Weltmodell hingegen dienen die lokalen Poseänderungen zusammen mit den Daten der Deckenkameras als Eingangsdaten für einen Zustandsschätzer. Mit Hilfe eines Kalmanfilters erfolgt eine optimale Schätzung der Pose des VTF (vgl. 6.4.3). Die Zustandsraumdarstellung des Bewegungsmodells eines VTF ist in (6-5) gegeben.

### **Übertragen der Poseänderung des Fahrzeugs und der Sensorsignale an den Fahrzeug-PC**

Die Kommunikation zwischen Fahrzeug-PC und dem Mikrocontroller „PC-Kommunikation“ basiert auf einer seriellen RS232-Verbindung. Da hier die Datenrate auf 38.400 Baud begrenzt ist, resultiert eine Kommunikationsfrequenz von 20 Hz. Pro Zyklus werden vom Fahrzeug-PC die Sollgeschwindigkeit und der Sollradius an den Mikrocontroller gesendet. Als Antwort sendet der Fahrzeug-PC die Positionsänderungen in  $x$ - und  $y$ -Richtung, die Änderung der Orientierung sowie die binären Werte der Unterbodensensoren und die Werte der Abstandssensoren zurück.

### 5.2.3 Pfadplaner

Damit ein VTF autonom ihm zugewiesene Transportaufgaben bewältigen kann, muss es in der Lage sein, den abzufahrenden Pfad zur Zielposition zu planen. Dieses Problem wird direkt im Zusammenhang mit autonomen Transportfahrzeugen [93], aber auch allgemein in einer Vielzahl von Schriften diskutiert. Ein Standardwerk hierfür stellt [62] dar. Das Pfadplanungsproblem lässt sich allgemein folgendermaßen formulieren: „Gegeben sind eine Startposition (die augenblickliche Position des Roboters), eine Zielposition und eine Umgebungskarte. Gesucht ist eine Abfolge von Aktionen, [...], die den Roboter störungsfrei (also z. B. ohne mit Hindernissen zusammen zu stoßen) von der Start- zur Zielposition bewegen.“ [40].

Um das Thema Pfadplanung zu diskutieren, erfolgt zunächst die Definition einiger Begriffe. Diese werden einheitlich in der Literatur verwendet. Inhaltlich sind die Erklärungen an [62] angelehnt, die deutsche Begrifflichkeit wird entsprechend [115] verwendet.

#### **Konfigurationsraum (configuration space):**

Der Konfigurationsraum  $\mathbf{C}$  bezeichnet die Gesamtheit aller möglichen Konfigurationen eines Roboters – wobei hier Roboter generell für das Objekt steht, das den geplanten Pfad abfahren soll. Die Anzahl seiner Dimensionen entspricht der Anzahl der Freiheitsgrade des Roboters. So ist der Konfigurationsraum bei einem sechssachsigen Industrieroboter 6-dimensional, bei einem autonomen Fahrzeug, das sich in der Ebene bewegt, 3-dimensional. Bei einem rotationssymmetrischen Fahrzeug 2-dimensional, da die Orientierung vernachlässigt werden kann.  $\mathbf{C}$  wird in die beiden Bereiche  $\mathbf{C}_{\text{obs}}$  und  $\mathbf{C}_{\text{free}}$  unterteilt, wobei  $\mathbf{C}_{\text{obs}}$  die Menge der Konfigurationen beschreibt, die zu einer Kollision mit einem Hindernis führen.

#### **Aktion (action):**

Eine Aktion ist der Übergang von einer Konfiguration eines Roboters in eine andere Konfiguration. „Jede Aktion  $u$  produziert aus einem Zustand  $x$  entsprechend einer Funktion  $f$  einen neuen Zustand  $x'$ .“ [62]

#### **Vollständigkeit (completeness):**

Ein Algorithmus wird als „vollständig“ bezeichnet, wenn er in endlicher Zeit entweder einen Pfad zum Ziel findet, oder richtigerweise rückmeldet, dass kein Pfad existiert. „Auflösungsvollständig“ („resolution complete“) bedeutet, dass der Algorithmus, wenn eine Lösung existiert, diese in endlicher Zeit finden wird. Wenn keine Lösung existiert, wird der Algorithmus ewig suchen. Die meisten stichprobenbasierten Algorithmen basieren auf Zufallszahlen. Das führt zu Algorithmen, die „probabilistisch vollständig“ („probabilistic complete“) sind. Das bedeutet, dass die Wahrscheinlichkeit, eine existierende Lösung zu finden, mit genügend Punkten gegen 1 strebt.

Pfadplanungsmethoden lassen sich grundsätzlich in verschiedene Kategorien unterteilen. Nach [86] und [40] werden Landkarten (bzw. Wegekarten-), Zellzerlegungs- (bzw. Zellunterteilungs-) und Potentialfeldmethode unterschieden. In [62] wird dagegen

hauptsächlich in die Kategorien kombinatorische Bewegungsplanung (combinatorial motion planning) und stichprobenbasierte Bewegungsplanung (sampling based motion planning) unterteilt. Die Potentialfeldmethode stellt hierbei einen Vorläufer zu stichprobenbasierten Methoden dar. Zusätzlich zu den genannten Algorithmen zur Bewegungsplanung wird das Thema „diskrete Planung“ (discrete planning) diskutiert. Hierin werden insbesondere Lösungen zum Finden einer optimalen Lösung in einem gegebenen diskreten Zustandsraum (Graphen) erläutert. Hier kommen unter anderem A\* [43] oder Dijkstras Algorithmus [28] zur Graphensuche zur Sprache. Zusammenfassend gilt, dass die Methoden zur Bewegungsplanung in den meisten Fällen einen Graphen erzeugen und die diskreten Planungsalgorithmen bzw. Algorithmen zur Graphensuche einen optimalen Pfad in diesem Graphen finden.

Es folgt eine kurze Vorstellung der Grundlagen populärer kombinatorischer Planungsalgorithmen sowie der Potentialfeldmethode. Aufgrund der aufgetretenen Probleme bei der Implementierung dieser Verfahren wurde schließlich ein stichprobenbasierter Ansatz gewählt, der im Anschluss beschrieben wird.

### ***Kombinatorische Algorithmen (combinatorial algorithms)***

Kombinatorische Algorithmen werden auch als „exakte“ Algorithmen bezeichnet, da sie im Konfigurationsraum einen Pfad ohne Approximationen finden. Alle kombinatorischen Algorithmen sind vollständig. Sie basieren meistens auf der Generierung eines Graphen und der anschließenden Suche nach dem kürzesten Weg zum Ziel in diesem Graph. Beispielsweise seien hier entsprechend [62] zwei populäre Vertreter dieser Kategorie kurz beschrieben:

- Sichtbarkeitsgraph (visibility graph oder shortest path roadmap). Hier wird zur Grapherzeugung von den Eckpunkten der Hindernisse ausgegangen. Dabei stellt jede Ecke einen Knoten des Sichtbarkeitsgraphen dar. Alle von einem Knoten aus sichtbaren anderen Knoten (Ecken), werden über gewichtete Kanten miteinander verbunden. Start- und Zielpunkt werden zusätzlich als Knoten in den Graphen aufgenommen. Ein Sichtbarkeitsgraph ist in Bild 20 (a) gezeigt.
- Zell Dekomposition bzw. Zellzerlegung (cell decomposition). Hierfür existieren verschiedene Verfahren (vertikale oder zylindrische Dekomposition...) Charakteristisch ist hier, dass die Ecken der Hindernisse Ausgangspunkte zur Bildung der verschiedenen Zellen bilden. Die Wegpunkte bzw. Knoten des Graphen, in dem später nach dem kürzesten Weg gesucht wird, ergeben sich sowohl aus den Mittelpunkten der gefundenen Zellen als auch aus den Mittelpunkten der (senkrechten) Linien zum Rand bzw. zum nächsten Hindernis (vgl. Bild 20 (b)). Es sei hier angemerkt, dass das Quadtree-Verfahren, welches häufig auch als Zellzerlegung bezeichnet wird, ein stichprobenbasiertes Verfahren darstellt und nicht vollständig ist.

### **Potentialfeldmethode (potential field method)**

Die Potentialfeldmethode basiert auf dem Gedanken, den Hindernissen im näheren Umfeld ein abstoßendes Potential zuzuweisen. Gleichzeitig wird global ein Potentialgefälle von der aktuellen Roboterposition zur Zielposition überlagert. Das so erzeugte Potentialfeld ermöglicht es dem Roboter, an jeder Stelle des Konfigurationsraums die Richtung des maximalen Potentialgefälles zu berechnen. So ergibt sich ein Pfad zum Zielpunkt, wobei Hindernissen im näheren Umfeld ausgewichen wird. Beispielhaft ist ein Potential in Bild 20 (d) dargestellt. In [57] werden jedoch „lokale Minima“, „kein Durchfahren von engen Zwischenräumen zwischen Hindernissen“, „Instabilität in der Gegenwart von Hindernissen“ und „Instabilität der Pfadplanung in engen Passagen“ als Probleme der Potentialfeldmethode genannt, die deren Einsatz bei mobilen Robotern schwierig gestalten. Um die genannten Probleme zu lösen, wurden einige Strategien entwickelt. Ein Ansatz ist die sogenannte „randomized potential field method“. Diese ermöglicht die Überwindung lokaler Minima durch Zufallsbewegungen (random walk). Obwohl diese Methode auch in hochdimensionalen Konfigurationsräumen lokale Minima überwinden kann, werden ihr aufgrund ihrer aufwändigen Parametrierung häufig andere Planungsverfahren vorgezogen.

### **Stichprobenbasierte Algorithmen (sampling based algorithms)**

Stichprobenbasierte Algorithmen bestehen hauptsächlich aus zwei Arbeitsschritten. Zunächst werden Punkte im Konfigurationsraum ausgewählt, die zur Pfadgenerierung in Betracht gezogen werden. Dies kann auf Basis eines fest vorgegeben Rasters, einer anderen vorgegebenen Verteilung oder zufallsbasiert erfolgen. Dann erfolgt die Überprüfung der ausgewählten Punkte auf Kollisionen mit Hindernissen im Konfigurationsraum, sodass schließlich lediglich Punkte im  $C_{\text{frei}}$  erhalten bleiben.

Eine populäre Möglichkeit, stichprobenbasierte Verfahren zu realisieren, stellen gitterbasierte (grid based) Ansätze dar. Hierfür wird ein Gitter über  $C$  bzw.  $C_{\text{frei}}$  definiert, wobei jeder Gitterpunkt eine realisierbare Pose des Roboters darstellt. Gitterbasierte Ansätze sind im Allgemeinen nicht vollständig, sondern lediglich auflösungsvollständig. Jedoch gibt es spezielle Verfahren, um die Gitterauflösung an Engstellen gezielt zu erhöhen. Das Quadtree-Verfahren [88], in [115] und in [86] als Zell-Dekomposition bzw. näherungsweise Zellzerlegung bezeichnet, stellt eine solche Lösung dar. In Bild 20 (c) ist das Quadtree-Verfahren dargestellt.

Eine gitterbasierte Lösung mit fester Auflösung ist in [125] beschrieben und wird als Zellenmethode bezeichnet. Die Unterteilung des Operationsgebiets des Roboters erfolgt hierbei in Zellen gleicher Größe, wobei jede Zelle einen individuellen Kosten-Wert zugewiesen bekommen kann. Dies ermöglicht es, Bereiche zu definieren, in denen der Roboter bevorzugt navigieren soll.

Statt mittels eines festen Gitters können die Stützpunkte des Pfades auch zufallsbasiert erzeugt werden. Algorithmen dieser Kategorie sind probabilistisch vollständig. Eine populäre Methode dieser Kategorie sind Rapidly exploring Random Trees (RRT) [61]. Die Erzeugung des Suchbaums erfolgt hierbei zufällig, wobei sich seine Auflösung über die Laufzeit erhöht.

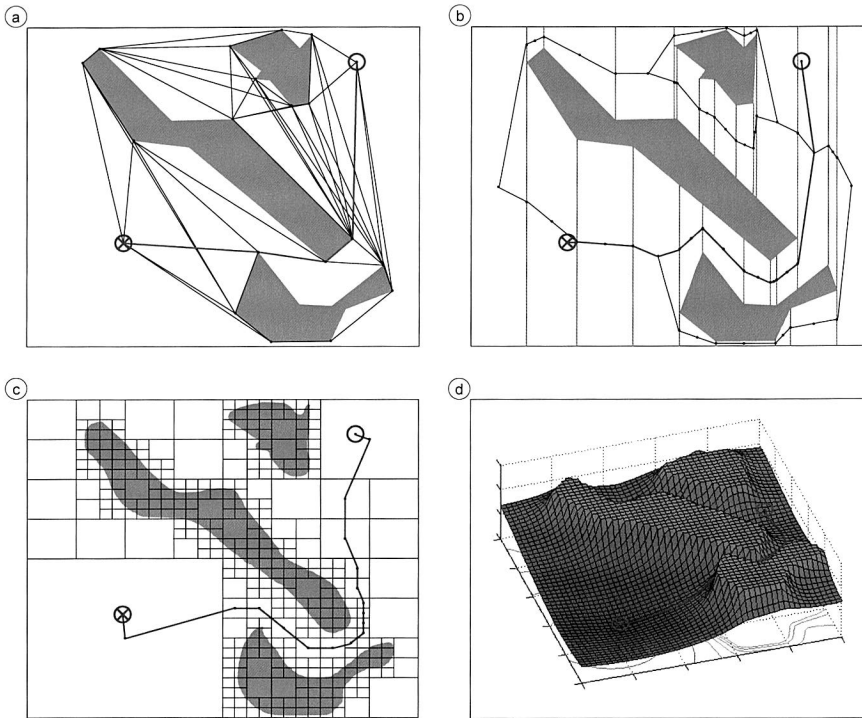


Bild 20: Verschiedene Methoden zur Pfadplanung. Sichtbarkeitsgraph (a) und Zellzerlegung (b) (kombinatorisch), Quadtree (c) (stichprobenbasiert) und Potentialfeldmethode (d)

Um eine für das VTF geeignete Pfadplanungsmethode zu finden, wurden einige der vorgestellten Algorithmen implementiert, sowie Implementierungen bei anderen Systemen untersucht. Da kombinatorische Algorithmen hauptsächlich für 2D-Planungsaufgaben eingesetzt werden und hierbei häufig gute Ergebnisse erreicht werden können, wurde zunächst ein Planer mittels Sichtbarkeitsgraph-Erzeugung und A\* zur Graphensuche implementiert. Hierbei hat es sich als problematisch erwiesen, dass die Erzeugung des Sichtbarkeitsgraphen [12] sowohl implementierungstechnisch aufwändig als auch rechenzeitintensiv ist. Da die zur Verfügung stehende Rechenleistung auf dem Fahrzeug-PC mit einem AMD Geode LX 800 Prozessor und 1GB DDR-RAM begrenzt ist, beträgt die Pfadplanungszeit – je nach Anzahl vorhandener Hindernisse und anderer Fahrzeuge und damit Größe des Sichtbarkeitsgraphen – ca. 2 s. Dabei ist zu bedenken, dass die gesamte Implementierung mittels der Scriptsprache Python erfolgt, was gegenüber kompilierten Sprachen wie C oder C++ erhebliche Performanceeinbußen mit sich bringt (vgl. 7.3.2).

Auch die Potentialfeldmethode wurde im Rahmen der Entwicklung des VTF implementiert. Die zu erwartenden Vorteile einer schnellen Wegplanung ließen sich damit

erreichen. Andererseits hat sich gezeigt, dass lokale Minima nicht zuverlässig überwunden werden konnten, sodass eine weitere Optimierung im Rahmen von Simulationen nicht sinnvoll erschien.

Der momentan verwendete Pfadplanungsalgorithmus basiert auf einem RRT. Hierfür wird die bereits fertig verfügbare Pfadplanungsbibliothek OMPL [98] – die auch in ROS Verwendung findet – erfolgreich eingesetzt. Obwohl der verwendete Pfadplaner nur probabilistisch vollständig ist, sind beim Einsatz im VTF gute Ergebnisse mit geringer Laufzeit realisiert worden. In Bild 21 ist das Ergebnis der Wegplanung mittels RRT für das VTF dargestellt. Aufgrund des zufallsbasierten Ansatzes lässt sich ein optimaler Pfad (mit minimaler Distanz zum Ziel) nur bei einer Laufzeit von  $t \rightarrow \infty$  finden. Rot eingezeichnet ist das Ergebnis der stichprobenbasierten Pfadplanung. Das Resultat der nachfolgenden Glättung – und damit der geplante Pfad – ist grün dargestellt.

Während für reine 2D-Pfadplanungsaufgaben häufig kombinatorische Algorithmen Verwendung finden, sind hier in Zukunft stichprobenbasierte Planer erfolgversprechend. Denn die Bewegungen dynamischer Hindernisse und anderer VTF sollen in die Pfadplanung mit einbezogen und somit Geschwindigkeiten, Wartepositionen und Wartezeiten auf Basis geschätzter Hindernistrajektorien errechnet werden. Auf diese Weise kann auch die Bewegungsplanung für den PR2 von Willow Garage realisiert werden [72]. Ebenso werden bei der RoboCup Small Size League [101] hochoptimierte RRT zur Pfadplanung für die Fußballroboter verwendet. Dabei werden Planungszeiten von 1-2 ms erreicht [51].

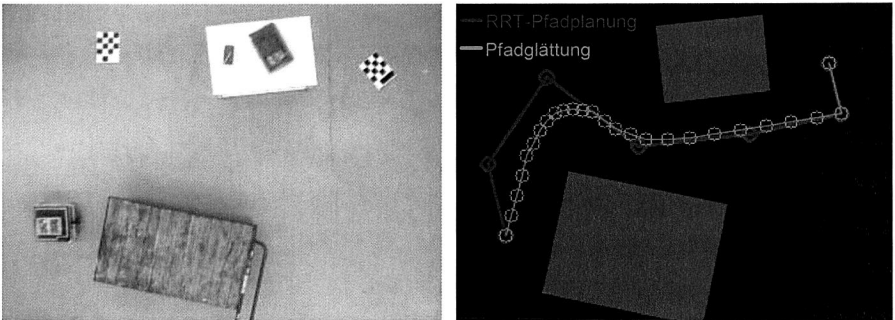


Bild 21: Der verwendete stichprobenbasierte RRT-Pfadplanungsalgorithmus. Links ist das Bild der Deckenkamera gezeigt, rechts das Resultat der Pfadplanung.

Der beschriebene Ansatz zur Pfadplanung ist im sogenannten Pfadplaner umgesetzt. Entsprechend Bild 15 werden ihm sowohl die Weltmodellldaten als auch die mittels Benutzerinterface eingegebene Zielmarker-ID (Identifikationsnummer) mitgeteilt. Der Pfadplaner errechnet dann alle Wegpunkte, die abzufahren sind, um zur Zielposition zu gelangen.

Eine detaillierte Beschreibung der Weltmodellldaten folgt in 6.1.

### 5.2.4 Tracker

In Bild 15 ist zu sehen, dass der geplante Pfad und die Pose des VTF dem Tracker zugeführt werden. Diese Softwareinstanz hat die Aufgabe, geeignete Fahrbefehle an die Antriebsmotoren zu senden, um dem geplanten Pfad mit möglichst geringen Abweichungen zu folgen. Dabei werden an den Mikrocontroller „PC-Kommunikation“ die Longitudinalgeschwindigkeit  $v_{soll}$  und der Radius des zu fahrenden Kreisbogens  $r_{soll}$  übermittelt.

Für 2D-path-tracking-Aufgaben existieren viele Ansätze. Der auf dem VTF implementierte Algorithmus [55] stellt eine Folgeregelung für nicht holonome Fahrzeuge (beweglich lediglich in fahrzeugfester  $x$ -Richtung und rotatorisch um die Hochachse) im 2D-Koordinatensystem dar. Hierin werden die Sollgeschwindigkeiten des Fahrzeugs in Längsrichtung  $v_{soll}$  und die Solldrehgeschwindigkeit um die Hochachse  $\omega_{soll}$  als Funktion der Referenzgeschwindigkeiten  $v_r$  und  $\omega_r$  und der Fehler  $x_e$ ,  $y_e$  und  $\varphi_e$  ermittelt. Dabei ist in (5-15) die Abhängigkeit der Größen  $v_{soll}$ ,  $\omega_{soll}$ ,  $v_r$ ,  $\omega_r$ ,  $x_e$ ,  $y_e$  und  $\varphi_e$  von der Zeit aus Einfachheitsgründen nicht dargestellt.

$$\begin{bmatrix} v_{soll} \\ \omega_{soll} \end{bmatrix} = \begin{bmatrix} v_r \cos(\varphi_e) + k_x x_e \\ \omega_r + v_r (k_y y_e + k_\varphi \sin(\varphi_e)) \end{bmatrix} \quad (5-15)$$

Hierbei stellen die Referenzgrößen die Geschwindigkeit (und damit die Pose) dar, welche das Fahrzeug einnimmt, wenn es dem vorgegebenen Pfad exakt folgt. Die Fehlergrößen beschreiben dagegen die Differenz zwischen Referenzgröße und aktueller, realer Pose des Fahrzeugs im fahrzeugfesten Koordinatensystem, wobei  $x_e$  in Fahrzeug-Längsrichtung und  $y_e$  in Fahrzeug-Querrichtung orientiert sind. Die positiven Konstanten  $k_x$ ,  $k_y$  und  $k_\varphi$  sind so zu wählen, dass eine ausreichend schnelle, nicht oszillatorische und stabile Ausregelung von Pfadabweichungen erfolgt. Mit den Werten  $k_x = 5$ ,  $k_y = 0,0001$  und  $k_\varphi = 0,02$  lässt sich dieses Ziel erreichen, sodass das Fahrzeug den in Bild 22 rechts gezeigten Pfad befährt, wenn der Pfad entsprechend Bild 21 geplant worden ist. Die Anzahl der geplanten Wegpunkte pro Wegstrecke (Auflösung) ist einstellbar im Pfadplaner. Dagegen hängt die Auflösung der Darstellung der real gefahrenen Trajektorie von der Fahrzeuggeschwindigkeit ab, weil die Fahrzeugpose im Weltmodell mit einer festen Frequenz (typisch 5 Hz) errechnet wird.

Da die Eingangsgrößen des Mikrocontrollers „PC-Kommunikation“ aus Sollradius und Sollgeschwindigkeit für den aktuell zu befahrenden Kreisbogen bestehen (vgl. 5.2.2), muss die aus (5-15) resultierende Rotationsgeschwindigkeit um die Hochachse  $\omega_{soll}$  in einen Sollradius umgerechnet werden

$$r_{soll} = \frac{v_{soll}}{\omega_{soll}}. \quad (5-16)$$

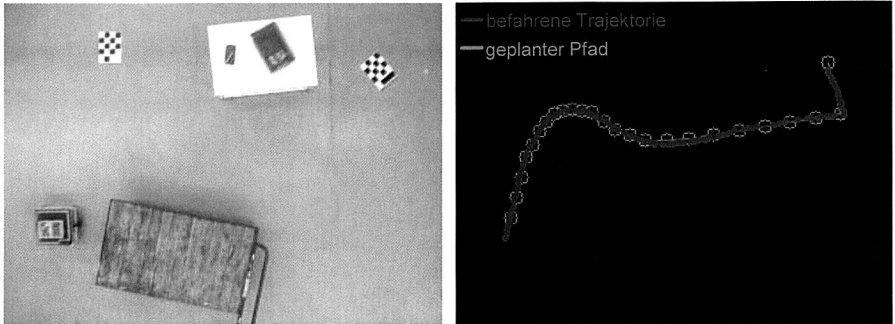


Bild 22: Ausgangssituation aufgenommen durch Deckenkamera (links) und geplanter Pfad mit befahrener Trajektorie des VTF beim Abarbeiten einer Transportaufgabe (rechts).

Um einen Zielmarker anzufahren, navigiert ein VTF mittels Pfadplanung zunächst bis zu einem Punkt vor dem Zielmarker. Beim anschließenden Andockvorgang erfolgt die Positionsbestimmung nicht mehr auf Basis der Weltmodelldaten, sondern relativ zum Zielmarker selbst (die Schätzung der eigenen Pose kommt vom multimodalen Positionsschätzer (vgl. Bild 15)). Bei diesem Andockvorgang wird die Zielposition angesteuert, ohne dass eine explizite Pfadplanung erfolgt. Während der multimodale Positionsschätzer die Poseschätzung des VTF über dem Zielpunktmarker dauernd aktualisiert, besteht der geplante Pfad beim Andockvorgang lediglich aus zwei Punkten auf der Mittellinie des Zielmarkers. Diese sind in Bild 23 hellblau dargestellt.

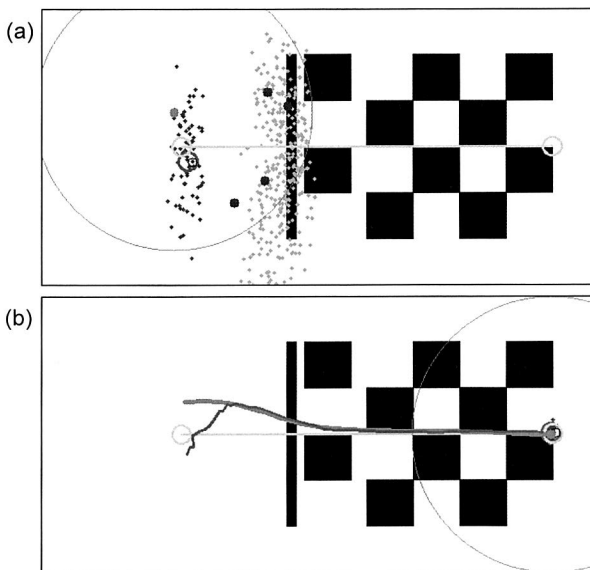


Bild 23: Simulation der vom VTF abgefahrenen Trajektorie beim Positionieren relativ zu einem Zielmarker (Andockvorgang).



In Bild 23 (a) ist die Ausgangssituation eines simulierten Andockvorgangs gezeigt. Die markierten Partikel (dunkelblaue Punkte) inklusive zugehöriger geschätzter Sensorpositionen (gelbe Punkte) dienen zur Zustandsschätzung. Sie werden ebenso wie die realen Positionen der VTF-Sensoren (dunkelrote Punkte) in 5.2.6 detailliert erläutert. Zu sehen sind auch die reale Fahrzeugposition (hellgrüner Punkt) und die geschätzte Fahrzeugposition (brauner Kreis). Bild 23 (b) zeigt die Verläufe der realen (hellgrün) und der geschätzten (braun) Position des VTF. Dabei erfolgt das Tracking auf Basis der geschätzten Position des VTF. Eine zügige Annäherung an die Mittellinie des Zielmarkers erfolgt daher erst, nachdem die Positionsschätzung der realen Position des VTF nahe kommt.

### 5.2.5 Ermittlung der Fahrzeugposition im Weltmodell

Aufgrund der in 5.2.3 beschriebenen Softwarekomponenten ist jedes Fahrzeug in der Lage, selbstständig einen Pfad zur Zielkoordinate zu planen und abzufahren. Eine notwendige Bedingung hierfür ist eine Karte, in der alle Hindernisse (Hindernisobjekte und Fahrzeugobjekte) sowie die Zielposition bzw. Zielpose (Zielobjekt) hinterlegt sind. Weiterhin muss jedem VTF seine eigene Position bekannt sein, welche den Startpunkt des zu planenden Pfades bildet.

Problematisch hierbei ist, dass die Fahrzeuge nicht, wie Zielmarker, optisch markiert sind (entsprechend Forderung aus 4.2). Damit lässt sich im Weltmodell zunächst nicht zwischen einem Hindernis und einem Fahrzeug unterscheiden, weshalb jedes Fahrzeug als Hindernisobjekt entsprechend 6.1.2 modelliert wird. Aus diesem Grund müssen die Fahrzeuge selbstständig herausfinden, wo sie sich befinden, bzw. durch welches Hindernisobjekt sie repräsentiert werden.

Zur Lokalisierung von Objekten innerhalb von Gebäuden existieren verschiedene Lösungen. Häufig besitzen die Häuser stationäre Sender, welche Signale emittieren. Die Positionen der Sender sind bekannt, sodass mit Hilfe eines mobilen Empfangsgeräts mittels Trilateration eine eindeutige Ortsbestimmung möglich ist. Varianten dieser Lokalisierungsmethode nutzen Ultraschall, Infrarot, oder elektromagnetische Wellen im GHz Bereich (Ultra Wide Band (UWB) oder GPS).

In [95] wird ein Ultraschall-basiertes System unter realitätsnahen Bedingungen zur Positionsbestimmung in einer Fertigungsumgebung untersucht. Es besteht aus 9 Ultraschallsendern, die in einem Abstand von 3 m in x-Richtung und 1,5 m in y-Richtung an der Hallendecke angebracht sind. Es ergeben sich Genauigkeiten der Positionsbestimmung von bis zu 0,1 m. Dabei bezeichnet die Genauigkeit die euklidische Distanz zwischen Referenzpunkt und dem Mittelwert der gemessenen Position.

Zum Vergleich erfolgt im gleichen Szenario die Positionsbestimmung mit Hilfe eines UWB-basierten Lokalisierungssystems. Im 2,4 GHz Band erfolgt hier ein bidirektionaler Datenaustausch parallel zur Positionsbestimmung. Die mobilen Sender emittieren hierzu kurze Impulse, sodass mit Hilfe der stationären Empfänger die Position ermittelt werden kann. Die erreichbare Genauigkeit der Positionsermittlung wird mit 0,4 m angegeben.

Auch GPS kann zur Positionsbestimmung innerhalb von Gebäuden verwendet werden. Da die Signale der GPS-Satelliten im Allgemeinen nicht in Gebäuden empfangen werden können, ist es notwendig, Pseudosatelliten innerhalb des Gebäudes zu installieren. Diese Pseudosatelliten emulieren die realen GPS-Satelliten. Die hiermit erreichbare Genauigkeit der Positionsbestimmung liegt laut [69] bei bis zu 1 cm.

Alternativ zu aktiven Systemen lässt sich mittels radio-frequency identification (RFID) die Position eines passiven Transponders ermitteln. Hierbei sendet ein Scanner elektromagnetische Wellen im GHz Bereich aus. Durch elektromagnetische Induktion wird der Transponder mit Energie versorgt, um seine ID dem Scanner zurückzumelden. Auf diese Weise benötigen die Transponder keine eigene Energieversorgung, was sie sehr kostengünstig in Anschaffung und Wartung macht. Zur Positionsbestimmung lassen sich RFID-Transponder, wie in 2.3 beschrieben, zur Lagepeilung mittels Navigationsraster verwenden. Dabei liegt die Genauigkeit der Positionsbestimmung bei ca. 0,1 m [69].

Tabelle 11: Technologien zur Indoor-Positionsbestimmung.

	Genauigkeit / m	Installationsart
RFID	0,1	Tags im Boden
US	0,1	Sender an Wänden/Decke
UWB	0,4	UWB Empfänger/Sender an Wänden/Decke
Indoor GPS	0,01	Pseudo Satelliten an Decke

Mit den beschriebenen und in Tabelle 11 zusammengefasst dargestellten Lösungen zur Indoor-Positionsbestimmung ließe sich die Absolutposition von VTF ermitteln. Es ist jedoch unumgänglich, hierfür weitere Hardware sowohl am VTF als auch in der Fertigungsumgebung zu installieren. Ziel ist es, mit den vorhandenen Deckenkameras und den Sensoren der VTS die Position der optisch nicht markierten VTF zu ermitteln.

Im vorliegenden Kapitel wird deshalb ein Verfahren vorgestellt, mit dem ein VTF seine Absolutposition mittels Trajektorienvergleich ermitteln kann. Hierzu bewegt sich ein VTF auf einer charakteristischen Trajektorie und vergleicht permanent seine eigene Trajektorie mit denen aller Hindernisobjekte. Falls die Trajektorien ähnlich sind, meldet das Fahrzeug selbstständig an das Weltmodell, welches Hindernisobjekt durch ein Fahrzeugobjekt zu ersetzen ist. In Bild 15 ist dieses Modul, als „Trajektorienvergleichler“ bezeichnet, im Datenfluss des Fahrzeug-PC dargestellt. Der grundsätzliche Ablauf ist mittels vereinheitlichter Modellierungssprache (unified modeling language, UML) in einem UML-Aktivitätsdiagramm in Bild 24 dargestellt.

Um Trajektorien – oder allgemein Formen – zu vergleichen, existieren in der Literatur verschiedene Ansätze. Diese bestehen im Allgemeinen aus einer Transformation einer der beiden Trajektorien und dem Errechnen eines Ähnlichkeitsmaßes [118]. Die Transformation einer Trajektorie, sodass sie bestmöglich einer anderen entspricht, stellt ein mathematisches Optimierungsproblem dar. Hierbei werden Verschiebung, Verdrehung und Skalierung errechnet [45].

Für zwei gegebene Trajektorien können verschiedene Ähnlichkeitsmaße wie z. B. die Fréchet-Distanz [82] angegeben werden. Hierbei muss die (zeitliche) Reihenfolge der abzufahrenden Punkte einer Trajektorie gegeben sein, jedoch nicht der Zeitpunkt, zu dem ein Wegpunkt erreicht wird. Die Ähnlichkeit beider Trajektorien wird ermittelt, wobei

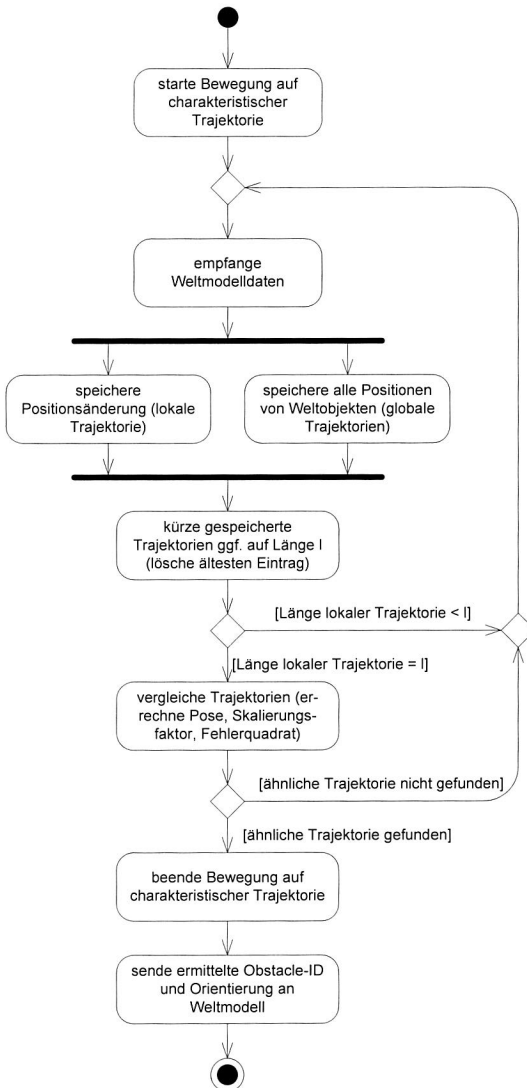


Bild 24: UML-Aktivitätsdiagramm zur Ermittlung der globalen Position eines VTF.

die Geschwindigkeit, mit der die Trajektorien abgefahren werden, im Rahmen der Ähnlichkeitsberechnung optimiert wird. Neben der Fréchet-Distanz existieren weitere Ähnlichkeitsmaße wie die Hausdorff-Distanz für Punktemengen unterschiedlicher Größe oder die Bottleneck-Distanz für Punktemengen gleicher Größe [118].

Im vorliegenden Falle zweier mit der gleichen Anzahl an Wegpunkten gegebener Trajektorien – wobei jeder Wegpunkt einem festen Zeitpunkt und damit seinem Pendant in der anderen Trajektorie zugeordnet ist – lässt sich als Ähnlichkeitsmaß die Summe der Euklidischen Abstände aller Wegpunktpaare angeben.

Zur Errechnung der notwendigen Transformationsparameter (Verschiebung, Rotation und Skalierungsfaktor) eignet sich die Methode der kleinsten Quadrate. Mit Hilfe der im Folgenden angewandten Pseudoinversen [3], lässt sich für ein überbestimmtes Gleichungssystem der Form  $\mathbf{Ax} = \mathbf{b}$  die Lösung im Sinne kleinster Fehlerquadrate (minimale euklidische Norm  $\|\mathbf{Ax} - \mathbf{b}\|_2$ ) ermitteln, also die Lösung, die im Sinne des o. g. Ähnlichkeitsmaßes optimal ist.

Um die gesuchten Transformationsparameter zur Verschiebung  $x^t, y^t$ , zur Verdrehung  $\varphi^t$  und zur Skalierung  $s^t$  zu ermitteln, wird zunächst die Transformationsvorschrift vom  $x'_0 y'_0$ -Koordinatensystem des Fahrzeugs (vgl. 5.2.2) in das globale  $xy$ -Koordinatensystem betrachtet. Ein Punkt im Fahrzeugkoordinatensystem mit  $(p'_x, p'_y)$  lässt sich entsprechend (5-17) in einen Punkt im Weltkoordinatensystem mit  $(p_x, p_y)$  umrechnen.

$$\begin{bmatrix} p_x \\ p_y \end{bmatrix} = \begin{bmatrix} \cos(\varphi^t) & -\sin(\varphi^t) \\ \sin(\varphi^t) & \cos(\varphi^t) \end{bmatrix} \begin{bmatrix} p'_x \\ p'_y \end{bmatrix} s^t + \begin{bmatrix} x^t \\ y^t \end{bmatrix} \quad (5-17)$$

Dieser Zusammenhang ist in Bild 25 dargestellt. Das dort eingezeichnete Rechteck visualisiert insbesondere den Einfluss des Skalierungsfaktors  $s^t$ .

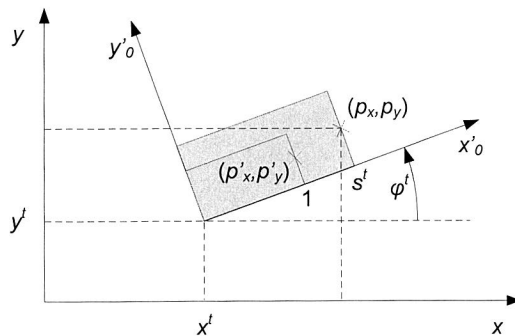


Bild 25: Mit Hilfe der Parameter  $x^t, y^t$  zur Verschiebung,  $\varphi^t$  zur Rotation und  $s^t$  zur Skalierung, lassen sich Punkte vom  $xy$ -Koordinatensystem in das  $x'_0 y'_0$ -Koordinatensystem umrechnen. Beispielhaft ist ein Punktepaar  $(p'_x, p'_y), (p_x, p_y)$  eingezeichnet.

Ziel ist es nun, die gesuchten Transformationsparameter zu finden. Daher wird (5-17) mit

$$a^t = \cos(\varphi^t) \cdot s^t \quad (5-18)$$

$$b^t = \sin(\varphi^t) \cdot s^t \quad (5-19)$$

umgeformt zu

$$\begin{bmatrix} p_x \\ p_y \end{bmatrix} = \begin{bmatrix} a^t & -b^t \\ b^t & a^t \end{bmatrix} \begin{bmatrix} p'_x \\ p'_y \end{bmatrix} + \begin{bmatrix} x^t \\ y^t \end{bmatrix} = \begin{bmatrix} p'_x & -p'_y & 1 & 0 \\ p'_y & p'_x & 0 & 1 \end{bmatrix} \begin{bmatrix} a^t \\ b^t \\ x^t \\ y^t \end{bmatrix} \quad (5-20)$$

In (5-20) stehen lediglich zwei Gleichungen zum Ermitteln von vier Transformationsparametern zur Verfügung. Zusätzliche Gleichungen lassen sich durch die Betrachtung weiterer Punktpaare erstellen. Hierzu werden die skalaren Größen  $p'_x$ ,  $p'_y$ ,  $p_x$  und  $p_y$  zu Spaltenvektoren mit  $q$  Werten erweitert. Ebenso wird 0 durch einen  $q \times 1$  Vektor  $\mathbf{n}$  und 1 durch einen  $q \times 1$  Vektor  $\mathbf{e}$  ersetzt, wobei jedes Element 0 bzw. 1 beträgt. Somit stehen  $2q$  Gleichungen zur Verfügung, um die vier gesuchten Parameter auszurechnen.

$$\begin{bmatrix} p_x \\ p_y \end{bmatrix} = \begin{bmatrix} p'_x & -p'_y & \mathbf{e} & \mathbf{n} \\ p'_y & p'_x & \mathbf{n} & \mathbf{e} \end{bmatrix} \begin{bmatrix} a^t \\ b^t \\ x^t \\ y^t \end{bmatrix} \quad (5-21)$$

Wenn mehr als  $q = 2$  Punktpaare in (5-21) eingesetzt werden, ist das Gleichungssystem überbestimmt. Somit kann keine exakte Lösung mehr gefunden werden. Formal lässt sich eine optimale Lösung jedoch mit Hilfe der Pseudoinversen [3], wie oben erläutert, angeben.

$$\text{pinv} \left( \begin{bmatrix} p'_x & -p'_y & \mathbf{e} & \mathbf{n} \\ p'_y & p'_x & \mathbf{n} & \mathbf{e} \end{bmatrix} \right) \begin{bmatrix} p_x \\ p_y \end{bmatrix} = \begin{bmatrix} a^t \\ b^t \\ x^t \\ y^t \end{bmatrix} \quad (5-22)$$

Die in (5-22) beschriebene Rechenoperation muss jedes Fahrzeug für die Trajektorie jedes Hindernisobjekts ausführen. Die euklidische Norm

$$r = \left\| \begin{bmatrix} \mathbf{p}'_x & -\mathbf{p}'_y & \mathbf{e} & \mathbf{n} \\ \mathbf{p}'_y & \mathbf{p}'_x & \mathbf{n} & \mathbf{e} \end{bmatrix} \begin{bmatrix} a^t \\ b^t \\ x^t \\ y^t \end{bmatrix} - \begin{bmatrix} \mathbf{p}_x \\ \mathbf{p}_y \end{bmatrix} \right\|_2 \quad (5-23)$$

ist als Wurzel der Summe der Quadrate der Vektorkomponenten definiert. Wenn  $r$  einen vorgegebenen Grenzwert unterschreitet und sich gleichzeitig der Skalierungsfaktor  $s^t$  in einem vorher definierten Bereich um 1 bewegt, wird davon ausgegangen, dass das betrachtete Hindernisobjekt das VTF repräsentiert. Dem Weltmodell wird daraufhin die ID des Hindernisobjekts mitgeteilt, welches es durch ein neues Fahrzeugobjekt an der gleichen Position ersetzt. Die Orientierung  $\varphi^t$  und der Skalierungsfaktor  $s^t$  müssen hierfür durch Auflösen von (5-18) und (5-19) ermittelt werden.

$$s^t = \sqrt{(a^t)^2 + (b^t)^2} \quad (5-24)$$

$$\varphi^t = \arccos\left(\frac{a^t}{s^t}\right) \quad (5-25)$$

In Bild 26 (b) ist beispielhaft das Ergebnis eines Vergleichs zweier Hindernisobjekt-trajektorien mit einer Fahrzeugtrajektorie (Bild 26 (a)) dargestellt.

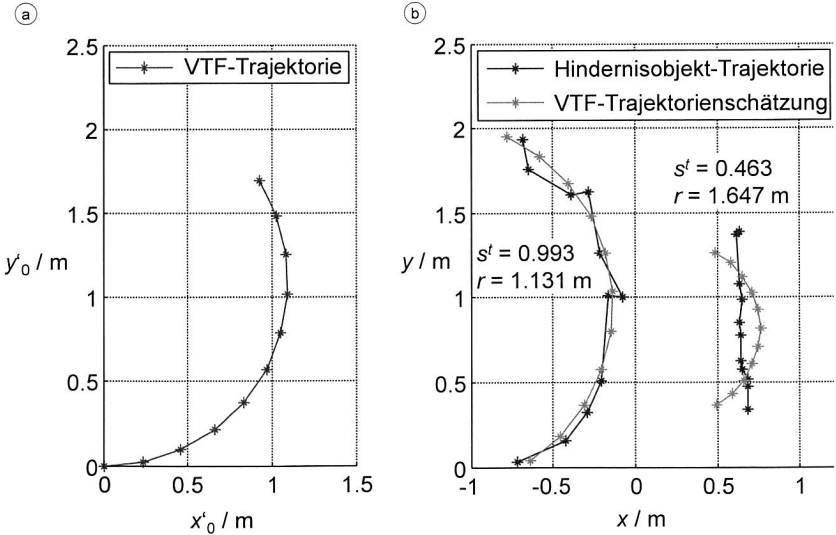


Bild 26: Die im  $x'_0y'_0$ -Koordinatensystem aufgenommene VTF Trajektorie (a) wird mit zwei Hindernisobjekt-Trajektorien im  $xy$ -Koordinatensystem verglichen (b).

Da der Skalierungsfaktor  $s^l$  nahe bei 1 liegt und  $r$  ausreichend niedrig ist, wird davon ausgegangen, dass die linke Hindernisstrajektorie der des VTF entspricht. Im Weltmodell erfolgt daraufhin die Substitution des betreffenden Hindernisobjekts durch ein Fahrzeugobjekt (vgl. 6.1).

### 5.2.6 Erhöhung der Navigationsgenauigkeit bei Andockvorgängen

Sobald ein VTF seine Position im Weltmodell einmal ermittelt hat (vgl. 5.2.5), bekommt es periodisch ein Update seiner absoluten Position vom Weltmodell mitgeteilt. Hierzu werden die Daten der Deckenkameras mit denen der Inkrementalencoder der Fahrtriebe mit Hilfe eines Kalmanfilters fusioniert (vgl. 6.4). Die Genauigkeit dieser Absolutpositionsschätzung liegt im Bereich einiger cm (vgl. Tabelle 20). Während diese Genauigkeit zur Kollisionsvermeidung im Fahrbetrieb ausreichend ist, wird bei Andockvorgängen eine wesentlich präzisere Positionsbestimmung benötigt. Um dies zu erreichen, erfolgt die Pfadplanung (vgl. 5.2.3) zu einem Zielmarker mittels Navigation zu einem Punkt in einem vorher definierten Abstand zur Orientierungslinie (z. B. im Abstand einer halben Fahrzeuglänge). Dieser Punkt lässt sich aufgrund o. g. Gründe nur verhältnismäßig ungenau anfahren. Im anschließenden Andockvorgang orientiert sich das Fahrzeug hingegen an den Farbkanten des Zielmarkers. Dies erlaubt eine sehr genaue Positionsbestimmung mit einer Varianz von unter  $2 \text{ mm}^2$  (vgl. Tabelle 14).

In diesem Kapitel wird ein Verfahren vorgestellt, das es einem VTF ermöglicht, seine Position relativ zu einem Zielpunktmarker zu bestimmen. Hierzu werden die Kanten der Orientierungslinie, des ID-Felds und des Schachbrettmusters eines Zielmarkers mittels optischer Sensoren erfasst. In Bild 27 ist ein Zielmarker mit der Zielposition (5) dargestellt. Aufgrund der beschriebenen Ungenauigkeiten in der Positionsbestimmung ist zu erwarten, dass sich das Fahrzeug (sein Mittelpunkt) zu Beginn eines Andockvorgangs im Startbereich (1) befindet.

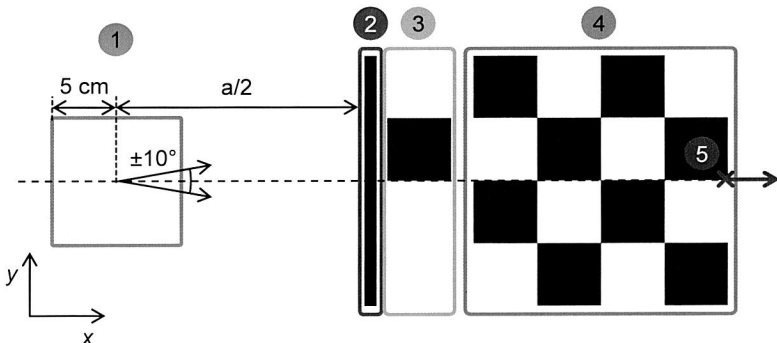


Bild 27: Optischer Marker zur Kennzeichnung einer Zielposition. Gekennzeichnet sind der Startbereich (1), die Orientierungslinie (2), das ID-Feld (3), das 4x4 Schachbrettmuster (4) und die anzufahrende Zielposition (5). Mit  $a$  wird die Länge des Fahrzeugs bezeichnet. Die horizontale, gestrichelte Linie wird als Markermittellinie bezeichnet.

Ausgehend von einer Standardabweichung von  $\pm 5$  cm in  $x$ - und  $y$ -Richtung bedeutet das, dass ca. 47 % aller Andockvorgänge in dem dargestellten Bereich beginnen. Die Orientierung relativ zur Normalen der Orientierungslinie wird mit einer Standardabweichung von  $\pm 10^\circ$  angenommen.

Wie in 5.1.1 beschrieben, werden optische Sensoren [119] zur Detektion der Farbe des Untergrunds eingesetzt. Diese Sensoren liefern jeweils einen Grauwert für eine idealisiert punktförmige Fläche des Untergrunds.

Mit Hilfe der zusätzlich vorhandenen Inkrementalencoder der Antriebseinheiten soll jetzt eine optimale Schätzung der Fahrzeugposition relativ zum Zielpunktmarker ermittelt werden. Dies passiert in dem hierfür entwickelten Modul „Multimodaler Positionsschätzer“, das, wie in Bild 15 dargestellt, in den Datenfluss des Fahrzeug-PC eingebunden ist.

Beim Überfahren des Zielmarkers liefern die Unterbodensensoren aufgrund des Schachbrettmusters ein binäres Signal. Eine exakte Positionsbestimmung ist also nicht an jeder beliebigen Stelle des Musters möglich. Vielmehr muss während des Überfahrens des Musters und bei der Detektion von Feldkanten (Übergänge zwischen Schwarz und Weiß) die Positionsschätzung immer weiter verbessert werden. Dabei kann es vorkommen, dass aufgrund der Symmetrie des Schachbrettmusters mehrdeutige Sensorsignale vorliegen – dass also das Fahrzeug an verschiedenen Stellen mit ähnlich hoher Wahrscheinlichkeit vermutet wird. Die Wahrscheinlichkeitsdichte für die Positionsschätzung hat also mehrere lokale Maxima. Eine solche Dichteverteilung wird als multimodal bezeichnet.

Stellvertretend für die Positionsschätzung wird im Folgenden allgemein die Schätzung eines Zustandsvektors zum Zeitpunkt  $k$   $\mathbf{x}_k$  verwendet. Die Beschreibung eines dynamischen Systems, z. B. im Zustandsraum, ermöglicht die Errechnung des Zustandsvektors auf Basis auf das System wirkender Eingangsgrößen  $\mathbf{u}_k$ . Wenn die Eingangsgrößen nicht exakt bekannt sind, sondern zusätzlich unbekannte Störeinflüsse existieren, kann der Zustandsvektor lediglich geschätzt werden. Ein Teil des Zustandsvektors ist im Allgemeinen messbar. Diese Messung zum Zeitpunkt  $k$  wird in dieser Arbeit allgemein als  $\mathbf{y}_k$  bezeichnet.

Um Zustandsgrößen eines dynamischen Systems auf Basis verrauschter Sensorsignale zu schätzen, werden Bayessche Filter [102] eingesetzt. Dabei wird zunächst eine a priori-Dichtefunktion für den Zustandsvektor  $\mathbf{x}$  zum Zeitpunkt  $k$  ermittelt

$$p^-(\mathbf{x}_k | \mathbf{y}_{k-1}, \mathbf{u}_k) = \int p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{u}_k) p(\mathbf{x}_{k-1} | \mathbf{y}_{k-1}, \mathbf{u}_{k-1}) d\mathbf{x}_{k-1} \quad (5-26)$$

Hier ist  $p^-(\mathbf{x}_k | \mathbf{y}_{k-1}, \mathbf{u}_k)$  über dem gesamten Zustandsraum definiert und liefert für jeden (infinitesimalen) Punkt des Zustandsraums eine bedingte Wahrscheinlichkeitsdichte für  $\mathbf{x} = \mathbf{x}_k$  unter der Bedingung, dass gilt  $\mathbf{y} = \mathbf{y}_{k-1}$  und  $\mathbf{u} = \mathbf{u}_k$ . Sie ergibt sich aus dem Integral zweier Verteilungen: Der bedingten Wahrscheinlichkeitsdichte  $p(\mathbf{x}_{k-1} | \mathbf{y}_{k-1}, \mathbf{u}_k)$  von  $\mathbf{x}_{k-1}$ , und der bedingten Wahrscheinlichkeitsdichte  $p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{u}_k)$ .



Dabei beschreibt  $p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{u}_k)$  die Systemdynamik, also wie sich der Zustandsvektor mit der Zeit ändert. In der im Folgenden beschriebenen Anwendung entspricht das der Veränderung der Pose des VTF bei einem gegebenen Eingangsvektor  $\mathbf{u}_k$  (Poseänderung  $\Delta\phi'_n$ ,  $\Delta x'_n$  und  $\Delta y'_n$  nach (5-12) - (5-14)). Diese Prädiktion ist fehlerbehaftet aufgrund äußerer Störeinflüsse und systematischer Fehler.

Die Lösung des Integrals in (5-26) wird im Allgemeinen nicht tatsächlich bei der Realisierung eines Bayesschen Filters implementiert. Stattdessen wird sich zumeist auf einen endlichen Zustandsraum mit einer in diesen gegebenen Grenzen diskretisierten Beschreibung der Dichtefunktion beschränkt. Beim Kalmanfilter wird sogar von einer fest vorgegebenen Form der Dichtefunktion (Normalverteilung) ausgegangen, sodass lediglich Parameter dieser Dichtefunktion (Auto- und Kreuzkovarianzen sowie der Mittelwert) errechnet werden.

Nachdem aufgrund des Eingangssignals  $\mathbf{u}_k$  und des Systemzustands  $\mathbf{x}_{k-1}$  nach (5-26) eine a priori-Schätzung für die Dichtefunktion des neuen Systemzustands  $\mathbf{x}_k$  errechnet werden kann, liefert

$$p(\mathbf{x}_k | \mathbf{y}_k, \mathbf{u}_k) = \alpha_k p(\mathbf{y}_k | \mathbf{x}_k) p^-(\mathbf{x}_k | \mathbf{y}_{k-1}, \mathbf{u}_k) \quad (5-27)$$

die Dichtefunktion der a posteriori-Schätzung. Das Filter korrigiert mittels (5-27) die a priori-Schätzung der Dichtefunktion  $p^-(\mathbf{x}_k | \mathbf{y}_{k-1}, \mathbf{u}_k)$  auf Basis des Messsignals  $\mathbf{y}_k$ . Dabei wird  $p^-(\mathbf{x}_k | \mathbf{y}_{k-1}, \mathbf{u}_k)$  mit der Wahrscheinlichkeit  $p(\mathbf{y}_k | \mathbf{x}_k)$  multipliziert, dass  $\mathbf{y}_k$  bei dem hypothetischen Zustandswert  $\mathbf{x}_k$  gemessen würde. Dieser Term trägt Sensorfehlern (Rauschen, systematische Fehler) Rechnung. Die Konstante  $\alpha_k$  ist notwendig, um das Integral von  $p(\mathbf{x}_k | \mathbf{y}_k, \mathbf{u}_k)$  auf 1 zu normieren.

In Bild 28 ist die Funktionsweise eines Bayesschen Filters dargestellt. Im gezeigten Beispiel wird die Position eines Roboters im eindimensionalen Zustandsraum geschätzt. Für die Positionsermittlung werden Türen sensortechnisch erfasst. Dem Roboter sind vorher die Positionen der Türen bekannt. Auch die Genauigkeit der Sensoren zur Erfassung der Türen ist dem Roboter bekannt. Damit kennt der Roboter die bedingte Wahrscheinlichkeit  $p(\mathbf{y}_k = 1 | \mathbf{x}_k)$ , dass der Sensor eine erkannte Tür in Abhängigkeit von der Position des Roboters meldet. Die anfängliche Position des Roboters ist unbekannt. Somit ist die Dichtefunktion  $p^-(\mathbf{x}_0)$  gleichverteilt über dem gesamten betrachteten Zustandsraum.

Zur Vereinfachung wird in diesem Beispiel die a posteriori-Dichtefunktion nach (5-27) nur dann errechnet, wenn das Sensorsignal  $\mathbf{y}_k$  eine Tür anzeigt. Wird keine Tür wahrgenommen, berechnet das Filter lediglich die a priori-Dichtefunktion nach (5-26) auf Basis des zurückliegenden  $p^-(\mathbf{x}_{k-1} | \mathbf{y}_{k-2}, \mathbf{u}_{k-1})$  anstelle von  $p(\mathbf{x}_{k-1} | \mathbf{y}_{k-1}, \mathbf{u}_{k-1})$ .

Diese Updates der jeweiligen a posteriori-Schätzung sind in (b) und (d) zu sehen. In (c) und (e) hingegen sind die Dichtefunktionen der a priori-Schätzungen dargestellt. Diese flachen nach einiger Zeit aufgrund der Systemungenauigkeit  $p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{u}_k)$  entsprechend (5-26) ab.

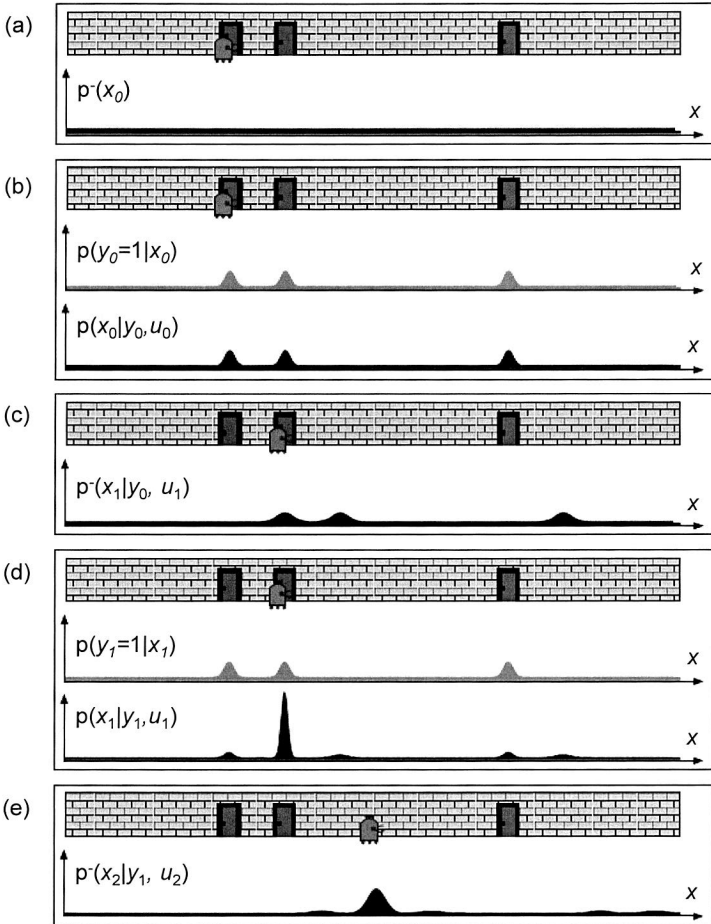


Bild 28: Ein Roboter schätzt seine Position anhand einer bekannten Wahrscheinlichkeit  $p(y = 1|x)$  auf Basis erkannter Türpositionen mit Hilfe eines Bayesschen Filters. Quelle [33]

Um ein Bayessches Filter zu implementieren, existiert eine Vielzahl von Möglichkeiten. In [33] werden verschiedene Implementierungen von Bayes-Filtern hinsichtlich abbildbarer Wahrscheinlichkeitsdichtefunktion, Genauigkeit, Stabilität, Effizienz sowie Implementierungsaufwand verglichen. Eine Auswahl ist in Tabelle 12 wiedergegeben. Da hier nicht im Detail auf die mathematischen Hintergründe der verschiedenen Schätzverfahren eingegangen werden kann, sei hierfür auf die Literatur [102], [33]

verwiesen. Zusammengefasst kann gesagt werden, dass das Kalmanfilter effizient arbeitet, jedoch von einer unimodalen (normalverteilten) Dichtefunktion ausgeht. Daher ist es zur Lösung des vorliegenden Problems nicht geeignet. Aufgrund seiner hohen Effizienz wird es in 6.4 detailliert beleuchtet, da es zur Zustandsschätzung von Weltobjekten eingesetzt wird.

Multi-Hypothesen-Tracker MHT [33] ermöglichen die Schätzung der Dichtefunktion durch mehrere Normalverteilungen. Es werden sinngemäß mehrere Kalmanfilter parallel betrieben, um verschiedene Hypothesen verfolgen zu können und somit mehrere ausgeprägte Maxima in der Dichtefunktion zu realisieren.

Gitterbasierte Ansätze [20] modellieren die Dichtefunktion diskret und stückweise konstant. Hierzu wird die Dichtefunktion in Regionen gleicher Größe aufgeteilt. Für jede der Regionen wird eine Wahrscheinlichkeitsdichte ermittelt. Somit lassen sich beliebige Dichtefunktionen abbilden – jedoch über einem diskreten Zustandsraum.

Partikelfilter, auch bekannt als SMC-Filter (Sequenzielle Monte-Carlo-Methode) [102], basieren auf einer konstanten Anzahl von Partikeln, die im Zustandsraum verteilt sind. Dabei repräsentiert das  $m$ -te Partikel  $\mathbf{x}^{[m]}$  die kleinste abbildbare Wahrscheinlichkeit für  $\mathbf{x} = \mathbf{x}^{[m]}$ . Durch die Dichte von Partikeln im Zustandsraum kann eine beliebige diskrete Wahrscheinlichkeitsdichte über einem quasi kontinuierlichen Zustandsraum repräsentiert werden (im Sinne der Rechengenauigkeit des benutzten Computers).

Tabelle 12: Vergleich von Bayes-Filter-Implementierungen (+ gut; 0 neutral; - schlecht) [33]

	Kalman	MHT	Gitterbasiert	Partikel
Dichtefunktion	Unimodal	Multimodal	Diskret	Diskret
Genauigkeit	+	+	0	+
Robustheit	0	+	+	+
Effizienz	+	0	-	0
Implementierung	0	-	0	+

Aufgrund der abbildbaren multimodalen Dichtefunktion, der hohen erreichbaren Genauigkeit, der hohen Robustheit und des geringen Implementierungsaufwands erfolgt die Schätzung der Position eines VTF über einem Zielmarker mittels Partikelfilter. Seine Funktionsweise ist in Tabelle 13 als Pseudocode dargestellt. Dabei wird von einer Menge  $\mathbf{X}_{k-1}$  mit  $M$  Partikeln ausgegangen. Jedes Partikel in dieser Menge stellt einen Punkt im Zustandsraum dar. Zusätzlich wird ein Eingangssignalvektor  $\mathbf{u}_k$  und ein Messsignalvektor  $\mathbf{y}_k$  dem Filter zugeführt. Die Partikelmenge  $\mathbf{X}_k$  für den Folgezeitschritt ergibt sich durch Generierung neuer Partikel  $\mathbf{x}_k^{[m]}$  aus den Partikeln der alten Partikelmenge  $\mathbf{X}_{k-1}$  (Zeile 4), Gewichtung dieser Partikel in Abhängigkeit des Messsignals  $\mathbf{y}_k$  (Zeile 5) und Ziehen von  $M$  Partikeln mit Zurücklegen aus  $\mathbf{X}_k$  (Zeile 9). Somit ergibt sich eine neue Partikelmenge  $\mathbf{X}_k$ , die die Dichteverteilung des Zustandsvektors  $\mathbf{x}$  zum Zeitpunkt  $k$  repräsentiert.

Tabelle 13: Der Partikelfilter-Algorithmus mit Resampling [102]

---

```

1:   Algorithmus Partikelfilter( $\mathbf{X}_{k-1}$ ,  $\mathbf{u}_k$ ,  $\mathbf{y}_k$ ):
2:        $\mathbf{X}_k = \mathbf{X}_k = \mathbf{0}$ 
3:       for  $m = 1$  to  $M$  do
4:           sample  $\mathbf{x}_k^{[m]} \sim p(\mathbf{x}_k | \mathbf{u}_k, \mathbf{x}_{k-1}^{[m]})$ 
5:            $w_k^{[m]} = p(\mathbf{y}_k | \mathbf{x}_k^{[m]})$ 
6:            $\mathbf{X}_k = \mathbf{X}_k + [\mathbf{x}_k^{[m]}, w_k^{[m]}]$ 
7:       endfor
8:       for  $m = 1$  to  $M$  do
9:           ziehe  $i$  mit Wahrscheinlichkeit  $\sim w_k^{[i]}$ 
10:             $\mathbf{X}_k = \mathbf{X}_k + \mathbf{x}_k^{[i]}$ 
11:       endfor
12:       return  $\mathbf{X}_k$ 

```

---

Um Zeile 4 des dargestellten Pseudocodes ausführen zu können, muss sowohl ein mathematisches Modell des Fahrzeugs als auch ein Modell des dabei auftretenden Fehlers vorhanden sein. In 5.2.2 wurden bereits die Änderungen der Fahrzeugpose hergeleitet. Eine Berechnung der Fahrzeugpose in einem beliebigen kartesischen Koordinatensystem lässt sich mit folgendem Gleichungssystem durchführen

$$\mathbf{x}_{k+1}^{[m]} = \begin{bmatrix} x_{k+1}^{[m]} \\ y_{k+1}^{[m]} \\ \varphi_{k+1}^{[m]} \end{bmatrix} = \begin{bmatrix} x_k^{[m]} \\ y_k^{[m]} \\ \varphi_k^{[m]} \end{bmatrix} + \begin{bmatrix} \cos(\varphi_k^{[m]}) & -\sin(\varphi_k^{[m]}) & 0 \\ \sin(\varphi_k^{[m]}) & \cos(\varphi_k^{[m]}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta x'_k \\ \Delta y'_k \\ \Delta \varphi'_k \end{bmatrix} + \begin{bmatrix} w x_k \\ w y_k \\ w \varphi_k \end{bmatrix} \quad (5-28)$$

Dieses wird auch zur Errechnung der Fahrzeugpose im Weltmodell (6-5) verwendet. Dabei bedeutet das hochgestellte  $[m]$ , dass die Fahrzeugpose im Zielmarker-Koordinatensystem für das Partikel  $m$  angegeben wird. Aufgrund des (nicht messbaren) Modellfehlers wird der Zustandswert durch Addition eines Rauschvektors  $w x_k$ ,  $w y_k$  und  $w \varphi_k$  zufällig leicht modifiziert. Im konkreten Fall wird von normalverteiltem, mittelwertfreiem Rauschen ausgegangen, mit einer Varianz von 1 mm<sup>2</sup> bzw. 0,03 rad<sup>2</sup>.

In Zeile 5 werden für alle Partikel Gewichtungen errechnet. Diese entsprechen jeweils der Wahrscheinlichkeit, dass das gemessene Sensorsignal vorliegt, wenn die Fahrzeugpose  $\mathbf{x}_k^{[m]}$  entspricht. Im vorliegenden Fall liefern Unterbodensensoren die binäre Untergrundfarbe an ihrer Position. Vergleichbar mit der Darstellung in Bild 28, liegt dem VTF eine Verteilung  $p(y_k = 1 | x_k)$  für erkannten, schwarzen Untergrund zur Verfügung. Das VTF weiß also, mit welchen Sensorwerten es in Abhängigkeit seiner Fahrzeugposition zu rechnen hat. Dazu wird ihm sowohl das exakte Layout des Zielpunktmarkers als auch die Position aller Fahrzeugsensoren relativ zum Fahrzeugmittelpunkt vor Beginn der Zustandsschätzung mitgeteilt.

Im Verlauf der Entwicklung des VTS hat sich gezeigt, dass bei einer Anzahl von 200 Partikeln die Ausführung des Codes aus Tabelle 13 0,141 s dauert (vgl. 7.3.2). Es ist also nicht möglich, die Daten der Unterbodensensoren mit einer Frequenz von 8 Hz oder mehr auszuwerten. Zusätzlich liefert eine Positionsbestimmung allein auf Basis der Farbwerte der Sensoren keine präzise Aussage über die Position. Das liegt daran, dass keine Rückschlüsse über die exakte Position des Fahrzeugs zu schließen sind, wenn der Sensor „schwarz“ meldet. Diesen Sachverhalt stellt  $p(y_k = 1|x_k)$  in Bild 29 dar. Wesentlich genauere Rückschlüsse über die Fahrzeugposition lässt eine Änderung des Sensorwerts – hier mit  $y_{toggle}$  bezeichnet – zu. Dann nämlich befindet sich der Sensor über einem Farbübergang von schwarz nach weiß oder weiß nach schwarz – hier mit Farbkante bezeichnet. Die zugehörige Dichtefunktion ist in Bild 29 mit  $p(y_{toggle}|x_k)$  bezeichnet.

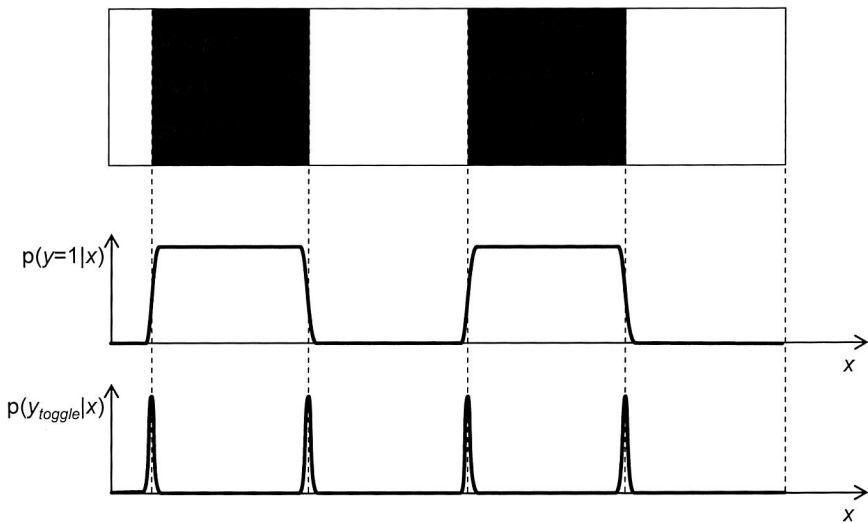


Bild 29: Wahrscheinlichkeitsdichten für Sensorsignale „schwarz“ ( $y = 1$ ) und „Farbkante“ ( $y_{toggle}$ ) über der Position  $x$  bei einer eindimensionalen Betrachtung des Überfahrens eines Schachbretts.

Entsprechend dieser Überlegungen wird die Strategie verfolgt, lediglich bei der Detektion von Farbkanten die a posteriori-Schätzung analog zu (5-27) durchzuführen. Konkret bedeutet das, in jedem Zyklus (immer, wenn neue Sensordaten der Inkrementalencoder,  $u_k$ , bereitstehen) eine a priori-Schätzung der aktuellen Dichtefunktion vorzunehmen. Dies geschieht durch Modifikation der gesamten Partikelpopulation mittels (5-28) entsprechend Tabelle 13, Zeile 4. Wenn  $y_k$  die Detektion einer Farbkante liefert, werden entsprechend Zeile 5 und 9 in Tabelle 13 alle Partikel gewichtet und eine neue Population aus der Alten durch Ziehen mit Zurücklegen generiert. Dabei hat es sich bewährt, zur Gewichtung der Partikel alle vier Unterbodensensoren heran zu ziehen. Sensoren, die keine Farbkante detektiert haben, liefern nämlich durch die

erkannte Untergrundfarbe weitere Informationen über die aktuelle Pose des VTF. Gute Ergebnisse der Poseschätzung lassen sich mit folgender Gewichtsfunktion erreichen,

$$w_k^{[m]} = \frac{1}{b(d^2)} \cdot a \quad (5-29)$$

mit der Basis  $b = 1,02$ , und dem Korrekturwert aufgrund der korrekten Farbe aller anderen Sensoren  $a$ :

$$a = 0.5^n. \quad (5-30)$$

Dabei stellt  $n$  die Anzahl der restlichen Sensoren dar, die nicht die tatsächlich vom VTF gemessene Farbe anzeigen. Der euklidische Abstand zwischen der zum Partikel  $\mathbf{x}_k^{[m]}$  gehörenden Sensorposition und der am nächsten gelegenen Farbkante (in mm) wird als  $d$  bezeichnet. Die sich ergebende Gewichtsfunktion ist in Bild 30 links für ein VTF mit zwei Unterbodensensoren dargestellt.

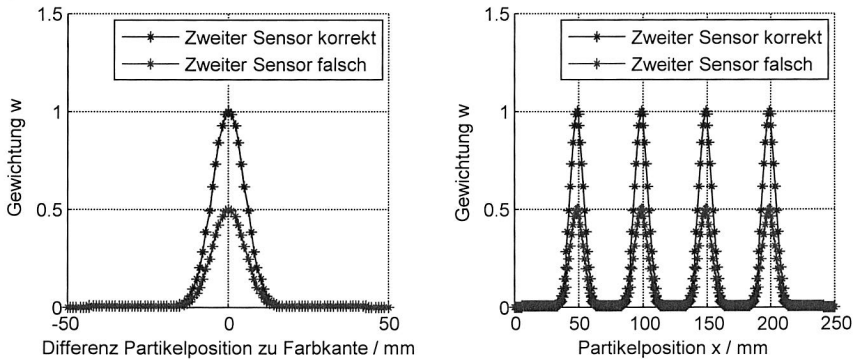


Bild 30: Links: Gewichtungsfunktion zur Bewertung eines Partikels im Zustandsraum in Abhängigkeit vom Sensorabstand zur nächsten Farbkante und dem Sensorwert des zweiten Sensors. Rechts: Sich ergebende Gewichtungsfunktion dargestellt über der x-Koordinate eines Schachbrettmusters mit 50 mm Schachbrettfeldbreite.

Es ist erkennbar, dass Partikel die einem Abstand von mehr als 15 mm zur nächsten Farbkante aufweisen, kaum mehr gewichtet werden ( $w(\pm 15 \text{ mm}) \approx 0,012$ ). Das ist sinnvoll, da die Breite eines Felds des Schachbrettmusters 50 mm beträgt. Somit ergibt sich die für eine Dimension (x-Richtung) in Bild 30 rechts gezeigte Gewichtung in Abhängigkeit von der Partikelposition, wenn ein Sensor eine Farbkante detektiert. Entsprechend Tabelle 13, Zeile 5 und Bild 29 müsste die Gewichtsfunction die Wahrscheinlichkeitsdichte  $p(y_{toggle} | x_k)$  abbilden. Diese hat jedoch wesentlich schmalere Peaks, als die verwendete, in Bild 30 rechts gezeigte Gewichtsfunction. Das liegt darin

begründet, dass das Partikelfilter mit einer endlichen Zahl von Partikeln arbeitet. Somit kann es vorkommen, dass in der a priori-Dichtefunktion  $bel(\mathbf{x}_{k-1})$  keine Partikel an der Stelle vorhanden sind, an der das VTF aufgrund der Sensorsignale vermutet wird. Somit müssen auch Partikel im nahen Umfeld, die entsprechend der Sensorsignale wahrscheinlich nicht den korrekten Zustand abbilden, möglichst hoch bewertet werden.

Eine Simulation des beschriebenen Vorgehens zur Positionsbestimmung relativ zu einem Zielmarker ist in Bild 31 gezeigt. Das VTF besitzt in diesem Fall zwei Unterbodensensoren, die um 95 mm nach vorne und um 45 mm nach links bzw. rechts vom Fahrzeugmittelpunkt verschoben sind. Das Partikelfilter ist entsprechend Bild 27 hier und in allen folgenden Simulationen mit 200 Partikeln initialisiert, die mit einer Standardabweichung von 50 mm in x- und y-Richtung um die erwartete Startposition normalverteilt sind. Ebenso ist die Orientierung der Partikel mittelwertfrei normalverteilt mit einer Standardabweichung von  $10^\circ$  relativ zum Lot auf die Orientierungslinie.

In Bild 31 (a) ist die resultierende Startpopulation der Partikel (die Position) zu sehen. Dabei zeigen blaue Punkte die Positionen der Partikel und gelbe Punkte die resultierenden Sensorpositionen. Die reale Fahrzeugposition ist grün dargestellt, die aus der Fahrzeugpose resultierenden, realen Sensorpositionen rot. Bei jeder inkrementellen Fahrzeugbewegung wird für jedes Partikel eine a priori-Zustandsschätzung entsprechend (5-28) durchgeführt. Sobald sich mindestens ein Sensorwert ändert, erfolgt die a posteriori-Schätzung entsprechend Tabelle 13 mit Gewichten der Partikel und Ziehen mit Zurücklegen (resampling). Dieser Schritt ist in Bild 31 (b) dargestellt. Hier sind die gewichteten Partikel orange mit einem zur Gewichtung proportionalen Radius dargestellt. Die von diesen gewichteten Partikeln gezogenen (teilweise auch mehrfach) sind blau gekennzeichnet. In Bild 31 (c) und (d) ist der Fortschritt der Zustandsschätzung gezeigt. Es ergeben sich schließlich drei ausgeprägte Maxima, an denen das Fahrzeug mit hoher Wahrscheinlichkeit vermutet wird.

Aufgrund der Symmetrie des Schachbrettmusters ist dieses Ergebnis zu erwarten gewesen. Um eine eindeutige Positionsschätzung zu ermitteln, müsste das Fahrzeug nach dem Schachbrettrand suchen. Hierfür wären aufwändige Manöver zu fahren. Alternativ lässt sich eine eindeutige Positionsbestimmung durch die Installation weiterer Unterbodensensoren erreichen.

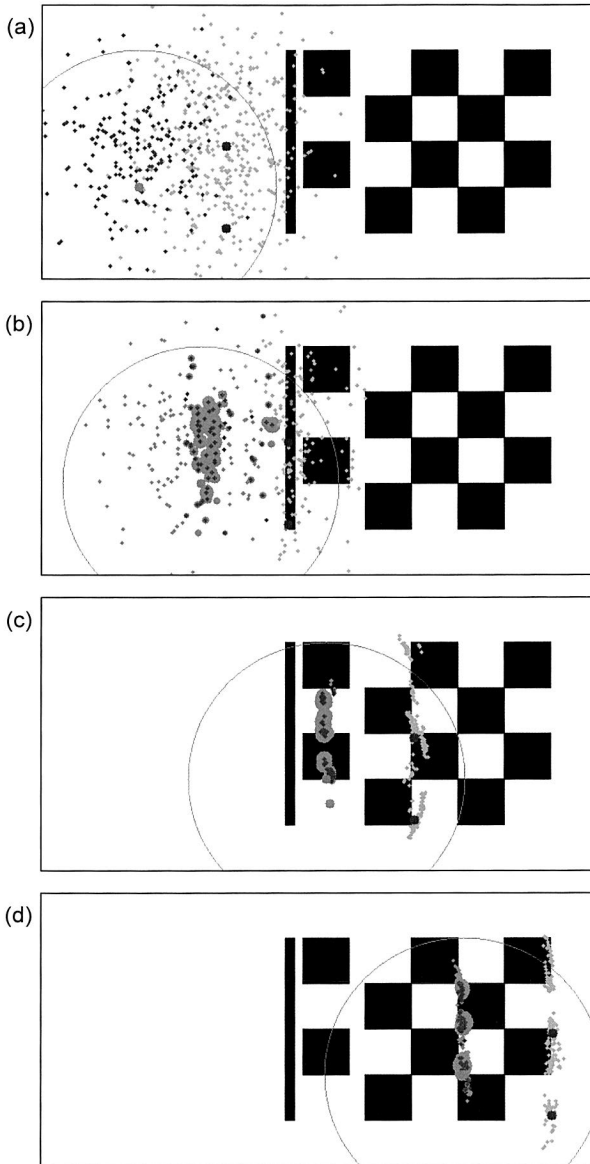


Bild 31: Poseschätzung des VTF beim Überfahren eines Zielmarkers: Fahrzeugmittelpunkt mit konzentrischem Kreis (grün), Fahrzeugsensoren (rot), a priori-Partikelpositionen (blau) mit Sensorpositionen (gelb) und gewichtete a posteriori-Partikelpositionen (orange).



Um zügig die Fahrzeugposition bei Andockvorgängen bestimmen zu können, werden weitere Unterbodensensoren beim VTF verwendet. Bild 32 zeigt verschiedene Konfigurationen zur Montage der Unterbodensensoren. Dabei wird lediglich die  $y$ -Position (Position in Querrichtung des VTF) relativ zu den Farbkanten des Schachbrettmusters dargestellt. Die Konfiguration  $a$  (über  $y_a$  aufgetragen) zeigt eine Verteilung, die eine Positionsauflösung in  $y$ -Richtung von  $1/4 s$  ermöglicht. Dabei gibt  $s$  die Seitenlänge eines Schachbrettfelds an. Die Konfiguration  $b$  (über  $y_b$  aufgetragen) trägt der Tatsache Rechnung, dass in der Mitte des Fahrzeugs ein mindestens 44 mm breiter Korridor existieren muss, da dort beim Befahren von Rampen die Führungsschiene lokalisiert ist (vgl. 5.1.3). Die schließlich gewählte Verteilung der Unterbodensensoren  $c$  (aufgetragen über  $y_c$ ) resultiert aus dem Bestreben, die Positionsauflösung im Bereich der Mittellinie zu erhöhen. Befindet sich das VTF im Bereich der Mittellinie, melden die Unterbodensensoren bereits bei einer Abweichung von  $1/5 s$  eine Farbkante.

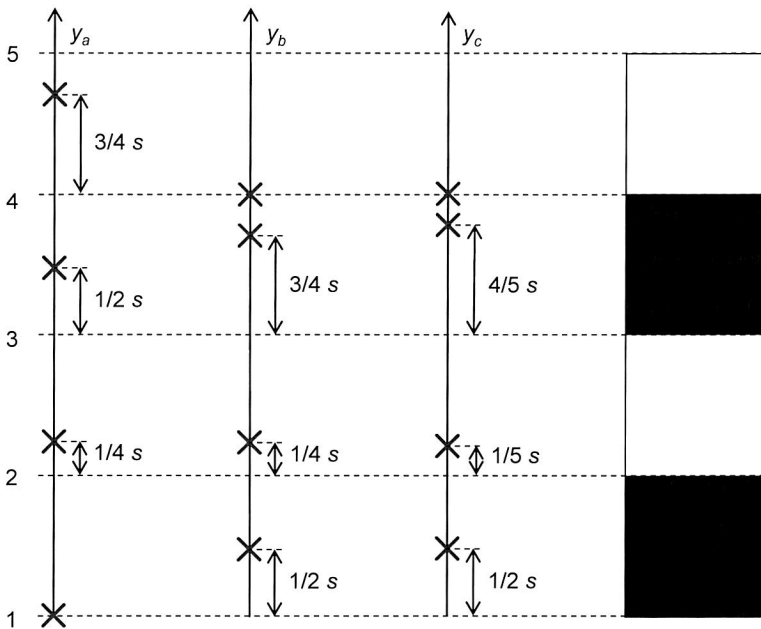


Bild 32: Verschiedene Sensorkonfigurationen in  $y$ -Richtung (Querrichtung) des VTF, relativ zu den Farbkanten in  $x$ -Richtung des Schachbrettmusters, bei angenommener Positionierung des Fahrzeugmittelpunkts über der Markermittellinie.

Bild 33 zeigt die Resultate der Positionsbestimmung mit den Sensorkonfigurationen  $b$  und  $c$ . Dabei ist einerseits zu erkennen, dass in beiden Fällen eine eindeutige Positionsbestimmung erfolgt, da ein ausgeprägtes Maximum der geschätzten Dichtefunktion vorliegt. Zusätzlich ist zu erkennen, dass – wie beabsichtigt – die Varianz der Dichteschätzung nach dem Überfahren des Zielmarkers geringer ist, wenn die

Unterbodensensoren entsprechend Konfiguration c montiert sind. Aus Tabelle 14 sind die Zahlenwerte der erreichten Varianzen zu entnehmen.

Tabelle 14: Varianz der Zustandsschätzung mittels Partikelfilter nach dem Überfahren des Zielmarkers zu den in Bild 33 (b) bzw. (d) gezeigten Resultaten mit Sensorkonfiguration b bzw. c entsprechend Bild 32.

	Konfiguration b	Konfiguration c
Varianz in x-Richtung	1,93 mm <sup>2</sup>	1,38 mm <sup>2</sup>
Varianz in y-Richtung	15,74 mm <sup>2</sup>	6,76 mm <sup>2</sup>
Varianz von $\varphi$	0,0012 rad <sup>2</sup>	0,0013 rad <sup>2</sup>

Um zu ermitteln, wann das Partikelfilter die Pose des VTF korrekt schätzt, wird permanent die Varianz der Positionsschätzung in y- und x-Richtung mit einem Referenzwert verglichen. Sobald beide Varianzen den jeweiligen Schwellwert unterschreiten, erfolgt durch Mittelwertbildung die exakte Positionsschätzung für das VTF. Von der dann ermittelten Pose aus wird mit dem in 5.2.3 beschriebenen Tracker der Weg zur Zielpose (Bild 27 (5)) zurückgelegt.

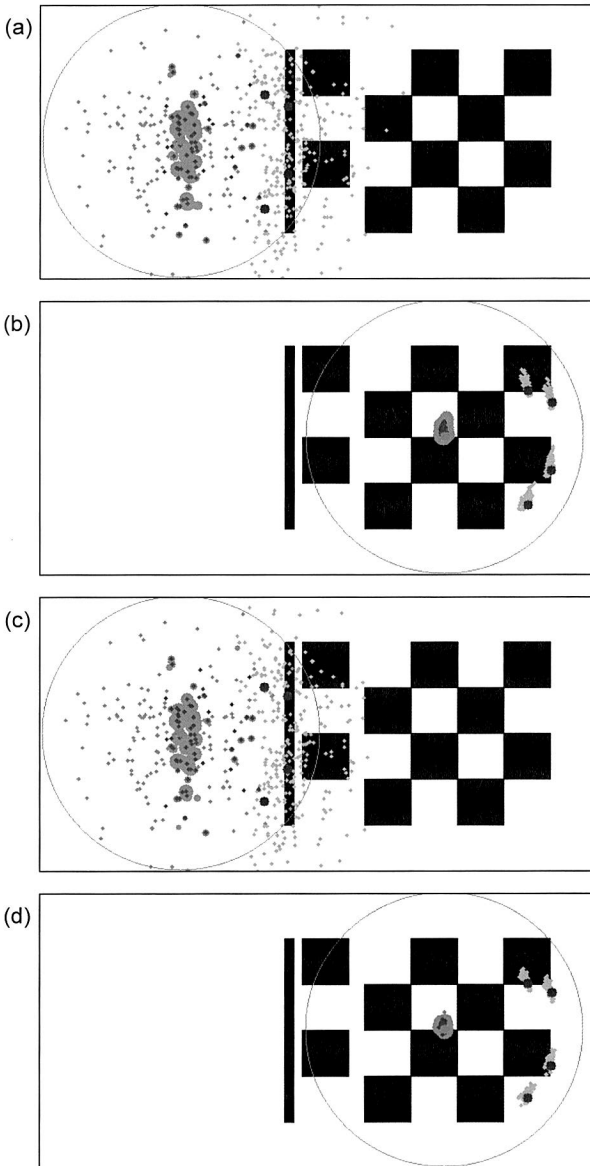


Bild 33: Vergleich der Positionsschätzung mit Sensorkonfiguration b (a) und (b) und c (c) und (d) aus Bild 32. In (a) und (c) sind jeweils die Populationen nach der ersten Detektion einer Farbkante dargestellt, in (b) und (d) die Positionsschätzungen nach dem Überfahren des Zielmarkers.



## 6 Ein zentrales Weltmodell befähigt VTF zum autonomen Betrieb

Die folgenden Kapitel beschäftigen sich mit den Softwarekomponenten eines zentralen Servers. Dieser erzeugt das Weltmodell mit Hilfe der Daten von Deckenkameras und aller zur Verfügung stehenden Sensordaten betriebener VTF. Um die in Tabelle 9 formulierten Anforderungen an die stationären Komponenten des VTS zu erfüllen, erfolgt auf dem Server die mathematische Modellierung der verschiedenen Weltobjekte und die Schätzung der zugehörigen Zustandsvektoren. Zusätzlich werden hier Methoden zur Bildverarbeitung und Methoden zur Zuweisung von gemessenen zu geschätzten Weltobjektpositionen ausgeführt. Einen Überblick über die im stationären Teil des VTS implementierten Softwarekomponenten und die dort ausgetauschten Daten gibt Bild 34.

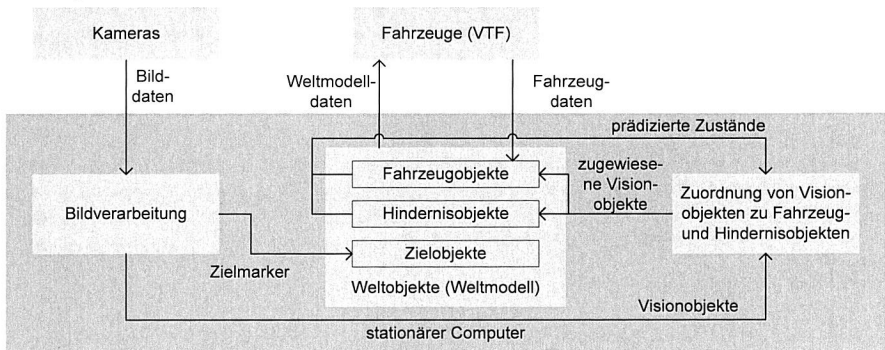


Bild 34: Datenfluss im stationären Computer (blau hinterlegt) des VTS. Das Weltmodell wird den VTF mittels WLAN zur Verfügung gestellt.

Die bei der prototypischen Umsetzung des Systems verwendete Hardware ist in 7 erläutert. Dort kommt lediglich eine Deckenkamera zum Einsatz. Auch in den folgenden Theorie-Kapiteln wird der Einfachheit halber von lediglich einer Deckenkamera ausgegangen.

### 6.1 Mathematische Modellierung von Weltobjekten

Wie in 4 bereits beschrieben und in Bild 34 dargestellt, gehören alle Objekte, die im Weltmodell hinterlegt sind, einer von drei verschiedenen Kategorien an. Je nach Kategorie werden verschiedene Eigenschaften eines Weltobjekts modelliert und verschiedene Algorithmen angewandt, um eine optimale Schätzung des Zustands des Objekts zu ermitteln. Im Folgenden ist erläutert, welche mathematische Modellierung den Kategorien Zielobjekt, Fahrzeugobjekt und Hindernisobjekt zugrunde liegt.

#### 6.1.1 Zielobjekte

Zur Markierung von Zielpositionen kommen im VTS optische Marker, sogenannte Zielmarker, zum Einsatz. Somit ist es möglich, auf bauliche Veränderungen des

Betriebsumfelds ohne weitere administrative Maßnahmen zu reagieren. Wird eine Zielposition und damit der Zielmarker verschoben (z. B. aufgrund der Modifikation einer Fertigungslinie), lässt sich über die Deckenkameras dessen neue Zielposition erfassen. Dazu müssen folgende drei Informationen von einem Zielmarker bereitgestellt werden:

- Position
- ID
- Orientierung

Dabei ist die eindeutige ID exakt einer charakteristischen Zielposition mit fester Funktion zugewiesen (z. B. Belichtungsautomat in der Halbleiterfertigung, Sterilisationsraum im Krankenhaus oder CNC-Bearbeitungszentrum im metallverarbeitenden Betrieb).

In Bild 35 ist ein Zielmarker dargestellt. Charakteristisch ist das 4 x 4 Felder große Schachbrettmuster, die Orientierungslinie und das vier Bit breite ID-Feld, welches die eindeutige ID des Markers (0 bis 15) wiedergibt. Dabei gilt von der Orientierungslinie aus gesehen das linke Feld als das höchstwertige Bit (most significant bit, MSB). Schwarze Felder werden als logische „1“ interpretiert, weshalb der dargestellte Zielmarker die ID 4 besitzt. Zur Bestimmung der Orientierung des Markers dient die Lage des erfassten ID-Felds. Mit Hilfe der Orientierungslinie lassen sich die Zuverlässigkeit und Präzision bei Andockvorgängen, wie in 5.2.6 beschrieben, erhöhen. Je nach Auflösung der Deckenkameras kann die ID auch durch mehr als 4 Schachbrettfelder dargestellt werden. Dann lassen sich mehr als 16 verschiedene Zielpositionen eindeutig markieren.

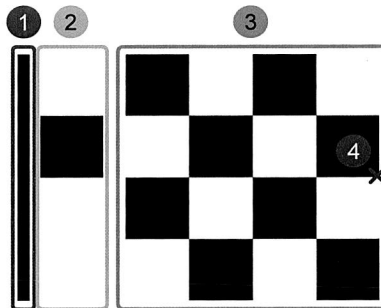


Bild 35: Ein Zielmarker lässt sich mittels Kamera anhand des Schachbrettmusters (3) erkennen und liefert neben der Zielposition (4) eine Orientierung und ID (2). Die Orientierungslinie (1) wird für Andockvorgänge benötigt.

Sobald die Bildverarbeitungsroutine einen Zielmarker erkennt, leitet sie die entsprechenden Daten direkt an das Weltmodell weiter. Das dort daraufhin erzeugte Zielobjekt besitzt den in (6-1) dargestellten Zustandsvektor:

$$\mathbf{x}_{Ziel} = \begin{bmatrix} x \\ y \\ \varphi \\ id \end{bmatrix} \quad (6-1)$$

Bei dieser Modellierung wird davon ausgegangen, dass die Zielpositionen unbeweglich sind. Eine Schätzung der Geschwindigkeit der Zielmarker erfolgt nicht.

Aufgrund der aufwändigen Rechenoperationen zur Zielmarkerdetektion (vgl. 6.2.2) und der damit verbundenen Rechenlast muss eine Aktualisierung der Zielobjekte manuell ausgelöst werden (vgl. 6.5). In Abhängigkeit der zur Verfügung stehenden Rechenleistung kann dies optional auch zyklisch geschehen. Vor dem Auslösen einer Zielmarkerdetektion werden alle im Weltmodell hinterlegten Zielobjekte gelöscht.

### 6.1.2 Hindernisobjekte

Alle von den Deckenkameras als Vordergrundobjekte erkannten Körper (siehe 6.2), die keinen Zielmarker repräsentieren, stellen sogenannte Visionobjekte dar. Diese werden als Positionsmessung eines beweglichen, physikalisch existenten Objekts interpretiert und repräsentieren somit entweder ein VTF oder ein Hindernis.

Die Repräsentation eines Hindernisobjekts im Weltmodell erfolgt mit Hilfe des in (6-2) dargestellten Zustandsvektors.

$$\mathbf{x}_{Hin} = \begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{bmatrix} \quad (6-2)$$

Dabei wird neben der Position des Hindernisses im globalen kartesischen Koordinatensystem die Geschwindigkeit in  $x$ - und  $y$ -Richtung modelliert. Bild 36 zeigt die Bedeutung des Zustandsvektors eines Hindernisobjekts im Weltkoordinatensystem.

Die zeitdiskrete Zustandsraumdarstellung eines Hindernisobjekts ist in (6-3) angegeben. Sie wird zur Errechnung der a priori-Zustandsschätzung nach (6-20) verwendet, wobei die Eingangsmatrix  $\mathbf{B}$  und das Eingangssignal  $\mathbf{u}$  bei Hindernisobjekten 0 betragen. In (6-3) sind die Abtastzeit der Deckenkameras und damit die Zykluszeit des Weltmodells mit  $dt$  bezeichnet.

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \dot{x}_{k+1} \\ \dot{y}_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & dt & 0 \\ 0 & 1 & 0 & dt \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \end{bmatrix} \quad (6-3)$$

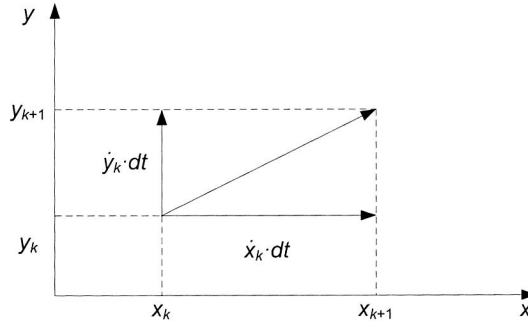


Bild 36: Der Zustandsvektor eines Hindernisobjekts gibt sowohl die Position als auch die Geschwindigkeit des Hindernisses im kartesischen Weltkoordinatensystem an.

Neben dem Zustandsvektor entsprechend (6-2) beinhaltet der Datensatz eines Hindernisobjekts eine ID. In Verbindung mit den zyklisch durch die Bildverarbeitung erkannten Vordergrundobjekten und der Zuordnung der Visionobjekte zu den existenten Hindernisobjekten (vgl. 6.3) lässt sich damit ein fehlertolerantes Trackingverfahren realisieren.

Weil die Geschwindigkeit eines neu erstellten Hindernisobjekts mit 0 initiiert wird, resultieren Geschwindigkeiten verschieden von 0 erst aus einer a posteriori-Schätzung des Zustandsvektors. Zur Schätzung des Zustandes von Hindernisobjekten kommt ein Kalmanfilter zum Einsatz, welches in 6.4.2 im Detail beleuchtet wird. Ob und wie ein Visionobjekt einem bereits bestehenden Hindernis- (oder Fahrzeug-) Objekt zugewiesen wird, ist in 6.3 beschrieben. Falls ein Visionobjekt keinem Weltobjekt zugewiesen werden kann, folgt die Instanziierung eines neuen Hindernisobjekts an der Position des Visionobjekts mit Geschwindigkeit 0. Die Löschung eines Hindernisobjekts erfolgt, nachdem ihm in 10 aufeinanderfolgenden Frames kein Visionobjekt zugewiesen wurde. Bild 37 stellt diese Zusammenhänge in einem UML-Aktivitätsdiagramm dar.

### 6.1.3 Fahrzeugobjekte

Visionobjekte (vgl. 6.2) können neben Hindernissen auch VTF repräsentieren. Wenn ein Visionobjekt keinem existenten Fahrzeug- oder Hindernisobjekt zuweisbar ist, führt das zur Generierung eines neuen Hindernisobjekts an der Position des Visionobjekts (vgl. 6.1.2). Somit werden VTF zunächst durch Hindernisobjekte modelliert. Wie in 5.2.5 beschrieben, erkennt ein VTF jedoch selbstständig mittels Trajektorienvergleich sein korrespondierendes Hindernisobjekt. Sobald das geschehen ist, sendet es diese Information an das Weltmodell. Dieses löscht das Hindernisobjekt und generiert ein Fahrzeugobjekt.



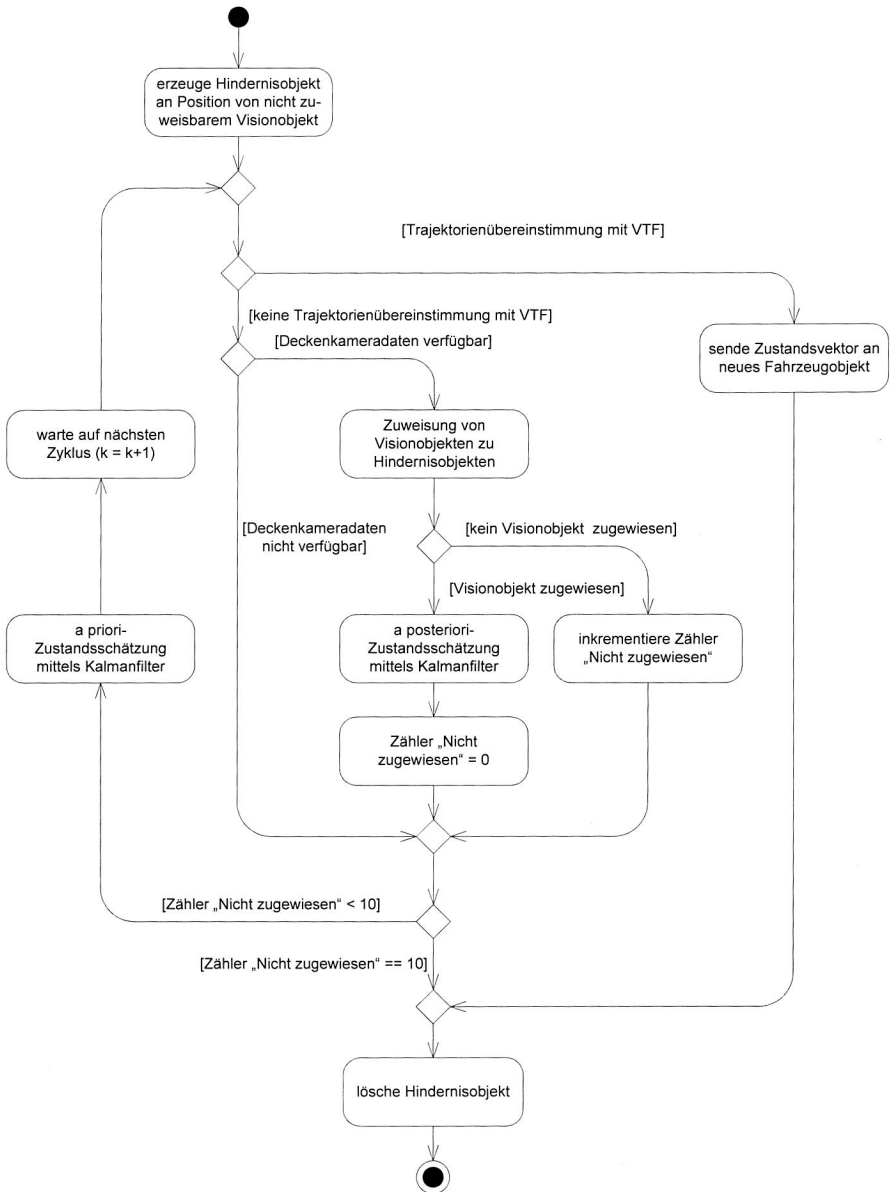


Bild 37: UML-Aktivitätsdiagramm eines Hindernisobjekts.

$$\mathbf{x}_{Fz} = \begin{bmatrix} x \\ y \\ \varphi \end{bmatrix} \quad (6-4)$$

Der in (6-4) beschriebene Zustandsvektor beinhaltet die Pose des Fahrzeugs, gegeben im Weltkoordinatensystem. Dabei wird die Orientierung  $\varphi$  im mathematisch positiven Sinn relativ zur positiven  $x$ -Achse dargestellt. Bild 38 zeigt die Bedeutung des Zustandsvektors eines Fahrzeugobjekts.

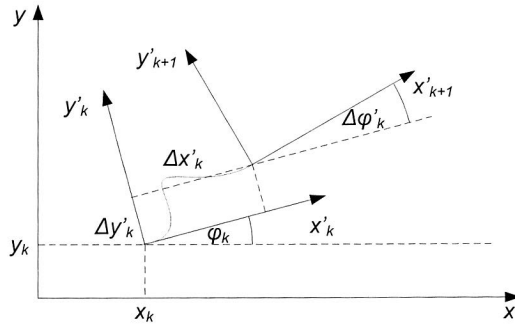


Bild 38: Der Zustandsvektor eines Fahrzeugobjekts gibt sowohl die Position als auch die Orientierung des Fahrzeugs im kartesischen Weltkoordinatensystem an.

Die zeitdiskrete Zustandsraumdarstellung für Fahrzeugobjekte (6-5) beinhaltet im Gegensatz zur Zustandsraumdarstellung von Hindernissen (6-3) auch ein Eingangssignal. Dieses wird von den Fahrzeugen entsprechend (5-12), (5-13) und (5-14) zur Verfügung gestellt und beschreibt die Poseänderung während des zurückliegenden Abtastschritts im  $x'y'$ -Koordinatensystem.

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \varphi_{k+1} \end{bmatrix} = \begin{bmatrix} x_k \\ y_k \\ \varphi_k \end{bmatrix} + \begin{bmatrix} \cos(\varphi_k) & -\sin(\varphi_k) & 0 \\ \sin(\varphi_k) & \cos(\varphi_k) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \Delta x'_k \\ \Delta y'_k \\ \Delta \varphi'_k \end{bmatrix} \quad (6-5)$$

Ebenso wie (6-3) für Hindernisobjekte wird (6-5) bei Fahrzeugobjekten zur Berechnung der a priori-Zustandsschätzung nach (6-20) verwendet. Konventionsgemäß besitzt die Zustandsraumdarstellung die Form  $\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$ . Anzumerken ist hierbei, dass die Poseänderung  $\mathbf{u}_k$  erst zum Zeitpunkt  $k+1$  verfügbar ist. Daher wird die a priori-Zustandsschätzung eines Fahrzeugobjekts im Gegensatz zur a priori-Schätzung eines Hindernisobjekts zu Beginn eines Zyklus ermittelt. Eine detaillierte Beschreibung der Zustandsschätzung von Fahrzeugobjekten ist in 6.4.3 gegeben.

Analog zu Hindernisobjekten findet eine a posteriori-Zustandsschätzung nur dann statt, wenn dem Fahrzeugobjekt ein Visionobjekt zugewiesen wird. Ebenso erfolgt die

Löschung, nachdem in zehn aufeinander folgenden Zyklen keine Zuweisung erfolgt ist. Bild 39 zeigt die beschriebenen Abläufe in einem UML-Aktivitätsdiagramm.

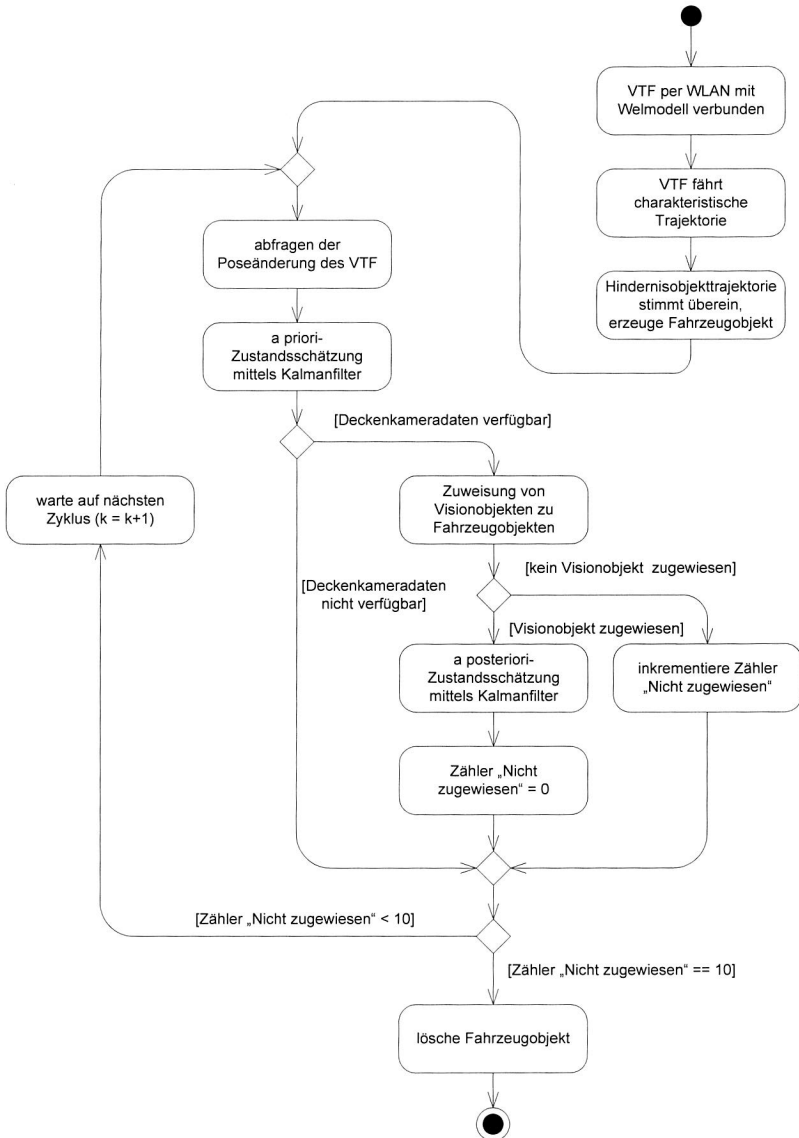


Bild 39: UML-Aktivitätsdiagramm eines Fahrzeugobjekts.

## 6.2 Optische Erkennung von Vordergrundobjekten und Zielmarkern

Um für die in 6.1 beschriebenen Weltobjekte mittels Deckenkameras eine Positionsmessung durchführen zu können, werden die Bilddaten der Deckenkameras verarbeitet. Ziel ist es, eine gute Messung der Position, Größe und Orientierung aller Objekte auf dem Hallenboden zu gewährleisten. Um dieses Ziel zu erreichen, können eine Vielzahl verschiedener Verfahren zur Bildverarbeitung eingesetzt werden [18].

Sind von einem Objekt bereits optische Merkmale bekannt, lassen sich diese in einem Bild suchen und somit Rückschlüsse auf die Pose des gesuchten Objekts ziehen. Populäre Methoden hierfür sind z. B. Eckendetektoren (corner detection) [42] oder die skaleninvariante Merkmalstransformation (scale-invariant feature transform SIFT) [66]. Bei beiden Verfahren wird eine kleine Menge von Bildpunkten ausgewählt, die eine ausreichende Menge an Informationen beinhaltet, um eindeutig im Bild wiedergefunden zu werden. Dabei lassen sich mittels SIFT die ausgewählten Punktmengen sogar wiederfinden, wenn sie in der Größe verändert oder verdreht sind.

Da die zu erkennenden Objekte im Falle des VTS keine optische Markierung besitzen und somit weder identifizierte Ecken oder skaleninvariante Merkmale aufweisen, können die genannten Verfahren nicht zur Erkennung von Hindernissen oder Fahrzeugen verwendet werden. Die Zielmarker sind jedoch durch ein Schachbrettmuster mit ausgeprägten Ecken gekennzeichnet. Daher lassen sich diese gut mittels Eckendetektoren erfassen (s. u.).

Um alle unmarkierten Objekte (Hindernisse und Fahrzeuge) optisch zu erkennen, lässt sich eines der einfachsten Verfahren zur Objektdetektion einsetzen. Dabei wird ausgenutzt, dass es sich beim VTS um eine indoor-Umgebung mit schwankungsarmen Lichtverhältnissen handelt. Zusätzlich ist bekannt, dass der Hallenboden den Hintergrund eines Kamerabilds darstellt. Wenn ein Referenzbild des Hintergrunds im Rechner hinterlegt ist, lassen sich durch Hintergrundsubtraktion [106] alle Vordergrundobjekte detektieren.

Im Folgenden wird zunächst auf die Methode der Hintergrundsubtraktion zur Objekterkennung eingegangen. Danach erfolgt die Beschreibung der Zielmarkererkennung mittels Eckendetektion.

### 6.2.1 Hintergrundsubtraktion zur Detektion von Vordergrundobjekten

Die Hintergrundsubtraktion stellt eines der am häufigsten eingesetzten Verfahren zur Objekterkennung in Sicherheitsanwendungen dar. Um Veränderungen in einer zu überwachenden Szenerie zu erkennen, wird permanent der bekannte Hintergrund von den aufgenommenen Bildern subtrahiert. Sobald ein Objekt in der Szene auftaucht, ist die Differenz zwischen aufgenommenem Bild und Hintergrundbild an dieser Stelle ungleich null. Somit lassen sich auf einfache Weise die Konturen des Objekts ermitteln. Dazu muss jedoch zunächst eine Schätzung des Hintergrundbilds vorhanden sein. Diese Hintergrundschätzung benötigt häufig ein Modell, welches alle statischen aber auch alle periodisch beweglichen Objekte des Hintergrunds (z. B. schwankender Baum im Wind) beinhaltet.

In dieser Arbeit werden verschiedene Ansätze zur Objekterkennung mittels Hintergrundsubtraktion untersucht. Neben der eigentlichen Hintergrundsubtraktion sind hierfür weitere Prozessschritte erforderlich:

- Grauwertbilderzeugung
- Hintergrundschätzung
- Hintergrundsubtraktion
- Konturerkennung
- Finden umschließender Rechtecke

Eine Diskussion jedes dieser Vorgänge folgt in den anschließenden Abschnitten. Bei diesen Überlegungen wird davon ausgegangen, dass das Hintergrundbild mit dem befahrbaren Boden mindestens einmal mittels Deckenkamera erfasst werden kann. Wird ein VTS installiert, nachdem sich bereits Hindernisse in der erfassten Szenerie befinden, müssen diese aus dem Hintergrundbild entfernt werden. Praktisch lässt sich das durch manuelle Modifikation der Bereiche darstellen, die ein Hindernis beinhalten. Dazu ist es nicht notwendig, exakt die Farbe des Hallenbodens zu kennen. Es genügt, eine vom Hindernis ausreichend verschiedene Farbe, um bei der Hintergrundsubtraktion das Hindernis als solches zu erkennen.

### **Grauwertbilderzeugung**

Während ein RGB-Bild aus den drei Farbanteilen für Rot, Grün und Blau (RGB) zusammengesetzt ist, stellt ein Grauwert den Helligkeits- oder Intensitätswert eines Pixels dar. Wird aus einem Farbbild ein Grauwertbild errechnet, gehen unweigerlich Informationen verloren. Aber zur Bildspeicherung und Weiterverarbeitung eines Grauwertbilds sind weniger Speicher und weniger Rechenzeit notwendig. Ebenso existieren in dem zur Bildverarbeitung verwendeten Softwaretool OpenCV (vgl. 7.3) viele Funktionen – wie z. B. Kantendetektoren oder Schwellwertvorgabe (thresholding) [90] –, die ausschließlich auf Basis von Grauwertbildern arbeiten. Daher wird das von den Deckenkameras aufgenommene Bild zunächst in ein Grauwertbild gewandelt. Hierfür existiert eine Konvention, die auf der Empfindlichkeit des menschlichen Auges für Lichtintensität in Abhängigkeit der Wellenlänge (Farbe) [13] beruht. Der Grauwert  $Y$  eines Pixels lässt sich aus seinem RGB Wert nach

$$Y = 0,299 \cdot R + 0,587 \cdot G + 0,114 \cdot B \quad (6-6)$$

errechnen [18]. Entsprechend (6-6) erfolgt die Grauwertbildung in dieser Arbeit.

Das ermittelte Grauwertbild wird zunächst einer Hintergrundsubtraktion unterzogen. Der Hintergrund beschreibt im vorliegenden Fall eines VTS den frei befahrbaren Hallenboden. Ein (dynamisches) Modell des Hintergrunds ist daher nicht notwendig, wenn ideal konstante Beleuchtung vorliegt. In diesem Falle genügt ein Referenzbild des Hintergrunds (Hallenboden). Im Rahmen der Arbeiten hat sich jedoch gezeigt, dass aufgrund von Sonneneinstrahlung ein statisches Referenzbild des Hintergrunds insbesondere nach längerer Betriebszeit zu Fehlern führt. Daher wurden verschiedene Verfahren zur Hintergrundschätzung untersucht.

### **Hintergrundschätzung**

Komplexe Hintergrundmodelle ermöglichen es, Szenen mit schwankenden Baumstäben, Wasseroberflächen, sich öffnenden und schließenden Türen, Licht, das an- und ausgeschaltet wird, oder Schatten von bewegten Objekten darzustellen. Hierbei ist prinzipiell das Ziel, sowohl den Mittelwert der Intensität als auch die Standardabweichung jedes einzelnen Pixels zu lernen. Somit ist es möglich, ein Pixel neben der reinen Intensität auch durch seine „Geschichte“ (stochastische Eigenschaften) dem Vorder- oder Hintergrund zuzuordnen. Diese Strategie lässt sich aufgrund des großen Speicherbedarfs und der hohen Rechenleistung nicht eins zu eins auf heutigen Rechnern implementieren. Stattdessen wird das Ziel verfolgt, vergleichbare Resultate mit effizienter Klassifizierung von Bildregionen und geeigneter, stochastischer Beschreibung dieser Regionen zu erreichen.

Um den Rechen- und Speicheraufwand bei der Modellierung von dynamischen Hintergrundsequenzen gering zu halten, eignet sich die Codebook Methode [18]. Hierzu ist es nicht notwendig, viele Bilder des Hintergrunds über einen gewissen Zeitraum zu speichern und zu verarbeiten. Stattdessen werden Regionen – sogenannte „Boxes“ – gelernt, in denen sich die drei zu einem Pixel gehörenden Größen (RGB bzw. YUV – je nach Farbraum) in der Vergangenheit bewegt haben. Sie beschreiben somit ausgeprägte Werte der Größe, die für dieses Pixel häufig in der betrachteten Vergangenheit angenommen wurden. Wenn die Größe des Pixels sich außerhalb der gelernten Regionen („Boxes“) bewegt, ist es wahrscheinlich, dass es sich um ein Vordergrundobjekt handelt.

Um die verhältnismäßig rechenaufwändige Codebook-Methode in Python zur Verfügung zu stellen, wurde der erstellte Pythoncode mit Hilfe von Cython (siehe 7.3) kompiliert. Trotz umfangreicher Optimierungsversuche ließ sich die benötigte Rechenzeit zur Schätzung des Hintergrunds nicht unter zwei Sekunden reduzieren. Aufgrund einer angestrebten Zykluszeit von maximal 0,2 s musste dieser Ansatz verworfen werden.

Geht man davon aus, dass in einer Fertigungsumgebung die Beleuchtungssituation hauptsächlich von der konstanten Raumbelichtung und nur zu geringem Anteil von schwankenden Umwelteinflüssen wie Sonnenlicht und Schattenwurf abhängt, liegt der Schluss nahe, auf die stochastischen Eigenschaften von Hintergrundpixeln bei der Hintergrundmodellierung zu verzichten. Aufgrund schnellerer Bearbeitung wurde daher eine Methode implementiert, die lediglich die aktuellen Bilddaten und die aktuelle Hintergrundschätzung heranzieht, um eine neue Hintergrundschätzung für jedes Pixel zu errechnen. Als gute Lösung hinsichtlich Laufzeit hat sich die Implementierung einer einfachen Hintergrundadaption mittels in den Python-Bindings verfügbarer OpenCV Funktionen erwiesen. Hier erfolgt die Veränderung des Intensitätswerts eines Pixels der Hintergrundschätzung, wenn das Vordergrundbild dem Hintergrundwert sehr ähnlich ist. Somit werden Stellen, an denen durch ein Vordergrundobjekt der Unterschied zwischen Hintergrund und Vordergrund groß ist, nicht verändert. Wenn sich der Hintergrund jedoch langsam verändert (z. B. weil die Sonne untergeht), kommen diese Änderungen auch in der Hintergrundschätzung zum Tragen. Wichtig ist es dabei, den Schwellwert

zur Erkennung von Änderungen des Hintergrunds hoch anzusetzen und damit ein ungewolltes Übernehmen von Vordergrundpixeln in den Hintergrund zu vermeiden. Wird der Schwellwert niedrig angesetzt, um Hintergrundänderungen zügig in die Hintergrundschätzung zu übernehmen, werden auch Vordergrundobjekte in den Hintergrund verschoben. In Bild 40 ist das Ergebnis des beschriebenen Algorithmus zur Hintergrundadaption zu sehen. Bild a) zeigt dabei das von der Deckenkamera aufgenommene Bild eines Transportwagens. Der im Rechner hinterlegte Referenzhintergrund (b) ist mit einer anderen Beleuchtung aufgenommen worden. Die Hintergrundschätzung (c) resultiert nach 1000 Zyklen bei verhältnismäßig hohem Schwellwert zur Übernahme von Vordergrundpixeln in den Hintergrund. Der resultierende Vordergrund (e) (Binärdarstellung (g)) stellt hohe Ansprüche an nachfolgende Filteralgorithmen, um das Vordergrundobjekt zu detektieren. Im Gegensatz dazu resultiert bei niedrigem Schwellwert eine bessere Schätzung der Vordergrundpixel (f) bzw. (h)). Es ist jedoch erkennbar, dass in die Hintergrundschätzung (d) wesentliche Teile des Vordergrunds übernommen worden sind. Das hat zur Folge, dass sich ein Großteil der eigentlichen Vordergrundpixel nicht als solche erkennen lässt. Darüber hinaus wird nach Entfernung des Transportwagens durch die Hintergrundsubtraktion eine Fehldetektion des Transportwagens erfolgen, und zwar so lange, bis die Hintergrundschätzung dem tatsächlichen Hintergrundwert angeglichen worden ist.

Es sei hier erwähnt, dass die in Bild 40 dargestellten Szenarien aus einer extremen, sprunghaften Veränderung der Beleuchtungssituation resultieren. Beide dargestellten Parametrierungen werden dem Problem nur zum Teil gerecht, sodass mit einer suboptimalen Vordergrunderkennung im Falle eines extremen Beleuchtungssprungs zu

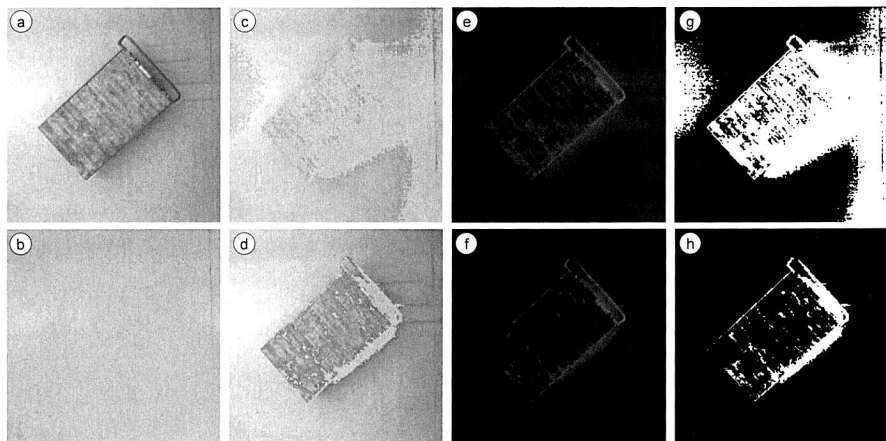


Bild 40: Implementierte Methode zur schnellen Hintergrundschätzung. a) Kamerabild (grau), b) Referenzhintergrund (andere Beleuchtung als in a)), c), d) Geschätzter Hintergrund; e), f) resultierender Vordergrund; g), h) Binärdarstellung des resultierenden Vordergrunds. c) e) und g) basieren auf einem hohen Schwellwert zu Übernahme von Pixeln in den Hintergrund, d), f) und h) basieren auf einem niedrigen Schwellwert. Nachteilhaft ist dabei die Übernahme von Teilen des Vordergrundobjekts in die Hintergrundschätzung

rechnen ist. Für sanfte Veränderungen der Beleuchtungsintensität stellt die in Bild 40 c), e) und g) dargestellte Lösung einen praktikablen Ansatz dar.

Aus oben genannten Gründen erfolgt die Hintergrundsubtraktion und anschließende Bildverarbeitung auf Basis von Grauwertbildern. Der Vollständigkeit halber sei hier angemerkt, dass bei Verwendung von Farbbildern, insbesondere im YUV Raum, die Luminanz Y (Lichtstärke pro Fläche) als eine der drei Farbkomponenten bei Beleuchtungsintensitätsschwankungen stark variiert. Die Chrominanz (Farbanteil U und V) hingegen ist weitestgehend unabhängig von der Beleuchtungsintensität und kann damit als beleuchtungsunabhängiger Indikator für Vordergrund- oder Hintergrundzugehörigkeit verwendet werden.

### **Hintergrundsubtraktion**

Sobald der Hintergrund geschätzt ist, lässt sich rechenaufwandsarm der Vordergrund durch Subtraktion des Hintergrunds von den Bilddaten errechnen. In Bild 41 ist das Ergebnis dargestellt.

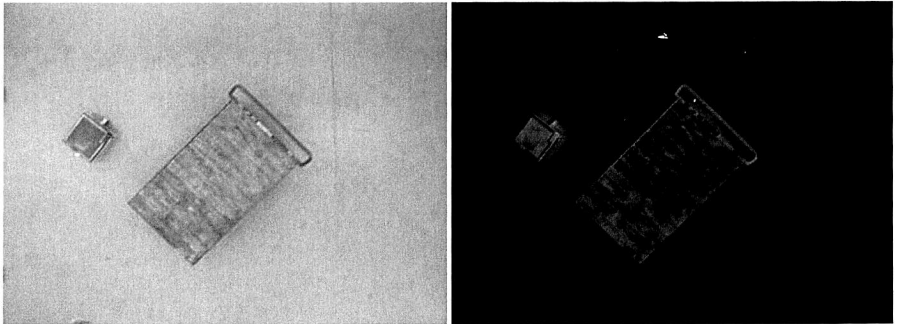


Bild 41: Grauwertbild der Deckenkamera eines VTF und eines Transportwagens (links). Durch die Hintergrundsubtraktion resultierender Vordergrund (rechts)

Es ist zu erkennen, dass einerseits der von den Objekten erzeugte Schatten zum Teil im Vordergrund auftaucht. Andererseits sind einige Pixel der Objekte nicht als zum Vordergrund gehörend erkannt worden. Dies passiert an den Stellen, wo der Grauwert des Hintergrunds dem Grauwert der Vordergrundobjekte entspricht. Um die Qualität der Objekterkennung zu erhöhen, lassen sich verschiedene Filter zur Bearbeitung des Vordergrundbilds verwenden. Bild 42 (a) zeigt einen Ausschnitt des Vordergrundbilds aus Bild 41, wobei im Original (a) Rauschpixel neben dem Rand der Vordergrundobjekte zu erkennen sind. Durch ein Medianfilter [52] mit einer  $3 \times 3$  Filtermaske lassen sich Rauschpixel sehr gut eliminieren. Zusätzlich hat es den Vorteil, kantenerhaltend zu wirken. Jedoch werden auch zum Rand der Vordergrundobjekte gehörende Pixel zum Teil gelöscht (Bild 42 (b)).

Ein Gaußfilter [18] hingegen zeichnet die Kanten „weich“. Das in Bild 42 (c) gezeigte Ergebnis resultiert aus dem Originalbild bei Verwendung eines Gaußfilters mit einer



3 x 3 Filtermaske. Vorteilhaft ist hier, dass die vorhandenen Kanten ausnahmslos erhalten bleiben.

Im Rahmen der Systementwicklung haben sich keine massiven Unterschiede der Qualität der Objekterkennung in Abhängigkeit des verwendeten Filters gezeigt. Zum Einsatz kommt aktuell ein Gaussfilter.

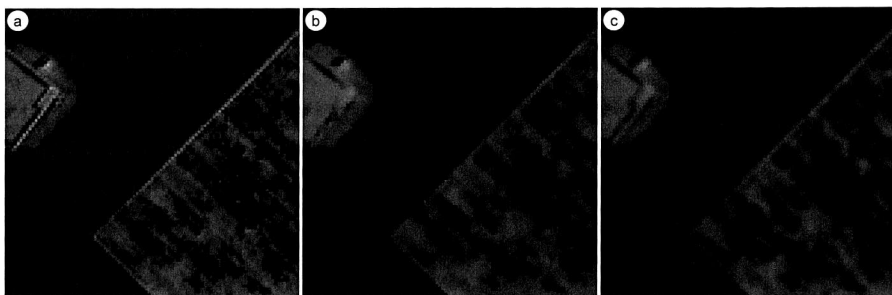


Bild 42: Original Vordergrundbild (a), gefiltert mit Medianfilter (b), gefiltert mit Gaussfilter (c)

### **Konturerkennung**

Die durch die Hintergrundsubtraktion gefundenen Daten sind reine Pixelwerte. Für jeden Bildpunkt existiert damit ein Wert, der je nach Höhe seines Betrags die Wahrscheinlichkeit für das Vorhandensein eines Vordergrundobjekts darstellt. Ziel ist es jetzt, die Vordergrundobjekte in ihrer Form zu erfassen. Dabei sind lediglich die Konturen der Vordergrundobjekte von Interesse. Hierzu muss zunächst ein binäres Bild errechnet und jedes Pixel eindeutig dem Vorder- oder Hintergrund zugeordnet werden. Mittels Schwellwertvorgabe lässt sich dieses Ziel erreichen. Der Schwellwert ist dabei als geeigneter Kompromiss zwischen „Erkennung von Vordergrundobjekten“ und „Vermeidung von Fehldetektionen (Rauschen)“ zu wählen. In Bild 43 (a) ist das resultierende Binärbild zu sehen.

Für die nachfolgenden Prozessschritte ist es wichtig, dass jedes Objekt aus zusammenhängenden Pixeln besteht. Hierzu wird das sogenannte „Closing“ durchgeführt. Diese Operation schließt Lücken zwischen nah beieinanderliegenden Pixeln. Closing umfasst zwei Unteroperationen, Dilatation und Erosion [94]. Bei der Dilatation wird eine Maske pixelweise über das Bild verschoben. Gehört das Pixel an der Position der Maske zum Vordergrund, wird die gesamte Menge der maskierten Bildpunkte in den Vordergrund übernommen. Bei der Erosion wird die gleiche Maske über das Bild geführt. Hierbei verbleibt jedoch jedes Pixel nur dann im Vordergrund, wenn die gesamte maskierte Pixelmenge zum Vordergrund gehört. Das Resultat des Closings mit einer 3 x 3 Maske zeigt Bild 43 (c).

Im Rahmen der Untersuchungen zur Bildverarbeitung kamen zwei Methoden zur Erstellung des Binärbilds zum Einsatz. Neben der beschriebenen Methode der Schwellwertvorgabe wurde auch die sogenannte Kantendetektion (edge detection) untersucht. Ein Kantendetektor sucht nach großen Änderungen der Bildintensität zwischen benachbarten Bildpunkten (hochfrequente Bildanteile bei Betrachtung im

Frequenzbereich). Neben Kanten lassen sich auf diese Weise aber insbesondere Rauschsignale detektieren. Daher existieren viele verschiedene Verfahren zur Kantendetektion, die häufig zunächst das Bild glätten, um Rauscheinflüsse zu reduzieren und dann verschiedene Methoden anwenden, um die vorhandenen Kanten zu verstärken. Schließlich wird mittels Schwellwertvorgabe eine Entscheidung getroffen, ob an einer Stelle eine Kante vorliegt oder nicht. Im Gegensatz zum reinen Schwellwertvorgabe (Bild 43 linke Spalte) kommen bei der Kantendetektion also Punkte aus der Nachbarschaft ebenso zum Tragen, um ein Pixel dem Vordergrund (Kante) oder dem Hintergrund zuzuordnen. Somit ist insbesondere für jene Pixel, die die Kontur der Vordergrundobjekte darstellen, die Wahrscheinlichkeit höher, als solche erkannt zu werden, als bei reiner Schwellwertvorgabe.

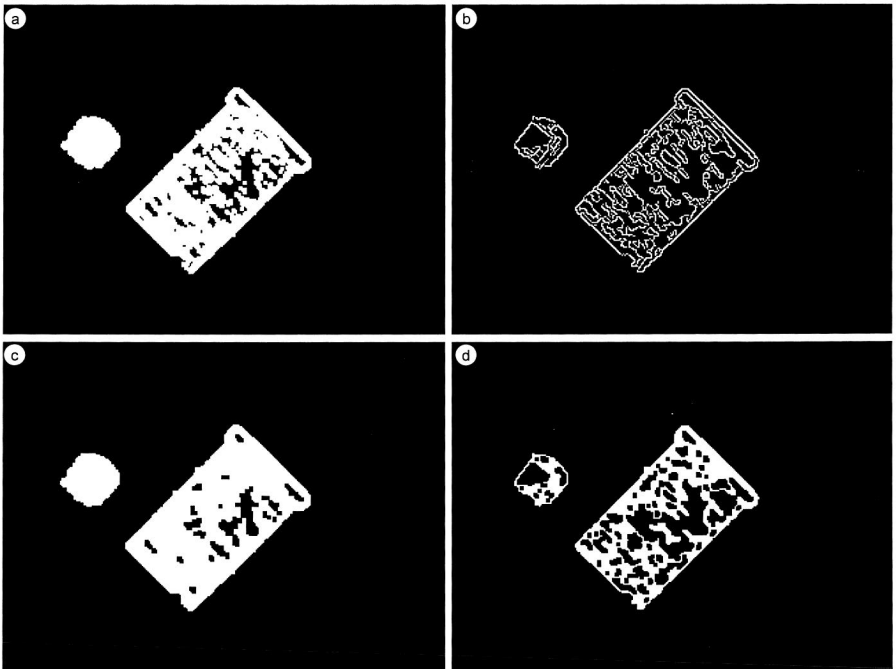


Bild 43: Eindeutige Zuordnung von Pixelgruppen zu Vorder- oder Hintergrund. Links Schwellwertvorgabe. Rechts: Kantendetektion. Zusätzlich werden mittels Closing nah zusammenliegende Pixel miteinander verbunden (zweite Zeile).

Der im Rahmen der Untersuchungen verwendete Kantendetektor ist der sogenannte Canny-Edge-Detektor [23]. Nach einer Bildglättung wird hier der Intensitätsgradient für jedes Pixel in  $x$ - und  $y$ -Richtung ermittelt und daraus eine resultierende Kantenrichtung errechnet. In Bild 43 (b) ist das Resultat der Kantendetektion zu sehen. Hier sind im Bereich von Kantenschnittpunkten keine Verbindungen zwischen den einzelnen Kanten zu erkennen. Dies ist eine Eigenheit des Canny-Edge-Detectors, die auf der Annahme beruht, dass sich immer nur eine Kante im Bereich des aktuell untersuchten Bildab-

schnitts (Canny-Operators) befindet. Eine Closing-Aktion verbindet die einzelnen detektierten Kantensegmente und liefert die gesuchte Außenkontur der Vordergrundobjekte (Bild 43 (d)).

### Finden umschließender Rechtecke

Nachdem die Außenkonturen aller Vordergrundobjekte auf einem der beschriebenen Wege als Binärbild ermittelt worden sind, ist es notwendig, diese Konturen auf möglichst einfache Art und Weise in Position, Orientierung und Form mathematisch zu beschreiben. Hierfür eignen sich Methoden zur Konturerkennung [99]. Diese liefern im Allgemeinen einen Baum von Konturen, wobei die Struktur des Baums die gegenseitige Umschließung der Konturen beschreibt. Der Wurzelknoten eines Baums stellt dabei immer die äußere gefundene Kontur dar. Dessen Kinderknoten sind sogenannte Löcher und beschreiben eine Innenkontur des Elternknotens. In Bild 44 sind die aus den Binärbildern von Bild 43 errechneten Konturen dargestellt. Außenkonturen sind weiß eingezeichnet, Innenkonturen rot. Von Interesse für die Objekterkennung sind nur jeweils die Wurzelknoten der Kontur-Bäume, also die Außenkonturen, die von keiner weiteren Kontur umschlossen sind.

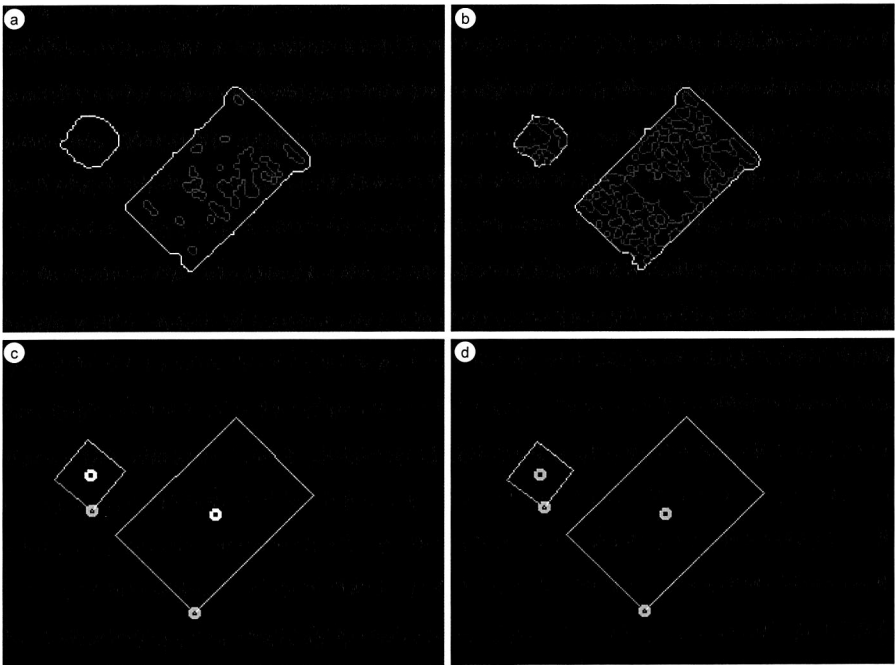


Bild 44: Aus den Binärbildern von Bild 43 errechnete Konturen (a) und (b). Ebenso sind in (c) und (d) die minimalen umschließenden Rechtecke zu sehen (links: Schwellwertvorgabe, rechts: Kantendetektion).

Die in Bild 44 in der ersten Zeile dargestellten Konturen sind geeignet, die Form und Pose aller durch die Deckenkameras erkannten Vordergrundobjekte zu beschreiben.

Nachteilhaft ist hierbei jedoch, dass die mathematische Modellierung einer beliebig komplexen Form rechen- und damit zeitintensiv wird. Um das Datenaufkommen bei der Beschreibung von Vordergrundobjekten niedrig zu halten, werden Vordergrundobjekte daher durch ein minimales, sie umschreibendes Rechteck repräsentiert. Im Verlauf der weiteren Berechnungen lässt sich demnach ein Vordergrundobjekt durch seinen Mittelpunkt, seine Größe (Seitenlängen  $a$  und  $b$ ) sowie seine Orientierung beschreiben. In Bild 44 sind die sich ergebenden, umschreibenden Rechtecke zur Repräsentation von Vordergrundobjekten zu sehen.

Um die beiden vorgestellten Methoden (Binärbilderzeugung mittels Schwellwertvorgabe bzw. mittels Kantendetektion) miteinander zu vergleichen, dient ein im Labor durchgeführtes Experiment:

Über je einen Zeitraum von ca. 200 s dokumentiert die Bildverarbeitungsroutine alle erfassten Objekte. Dabei wird die Szenerie aus Bild 41 mit statischer Hintergrundaufnahme betrachtet und die Konturerkennung mittels Schwellwertvorgabe und anschließend mittels Kantendetektion durchgeführt. Die Vorverarbeitung der Grauwertbilder erfolgt dabei einheitlich durch ein Gaussfilter. Als Gütemaß dient die Summe aller nicht erkannten Objekte (kein erkanntes Objekt im Abstand von bis zu 10 cm zum Erwartungswert), die Varianz der Positionsmessung der korrekt detektierten Objekte und die Summe der fehldetektierten Objekte. Tabelle 15 zeigt das Resultat der Untersuchung.

Tabelle 15: Vergleich der Methoden zur Objekterkennung bei Betrachtung der Szenerie aus Bild 41 über 1038 Frames (ca. 200 s). Dokumentiert werden alle nicht erkannten Vordergrundobjekte, die Varianz der Positionsmessung der korrekt erkannten Vordergrundobjekte und alle Fehldetektionen von Vordergrundobjekten.

	Schwellwertvorgabe	Kantendetektion
Nicht erkannt	0	0
Varianz Positionsmessung Objekt 0 (VTF)	x: 2,12 mm <sup>2</sup> y: 3,62 mm <sup>2</sup>	x: 24,69 mm <sup>2</sup> y: 18,12 mm <sup>2</sup>
Varianz Positionsmessung Objekt 1 (Transportwagen)	x: 2,65 mm <sup>2</sup> y: 2,47 mm <sup>2</sup>	x: 1,23 mm <sup>2</sup> y: 1,15 mm <sup>2</sup>
Fehldetektionen	0/1038	2/1038

Das in Tabelle 15 dargestellte Resultat zeigt keinen eindeutigen Favoriten hinsichtlich Varianz der Positionsmessung von Objekten. Da die Rate der Fehldetektionen bei Anwendung der Schwellwertvorgabe jedoch bei 0 liegt, ist diese Methode leicht favorisiert und kommt aktuell zur Anwendung.

Wie in Bild 34 zu sehen, stellt die Bildverarbeitungsroutine die Pose der Zielmarker und die sogenannten Visionobjekte (vgl. 6.1.2) zur Verfügung. Die bisher beschriebene Detektion von Vordergrundobjekten umfasst sowohl die Zielmarker, als auch die Visionobjekte. Das folgende Kapitel gibt daher darüber Aufschluss, wie sich die Zielmarker als solche erkennen lassen. Aus der Menge der von Rechtecken umschlossenen Vordergrundobjekte lassen sich daraufhin alle Zielmarker entfernen. Übrig bleiben die Visionobjekte, die zur Ermittlung von Hindernis- und Fahrzeugobjektposen

Anwendung finden (vgl. 6.3). Bild 45 zeigt ein Schema des gesamten Bildverarbeitungsprozesses.

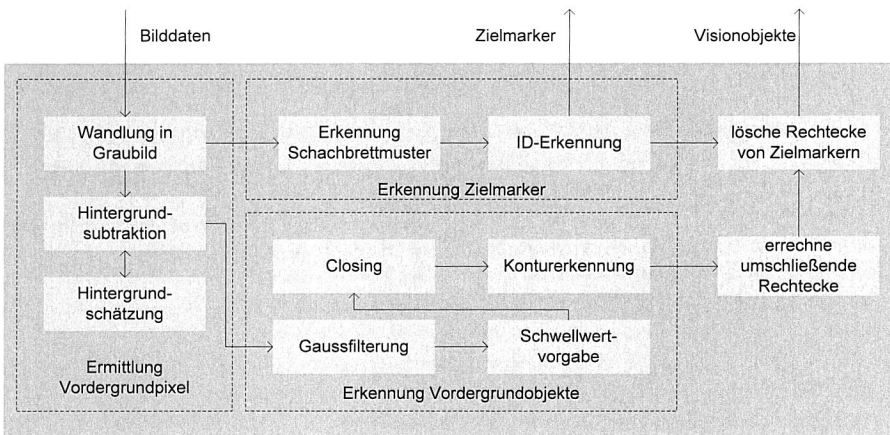


Bild 45: Gesamtprozess der Bildverarbeitung

### 6.2.2 Mustererkennung zur Detektion von Zielmarkern

Im Gegensatz zu Hindernissen und VTF sind Zielpositionen durch eindeutig erkennbare Marker gekennzeichnet. In Bild 27 ist ein solcher Zielmarker abgebildet. Die Erkennung der Position, Orientierung und ID eines Zielmarkers passiert mittels Deckenkamera. Hierfür ist es notwendig, die Schachbrettmuster der Zielpunktmarker zu detektieren und das ID-Feld zu finden.

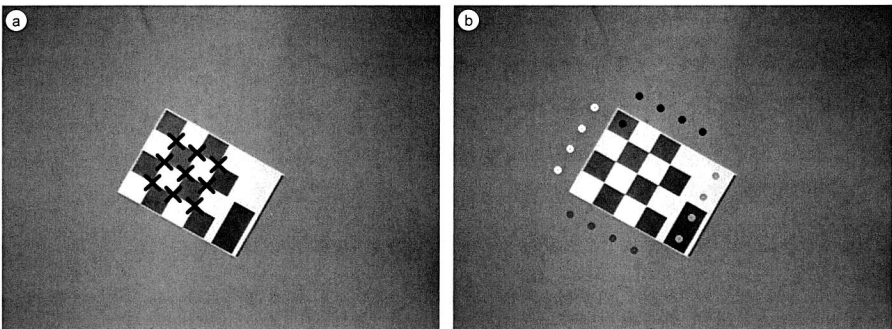


Bild 46: Erkannte Ecken des Schachbrettmusters (a) und Punkte, an denen nach einem ID-Feld gesucht wird (b).

Ein 4 x 4 Schachbrettmuster besitzt 3 x 3 ausgeprägte Ecken. Diese lassen sich z. B. mittels Harris Eckendetektor [42], basierend auf der zweiten Ableitung der Bildintensität in x- und y-Richtung um ein betrachtetes Pixel herum, erkennen. Können in einem Bild neun äquidistante Ecken in der 3 x 3 Struktur entsprechend Bild 46 (a) ermittelt werden, ist es wahrscheinlich, dass sich an dieser Stelle ein Zielmarker befindet. Das ID-Feld

wird anschließend an den vier möglichen Positionen gesucht. Dies passiert durch Vergleich der Intensitäten der Schachbrettfelder mit den Intensitäten der möglichen ID-Feld-Rechtecke. Dieser Vorgang ist in Bild 46 (b) dargestellt.

Sobald das ID-Feld gefunden ist, ist die Orientierung des Zielmarkers bekannt. Die ID wird aus den erkannten Intensitäten der ID-Feld Rechtecke entsprechend 6.1.1 errechnet.

### 6.3 Zuweisung von Visionobjekten zu Fahrzeug- und Hindernisobjekten

Im vorangegangenen Kapitel ist beschrieben, wie sich markierte und unmarkierte Objekte in einem einzelnen 2D-Bild erkennen lassen. Liegt eine Sequenz von Bildern vor – und das ist beim VTS der Fall – werden sogenannte Multi-Object Tracking-verfahren eingesetzt, um die Trajektorien von Objekten in den einzelnen Bildern zu ermitteln. In der Literatur sind verschiedene Ansätze zur Lösung dieses Problems zu finden. Prinzipiell sind hierfür zwei Aufgaben zu lösen: „Detektion von Objektpositionen“ und „Errechnen der wahrscheinlichsten Trajektorien für alle Objekte, durch Verbinden der detektierten Objektpositionen in geeigneter Weise“. Während in 6.2 die Detektion von Objekten erläutert ist, liegt hier der Fokus auf der korrekten Zuweisung der detektierten Positionen zu einer Trajektorie und damit zu einem Objekt.

In vielen Fällen stellt der Trackingprozess eine Markov-Kette erster Ordnung dar. Bei diesem rekursiven Ansatz erfolgt die Schätzung des folgenden Punkts der Trajektorie ausschließlich auf dem aktuellen Schätzwert des Objektzustands (z. B. Pose und Geschwindigkeit). Weiter in der Vergangenheit liegende Trajektorienpunkte werden zur Errechnung des folgenden Trajektorienpunkts nicht direkt einbezogen. Die Zuordnung der gemachten Messungen zu den prädierten Trajektorienpunkten erfolgt häufig im Sinne kleinster Distanzen zwischen Messung und Prädiktion. Ein Beispiel für einen sogenannten „nearest neighbor data association tracker“ ist in [67] gegeben.

Um die Fehleranfälligkeit auf Markov-Ketten erster Ordnung basierender Tracker zu reduzieren, lässt sich zusätzlich zum Tracking noch eine Objekterkennung durchführen. Somit ist es möglich, ein Objekt zusätzlich durch seine optischen Eigenschaften (z. B. Form oder Farbe) zu erkennen und seine Trajektorie zu errechnen [9].

Die genannten Methoden kommen jedoch, wenn alle Objekte ähnlich aussehen oder Verdeckungen und Fehldetektionen von Objekten auftreten, an ihre Grenzen.

Nicht an Markov-Ketten gebunden sind hingegen z. B. Multi-Hypothesen-Tracker, die mehrere zurückliegende Positionsmessungen mit mehreren hypothetischen zurückliegenden Pfaden abgleichen und den wahrscheinlichsten Pfad aller Objekte für den betrachteten Zeitraum zurückgeben [83]. Hier steigt die Zahl der möglichen Trajektorien (Hypothesen) exponentiell mit der Dauer des betrachteten Zeitraums an. Daher kann entweder nur ein kurzer zurückliegender Zeitraum betrachtet oder es müssen bereits vor dem Suchen der optimalen Lösung vielversprechende Trajektorien vorausgewählt werden.

Die getroffenen Hypothesen erlauben es auch, physikalische Randbedingungen wie Kollisionsvermeidung zu berücksichtigen [63]. Je größer die Anzahl der zusätzlich

betrachteten Randbedingungen ist, umso rechenaufwändiger stellt sich die Suche nach der optimalen Lösung dar. Diese beinhaltet alle Trajektorien der erfassten Objekte über den gesamten betrachteten Zeitraum. Zur Reduktion der Komplexität des Problems ist es dann umso wichtiger, zuvor eine Menge plausibler Hypothesen (Trajektorien) aufzustellen.

Ebenso werden die genannten Methoden kombiniert. In [77] erfolgt zunächst die Generierung kleiner Pfade mittels Kalmanfilter basierter nearest neighbor data association Tracker, die relativ strengen Abbruchkriterien unterliegen. Anschließend erfolgt die paarweise Zuordnung der verschiedenen Pfade durch die Anwendung der im Folgenden erläuterten Ungarischen Methode [59].

In der vorliegenden Arbeit stehen neben den über die Deckenkameras erfassten Objektpositionen Sensordaten der Fahrzeuge (vgl. 5.2.2) zur Verfügung. Das macht es einfacher, die Trajektorien von VTF fehlerfrei zu berechnen. Da zusätzlich nur begrenzt Rechenleistung zur Trajektorien- bzw. Zustandsschätzung von Fahrzeug- und Hindernisobjekten verfügbar ist, wird hier die Strategie verfolgt, das Tracking-Problem über eine Markov-Kette erster Ordnung zu lösen. Während die a priori-Zustandsschätzung für Fahrzeug- und Hindernisobjekte daher mittels Kalmanfilter auf Basis der in (6-3) und (6-5) gegebenen Objektmodelle erfolgt, passiert die Zuweisung von Positionsmessungen zu den Objektpositionsprädiktionen mit Hilfe der Ungarischen Methode. Dieses Kapitel richtet den Blick auf die Modifikation und Anwendung der Ungarischen Methode auf das vorliegende Problem. In 6.4 kommt das Kalmanfilter speziell unter dem Aspekt großer Totzeiten bei der Sensorsignalerfassung zur Sprache.

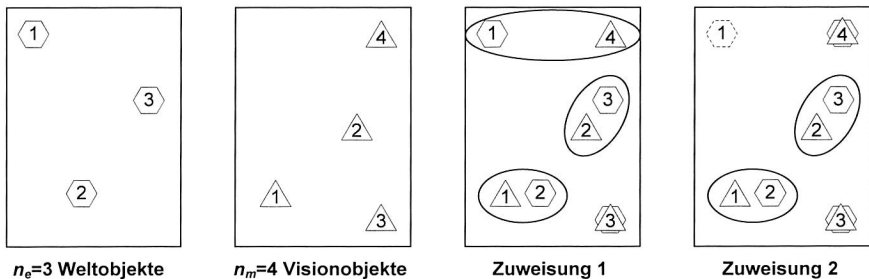


Bild 47: Zwei Mengen von Elementen (Weltobjekte und Visionobjekte) sollen einander zugewiesen werden. Zuweisung 1 ergibt sich mit der Ungarischen Methode. Mit Hilfe der beschriebenen Modifikation ist es möglich, eine maximal erlaubte Zuweisungsdistanz vorzugeben. Damit ergibt sich Zuweisung 2. Visionobjekte, die sich keinem Weltobjekt zuweisen lassen, führen zur Generierung neuer Weltobjekte.

Die gewichtete Zuordnung von  $n_e$  (a priori-) Objektpositionsprädiktionen zu  $n_m$  Visionobjekten stellt ein bipartites Matchingproblem dar [21]. Es umfasst genau zwei Mengen von Elementen (Messungen und Prädiktionen), beispielhaft dargestellt in Bild 47. Ziel ist es, jedem Element aus der einen Menge exakt ein Element aus der zweiten Menge zuzuordnen und dabei die globalen Zuordnungskosten zu minimieren. Lösungen zu dem beschriebenen Problem lassen sich mit der Ungarischen Methode ermitteln.

Hierzu wird von einer quadratischen Kostenmatrix  $\mathbf{C}$  ausgegangen. Diese basiert auf der nichtnegativen Matrix  $\mathbf{C}_0$  mit

$$\mathbf{C}_0 = (d_{ij}) \quad (6-7)$$

wobei  $d_{ij}$  die Zuweisungskosten (den Euklidischen Abstand) zwischen Zustandsschätzung  $e_i$  und Messung  $m_j$  beschreibt.

Wenn die Anzahl der Weltobjekte  $n_s$  von der Anzahl der Visionobjekte  $n_m$  differiert, ist  $\mathbf{C}_0$  nicht quadratisch. In diesem Fall lässt sich  $\mathbf{C}$  generieren, indem die geeignete Nullmatrix mit  $\mathbf{C}_0$  entsprechend (6-8) verbunden wird.

$$\mathbf{C} = \begin{cases} \mathbf{C}_0 & n_e = n_m \\ \begin{bmatrix} \mathbf{C}_0 & \mathbf{0}_{n_e \times (n_e - n_m)} \end{bmatrix} & n_e > n_m \\ \begin{bmatrix} \mathbf{0}_{(n_m - n_e) \times n_m} & \mathbf{C}_0 \end{bmatrix} & n_e < n_m \end{cases} \quad (6-8)$$

Die Ungarische Methode liefert die optimale Zuordnungsmatrix  $\mathbf{X}$  mit exakt einer „1“ in jeder Zeile und Spalte. Optimal bedeutet in diesem Kontext, dass  $\mathbf{X}$  den Ausdruck

$$\sum_{i=1}^n \sum_{j=1}^n C_{ij} x_{ij} \quad (6-9)$$

minimiert, wobei gilt

$$x_{ij} = \begin{cases} 1 & \text{wenn } e_i \text{ } m_j \text{ zugewiesen wird} \\ 0 & \text{sonst} \end{cases} \quad (6-10)$$

Das Resultat des Zuweisungsprozesses für das Beispiel aus Bild 47 ist dort mit „Zuweisung 1“ betitelt. Es lässt sich erkennen, dass jedes Weltobjekt ein Visionobjekt zugewiesen bekommt. Somit ist es möglich, die a posteriori-Zustandsschätzungen für alle Weltobjekte zu errechnen. Ein Visionobjekt lässt sich in dem gezeigten Beispiel keinem Weltobjekt zuweisen. Wie in 6.1.2 bzw. im zugehörigen UML-Aktivitätsdiagramm Bild 37 beschrieben, hat dies die Erzeugung eines neuen Hindernisobjekts zur Folge. Ebenso zeigt Bild 47 (Zuweisung 1) einen Nachteil der beschriebenen Methode. Obwohl Weltobjekt 1 und Visionobjekt 4 weit auseinander liegen, werden diese einander zugewiesen. Dies lässt sich durch folgende Modifikation von  $\mathbf{C}_0$  verhindern:

Überschreiten alle Zahlenwerte einer Spalte oder Zeile von  $\mathbf{C}_0$  einen vorgegebenen Schwellwert  $t$ , wird diese aus  $\mathbf{C}_0$  entfernt, bevor  $\mathbf{C}$  entsprechend (6-8) gebildet wird. Auf



diese Weise lassen sich Objekte vom Zuweisungsprozess ausschließen, die von allen zuweisbaren Objekten weiter als  $t$  entfernt sind. Zusätzlich ist es notwendig, alle übrigen Einträge von  $\mathbf{C}_0$ , die diesen Schwellwert überschreiten, durch den gleichen Zahlenwert von mindestens

$$d_t = \min(n_e, n_m) \cdot t \quad (6-11)$$

zu ersetzen. Dabei bezeichnet  $d_t$  die maximal möglichen Gesamtkosten des gesamten Zuweisungsprozesses, wenn ausschließlich Zuweisungen mit individuellen Kosten von  $d_{ij} \leq t$  erfolgen. Aus dem Ergebnis des Zuweisungsprozesses müssen schließlich alle Paarungen mit individuellen Kosten  $d_{ij} > t$  entfernt werden. Die übrigen Zuweisungen bilden das optimale Ergebnis unter den gegebenen Voraussetzungen. Im Beispiel aus Bild 47 ist das resultierende Ergebnis mit „Zuweisung 2“ bezeichnet. Hier bekommt das Weltobjekt 1 kein Visionobjekt zugewiesen und wird ggf. nach weiteren neun Zeitschritten gelöscht (vgl. UML-Diagramme Bild 37 und Bild 39). Zusätzlich entsteht ein neues Hindernisobjekt an der Stelle von Visionobjekt 4.

## 6.4 Zustandsschätzung von Objekten bei totzeitbehaftetem Messsignal

Zur Errechnung der Zustandsschätzung von Fahrzeug- und Hindernisobjekten (vgl. 6.1) werden zunächst die Kameradaten verwendet. Jedoch ist die Erfassung der Pose eines Objektes aufgrund des Rauschens der Bilddaten und auftretender Fehler im Bildverarbeitungsprozess (vgl. 6.2) ungenau. Die entsprechend Bild 34 von der Bildverarbeitung bereitgestellten Visionobjekte werden daher nicht direkt als Pose der Hindernisobjekte und Fahrzeugobjekte im Weltmodell übernommen. Stattdessen erfolgt für beide Weltobjekttypen eine Poseschätzung mit Hilfe Bayesscher Filter. Hinsichtlich der Auswahl einer effizienten Implementierung eines Bayesschen Filters ist es wichtig, dass sich der Zustandsvektor eines Objekts relativ gut durch eine normalverteilte Dichtefunktion (unimodal) über dem Zustandsraum darstellen lässt. Ebenso werden hohe Anforderungen an die Effizienz des Filters gestellt, da für alle Fahrzeug- und Hindernisobjekte gleichzeitig eine Zustandsschätzung errechnet werden muss. Entsprechend den Überlegungen aus 5.2.6 und der Übersicht in Tabelle 12 kommen daher Kalmanfilter zur Berechnung der Zustandsschätzungen zum Einsatz [20].

### 6.4.1 Das Kalmanfilter

Nachdem R.E. Kalman 1960 in seiner Arbeit [53] ein rekursives, diskretes, lineares Filter zur Zustandsschätzung beschrieben hat, ist es vielfach in der Literatur diskutiert und technisch umgesetzt worden. So ist der Algorithmus z. B. in der Saturn-V-Rakete des Apollo-Programms zum Einsatz gekommen [14], wird zur Parameterschätzung bei Bildverarbeitungsprozessen [24] oder zur Erfassung des Ladezustands von Li-Ion-Batterien [30] verwendet.

In der hier zunächst betrachteten, linearen, zeitinvarianten, zeitdiskreten Form basiert das Filter auf einer zeitdiskreten Zustandsraumdarstellung eines rauschbehafteten, linearen, zeitinvarianten (linear time-invariant, LTI) Systems.

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{w}_k \quad (6-12)$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{v}_k \quad (6-13)$$

Mit

$\mathbf{x} \in \mathfrak{R}^n$	Zustandsvektor
$\mathbf{y} \in \mathfrak{R}^l$	Messvektor
$\mathbf{u} \in \mathfrak{R}^r$	Eingangsvektor
$\mathbf{A} \in \mathfrak{R}^{n \times n}$	Systemmatrix
$\mathbf{B} \in \mathfrak{R}^{n \times r}$	Eingangsmatrix
$\mathbf{C} \in \mathfrak{R}^{l \times n}$	Ausgangsmatrix
$\mathbf{w} \in \mathfrak{R}^n$	Eingangsrauschvektor
$\mathbf{v} \in \mathfrak{R}^l$	Messrauschvektor

Eine notwendige Bedingung für den Einsatz eines Kalmanfilters ist, dass für die stochastischen Eigenschaften des Eingangs- und Messrauschens folgendes gilt:

$$E\{\mathbf{w}_i \mathbf{w}'_k\} = \begin{cases} \mathbf{Q}, & i = k \\ 0, & i \neq k \end{cases} \quad (6-14)$$

$$E\{\mathbf{v}_i \mathbf{v}'_k\} = \begin{cases} \mathbf{R}, & i = k \\ 0, & i \neq k \end{cases} \quad (6-15)$$

$$E\{\mathbf{w}_i \mathbf{v}'_k\} = 0 \quad \forall i, k \quad (6-16)$$

Die Rauschsignale  $\mathbf{w}$  und  $\mathbf{v}$ , die auf das System wirken, werden nach (6-14), (6-15) und (6-16) als weiß, mittelwertfrei und unkorreliert vorausgesetzt. Sie werden hier als stationär angenommen, womit die  $n \times n$  Matrix  $\mathbf{Q}$  und die  $l \times l$  Matrix  $\mathbf{R}$  zeitlich unveränderlich sind.

Die fünf in der Literatur angegebenen Gleichungen des diskreten Kalmanfilters lauten für den Spezialfall von zeitinvarianten Systemen:

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{C}' (\mathbf{C} \mathbf{P}_k^- \mathbf{C}' + \mathbf{R})^{-1} \quad (6-17)$$

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{y}_k - \mathbf{C} \hat{\mathbf{x}}_k^-) \quad (6-18)$$

$$\mathbf{P}_k^+ = (\mathbf{I} - \mathbf{K}_k \mathbf{C}) \mathbf{P}_k^- \quad (6-19)$$

$$\hat{\mathbf{x}}_{k+1}^- = \mathbf{A}\hat{\mathbf{x}}_k^+ + \mathbf{B}\mathbf{u}_k \quad (6-20)$$

$$\mathbf{P}_{k+1}^- = \mathbf{A}\mathbf{P}_k^+ \mathbf{A}' + \mathbf{Q} \quad (6-21)$$

wobei die Reihenfolge der genannten Gleichungen in der Literatur verschieden ist. Häufig wird die a priori-Schätzung (6-20) und (6-21) als erster Schritt genannt. In dieser Arbeit werden zuerst die a posteriori-Schätzung  $\hat{\mathbf{x}}_k^+$  sowie die zugehörige Kovarianzmatrix  $\mathbf{P}_k^+$  nach (6-17), (6-18) und (6-19) genannt und dann die a priori-Schätzung  $\hat{\mathbf{x}}_{k+1}^-$  inklusive Kovarianz  $\mathbf{P}_{k+1}^-$  für den folgenden Zeitschritt. Zur mathematischen Herleitung des Kalmanfilters sei an dieser Stelle auf die Literatur verwiesen [49], [120].

Bei dem in dieser Arbeit beschriebenen VTS wird das Messsignal (Kamerabild)  $\mathbf{y}_{k-d}$  zwar exakt zum Zeitpunkt  $k-d$  aufgenommen, jedoch durch Prozesse der Datenübertragung und Datenverarbeitung um  $d$  Zeitschritte versetzt dem Kalmanfilter zugeführt. Zusätzlich ist es aufgrund der niedrigen Abtastrate der Kameras möglich, die a priori-Schätzung mit Hilfe der Bewegungsgleichungen (6-3) bzw. (6-5) häufiger zu errechnen als die a posteriori-Schätzung. Daher wird in den folgenden Abschnitten ein Algorithmus entwickelt, der eine korrekte Lösung zur Zustandsschätzung von LTI-Systemen mit Kalmanfilter darstellt, wobei das Messsignal totzeitbehaftet sein kann und die Abtastzeiten von Eingangs- und Messsignal verschieden sein können. Es gilt lediglich die Bedingung, dass sowohl die Totzeit als auch die Abtastzeit des Messsignals ein ganzzahliges Vielfaches von der Zykluszeit der a priori-Schätzung betragen müssen. Während die Lösung für LTI-Systeme anwendbar für die lineare Bewegungsgleichung von Hindernisobjekten (vgl. (6-3)) ist, wird in 6.4.3 eine spezielle Lösung für das in (6-5) beschriebene, nichtlineare System aufgestellt, um auch für Fahrzeugobjekte eine optimale Zustandsschätzung ermitteln zu können.

## 6.4.2 Kalmanfilter für Hindernisobjekte

Ausgangspunkt dieses Kapitels ist ein Messsignal  $\mathbf{y}$ , welches einer Totzeit von  $d$  Zeitschritten unterliegt. Dabei muss  $d$  kein fester Zahlenwert sein, sondern kann in jedem Abtastschritt variieren. Da es jedoch unbefriedigend ist, bei großen Totzeiten lange auf die Errechnung der a posteriori-Schätzung zu warten, wird, solange das Messsignal  $\mathbf{y}_{k-d}$  nicht bereitsteht, mit einer a priori-Schätzung zum Zeitpunkt  $k-d$ ,  $\hat{\mathbf{x}}_{k-d}^-$  mit der Kovarianzmatrix  $\mathbf{P}_{k-d}^-$  gearbeitet. Wenn das Messsignal bereitgestellt ist, kann einerseits die a priori-Schätzung  $\hat{\mathbf{x}}_{k-d}^-$  nachträglich zur a posteriori-Schätzung  $\hat{\mathbf{x}}_{k-d}^+$  aktualisiert und andererseits die aktuelle a priori-Schätzung  $\hat{\mathbf{x}}_k^-$  zur Schätzung  $\hat{\mathbf{x}}_k^+$  korrigiert werden. Gleiches gilt für die Korrektur der zugehörigen Kovarianzmatrix von  $\mathbf{P}_k^-$  zu  $\mathbf{P}_k^+$ .

Die a priori-Schätzung  $\hat{\mathbf{x}}_k^-$  lässt sich als Funktion der  $d$  Abtastschritte zurückliegenden Schätzung  $\hat{\mathbf{x}}_{k-d}^-$  entsprechend (6-20) folgendermaßen angeben:

$$\hat{\mathbf{x}}_k^- = \mathbf{A}^d \hat{\mathbf{x}}_{k-d}^- + \Delta \hat{\mathbf{u}}_{k,d} \quad (6-22)$$

mit

$$\Delta \hat{\mathbf{u}}_{k,d} = \mathbf{A}^{d-1} \mathbf{B} \mathbf{u}_{k-d} + \mathbf{A}^{d-2} \mathbf{B} \mathbf{u}_{k-d+1} + \dots + \mathbf{A} \mathbf{B} \mathbf{u}_{k-2} + \mathbf{B} \mathbf{u}_{k-1}. \quad (6-23)$$

Die a priori-Schätzung  $\hat{\mathbf{x}}_k^-$  resultiert aus  $d$  aufeinander folgenden a priori-Schätzungen basierend auf der a priori-Zustandsschätzung zum Zeitpunkt  $k-d$ . Wobei der mit  $\Delta \hat{\mathbf{u}}_{k,d}$  bezeichnete Teil in (6-22) lediglich abhängig vom Eingangssignal  $\mathbf{u}$  und nicht abhängig vom Zustand  $\mathbf{x}$  ist.

Es ergibt sich die durch das Eingangssignal  $\mathbf{u}$  bewirkte Änderung der Zustandsschätzung,  $\Delta \hat{\mathbf{u}}_{k,d}$ :

$$\Delta \hat{\mathbf{u}}_{k,d} = \hat{\mathbf{x}}_k^- - \mathbf{A}^d \hat{\mathbf{x}}_{k-d}^-. \quad (6-24)$$

Wird in (6-22) eine a posteriori-Schätzung zum Zeitpunkt  $k-d$  eingesetzt, ergibt sich die korrigierte aktuelle Schätzung

$$\hat{\mathbf{x}}_k^* = \mathbf{A}^d \hat{\mathbf{x}}_{k-d}^+ + \Delta \hat{\mathbf{u}}_{k,d} \quad (6-25)$$

und mit (6-24)

$$\hat{\mathbf{x}}_k^* = \hat{\mathbf{x}}_k^- + \mathbf{A}^d (\hat{\mathbf{x}}_{k-d}^+ - \hat{\mathbf{x}}_{k-d}^-). \quad (6-26)$$

Somit lässt sich ein Vorgehen in zwei Schritten zur Korrektur der aktuellen a priori-Schätzung zum Zeitpunkt  $k$  durch eine um  $d$  Abtastschritte zurückliegende a posteriori-Schätzung angeben:

- 1) Errechnung des a posteriori-Schätzwertes  $\hat{\mathbf{x}}_{k-d}^+$  vom zurückliegenden Zeitpunkt  $k-d$  mit dem gespeicherten a priori-Schätzwert  $\hat{\mathbf{x}}_{k-d}^-$  und der gespeicherten Kovarianzmatrix  $\mathbf{P}_{k-d}^-$  sowie dem bereitstehenden Messsignal  $\mathbf{y}_{k-d}$  auf Basis von (6-18).
- 2) Errechnung der neuen a priori-Schätzung zum aktuellen Zeitpunkt  $k$ ,  $\hat{\mathbf{x}}_k^*$ , mit Hilfe von (6-26).

Dazu müssen zum Zeitpunkt  $k-d$  der Messung  $\mathbf{y}_{k-d}$  sowohl die Zustandsschätzung  $\hat{\mathbf{x}}_{k-d}^-$  als auch die zugehörige Kovarianzmatrix  $\mathbf{P}_{k-d}^-$  gespeichert worden sein.

Analog zur Aktualisierung der Zustandsschätzung lässt sich auch die aktualisierte Kovarianzmatrix  $\mathbf{P}_k^*$  ermitteln. Aus (6-21) resultiert für  $d$ -maliges Ausführen die a priori-Kovarianzmatrix  $\mathbf{P}_k^-$  zum Zeitpunkt  $k$  auf Basis der a priori-Kovarianzmatrix  $\mathbf{P}_{k-d}^-$  zum Zeitpunkt  $k-d$ ,

$$\mathbf{P}_k^- = \mathbf{A}^d \mathbf{P}_{k-d}^- \mathbf{A}^{td} + \Delta \mathbf{O}_{k,d} \quad (6-27)$$

mit

$$\Delta \mathbf{O}_{k,d} = \mathbf{A}^{d-1} \mathbf{Q} \mathbf{A}^{d-1} + \mathbf{A}^{d-2} \mathbf{Q} \mathbf{A}^{d-2} + \dots + \mathbf{A} \mathbf{Q} \mathbf{A} + \mathbf{Q}. \quad (6-28)$$

Umstellen von (6-27) ergibt die durch das Eingangsrauschen bewirkte Änderung  $\Delta \mathbf{O}_{k,d}$  der Kovarianzmatrix seit der um  $d$  Schritte zurückliegend berechneten Kovarianz  $\mathbf{P}_{k-d}^-$ ,

$$\Delta \mathbf{O}_{k,d} = \mathbf{P}_k^- - \mathbf{A}^d \mathbf{P}_{k-d}^- \mathbf{A}^{d^T}. \quad (6-29)$$

Entsprechend dem Vorgehen zur Errechnung der aktualisierten Zustandsschätzung  $\hat{\mathbf{x}}_k^*$  kann die aktualisierte Kovarianzmatrix  $\mathbf{P}_k^*$  ermittelt werden, indem in (6-27) ein aktualisierter Wert der Kovarianzmatrix zum Zeitpunkt  $k-d$ ,  $\mathbf{P}_{k-d}^+$  eingesetzt wird,

$$\mathbf{P}_k^* = \mathbf{A}^d \mathbf{P}_{k-d}^+ \mathbf{A}^{d^T} + \Delta \mathbf{O}_{k,d}. \quad (6-30)$$

Mit (6-29) ergibt sich

$$\mathbf{P}_k^* = \mathbf{P}_k^- + \mathbf{A}^d (\mathbf{P}_{k-d}^+ - \mathbf{P}_{k-d}^-) \mathbf{A}^{d^T}. \quad (6-31)$$

Demnach lässt sich die Aktualisierung der Kovarianzmatrix  $\mathbf{P}_k^-$  in weiteren zwei Schritten zusammenfassen:

- 3) Errechnung der neuen a posteriori-Kovarianz  $\mathbf{P}_{k-d}^+$  zum zurückliegenden Zeitpunkt  $k-d$  mit der gespeicherten Kovarianzmatrix  $\mathbf{P}_{k-d}^-$  auf Basis von (6-19).
- 4) Errechnung der neuen a priori-Kovarianzmatrix  $\mathbf{P}_k^*$  zum aktuellen Zeitpunkt  $k$  mit Hilfe von (6-31).

Im Spezialfall, dass die Totzeit des Messsignals von  $d$  Zeitschritten größer ist als die Abtastzeit des Messsignals (und damit der Zykluszeit der a posteriori-Schätzung) von  $s$  Zeitschritten, müssen beim Eintreffen eines Messsignals alle zwischen Zeitpunkt  $k-d$  und dem aktuellen Zeitpunkt  $k$  abgespeicherten Werte für die Zustandsschätzung  $\hat{\mathbf{x}}^-$  und die Kovarianzmatrix  $\mathbf{P}^-$  aktualisiert werden. Diese Aktualisierung erfolgt laut (6-26) und (6-31) durch Korrektur aller abgespeicherten Zustandsschätzungen  $\hat{\mathbf{x}}_{k-d+n}^-$  und Kovarianzmatrizen  $\mathbf{P}_{k-d+n}^-$ , die  $n$  Zeitschritte nach dem Zeitpunkt  $k-d$  abgespeichert wurden, entsprechend

$$\hat{\mathbf{x}}_{k-d+n}^* = \hat{\mathbf{x}}_{k-d+n}^- + \mathbf{A}^n (\hat{\mathbf{x}}_{k-d}^+ - \hat{\mathbf{x}}_{k-d}^-) \quad \forall n = s, 2s, 3s, \dots < d \quad (6-32)$$

$$\mathbf{P}_{k-d+n}^* = \mathbf{P}_{k-d+n}^- + \mathbf{A}^n (\mathbf{P}_{k-d}^+ - \mathbf{P}_{k-d}^-) \mathbf{A}^{n^T} \quad \forall n = s, 2s, 3s, \dots < d. \quad (6-33)$$

In Bild 48 ist beispielhaft der zeitliche Ablauf des beschriebenen Verfahrens zur Zustandsschätzung dargestellt. Die Totzeit ist mit  $d = 5$  Zeitschritten größer als die



(6-35) dabei in jedem Zeitschritt um den aktuellen Arbeitspunkt linearisiert werden. Damit lässt sich das in (6-17) bis (6-21) beschriebene, lineare Kalmanfilter auch für den nichtlinearen Fall anwenden. Jedoch werden die Matrizen  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$ ,  $\mathbf{Q}$  und  $\mathbf{R}$  durch die zu jedem Zeitpunkt notwendige Linearisierung abhängig von dem aktuell geschätzten Zustand. Somit sind die in (6-23) und (6-28) beschriebenen Teile  $\Delta \hat{\mathbf{u}}_{k,d}$  und  $\Delta \mathbf{O}_{k,d}$  nicht mehr nur abhängig vom Eingangssignal  $\mathbf{u}$  bzw. der Kovarianz des Eingangsrauschens  $\mathbf{Q}$ , sondern auch vom Zustand  $\mathbf{x}$ . Damit lässt sich die im linearen Fall vorgestellte Methode zur effizienten, nachträglichen Korrektur der aktuellen Zustandsschätzung nicht allgemein auf den nichtlinearen Fall übertragen.

Bei der Zustandsschätzung für die in 6.1.3 beschriebenen Fahrzeugobjekte liefern fahrzeuginterne Sensoren in jedem Zeitschritt die Änderung der aktuellen Position in  $x'$ - und  $y'$ -Richtung,  $\Delta x'$  und  $\Delta y'$  und gleichzeitig die Änderung der Orientierung  $\Delta \varphi'$  (vgl. (6-5)). Die Pose eines Fahrzeugobjekts im globalen, ortsfesten Koordinatensystem wird allgemein mit der folgenden Systemgleichung

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}_k\mathbf{u}_k + \mathbf{w}_k \quad (6-36)$$

beschrieben. Die Messgleichung lautet

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{v}_k. \quad (6-37)$$

Das Eingangsrauschen  $\mathbf{w}$  und das Messrauschen  $\mathbf{v}$  werden, wie in (6-14) bis (6-16) beschrieben, als mittelwertfrei, weiß und unkorreliert vorausgesetzt.

Während der Zustandsvektor durch (6-4) gegeben ist, lautet entsprechend (6-5) die (lineare) Systemmatrix

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (6-38)$$

die von der Orientierung  $\varphi$  abhängige nichtlineare Eingangsmatrix

$$\mathbf{B}_k = \begin{bmatrix} \cos(\varphi_k) & -\sin(\varphi_k) & 0 \\ \sin(\varphi_k) & \cos(\varphi_k) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (6-39)$$

und das Eingangssignal

$$\mathbf{u}_k = \begin{bmatrix} \Delta x'_k \\ \Delta y'_k \\ \Delta \varphi'_k \end{bmatrix}. \quad (6-40)$$

Da lediglich die Position und nicht die Orientierung von den Deckenkameras erfasst wird, lautet die Messmatrix

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}. \quad (6-41)$$

Die Zustandsschätzung wird auf Basis der Systemgleichung (6-36) durchgeführt. Dazu wird in  $\mathbf{B}_k$  der aktuelle Schätzwert für die Orientierung,  $\hat{\varphi}_k^-$ , eingesetzt. Diese Matrix wird im Folgenden mit  $\hat{\mathbf{B}}_k^-$  bezeichnet. Für den Zustand  $\mathbf{x}_k$  wird die Schätzung  $\hat{\mathbf{x}}_k^-$  verwendet und das Eingangsrauschen  $\mathbf{w}_k$  wird naturgemäß in der Schätzgleichung vernachlässigt.

$$\hat{\mathbf{x}}_{k+1}^- = \mathbf{A}\hat{\mathbf{x}}_k^- + \hat{\mathbf{B}}_k^- \mathbf{u}_k \quad (6-42)$$

Wie im linearen Fall soll auch hier ein totzeitbehaftetes Messsignal betrachtet werden. Die Position des Fahrzeugs wird mittels Deckenkamera erfasst und unterliegt durch aufwändige Bildverarbeitungsprozesse einer Totzeit von  $d$  Zeitschritten. Wie zu Beginn dieses Abschnittes beschrieben, lässt sich die auf einem eingetroffenen Messsignal basierende aktuelle Schätzung der Pose zum Zeitpunkt  $k$  nicht entsprechend (6-26) und deren Kovarianz nicht mit (6-31) errechnen. Eine Möglichkeit, dieses Problem zu lösen, wäre, alle Eingangswerte  $\mathbf{u}$  seit dem Zeitpunkt  $k-d$  der Messung abzuspeichern und beim Eintreffen des Messsignals alle Zustände bis zur aktuellen Zustandsschätzung neu zu berechnen. Für den hier beschriebenen Spezialfall lässt sich jedoch eine effizientere, empirische Lösung angeben.

Ein Fehler in der Zustandsschätzung zum Zeitpunkt der Messung  $k-d$  hat eine falsche Position und Orientierung der im Folgenden geschätzten, vom Fahrzeug zurückgelegten Trajektorie zur Folge. Die Form der geschätzten Trajektorie ist aber unabhängig davon. Daher kann bei der Bereitstellung der korrigierten Zustandsschätzung zum Zeitpunkt  $k-d$  die korrekte Trajektorie durch Verschieben und Verdrehen der fehlerhaften Trajektorie ermittelt werden. Zusätzlich ist die exakte Form der Trajektorie zur Korrektur der aktuellen Zustandsschätzung  $\hat{\mathbf{x}}_k^-$  bedeutungslos. Lediglich die Änderung der Fahrzeugpose vom Messzeitpunkt  $k-d$  bis zum aktuellen Zeitpunkt  $k$

$$\Delta \hat{\mathbf{x}}_{k,d} = (\hat{\mathbf{x}}_k^- - \hat{\mathbf{x}}_{k-d}^-) \quad (6-43)$$

ist zur Ermittlung der korrigierten, aktuellen Pose essentiell. Diese Änderung der Pose ist unabhängig vom Zustandsvektor und nur abhängig vom Eingangssignal  $\mathbf{u}$ . Somit



lässt sich die aktuelle, korrigierte Zustandsschätzung  $\hat{\mathbf{x}}_k^*$  auf Basis der korrigierten Zustandsschätzung  $\hat{\mathbf{x}}_{k-d}^+$  und der Addition der Änderung der Fahrzeugpose errechnen. Hierbei ist wichtig, dass die Poseänderung  $\Delta\hat{\mathbf{x}}_{k,d}$  in Richtung der korrigierten Orientierung des Fahrzeugs zum Zeitpunkt  $k-d$  erfolgt. Sie muss daher um die Differenz der Orientierung zwischen korrigierter und fehlerhafter Pose zum Zeitpunkt  $k-d$  gedreht werden. Mit der Differenz der Orientierung

$$\Delta\hat{\phi} = (\hat{\phi}_{k-d}^+ - \hat{\phi}_{k-d}^-) \quad (6-44)$$

ergibt sich die korrigierte Zustandsschätzung

$$\hat{\mathbf{x}}_k^* = \hat{\mathbf{x}}_{k-d}^+ + \begin{bmatrix} \cos(\Delta\hat{\phi}) & -\sin(\Delta\hat{\phi}) & 0 \\ \sin(\Delta\hat{\phi}) & \cos(\Delta\hat{\phi}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \Delta\hat{\mathbf{x}}_{k,d}. \quad (6-45)$$

Die zugehörige Korrektur der Kovarianzmatrix  $\mathbf{P}$  wird für den hier betrachteten Spezialfall im Folgenden hergeleitet. Nach [120] lässt sich die Kovarianzmatrix der a priori-Zustandsschätzung im erweiterten Kalmanfilter folgendermaßen darstellen

$$\mathbf{P}_{k+1}^- = \mathbf{A}\mathbf{P}_k^+ \mathbf{A}' + \mathbf{W}_k \mathbf{Q} \mathbf{W}_k', \quad (6-46)$$

wobei (6-46) bereits den Randbedingungen des hier betrachteten Spezialfalls Rechnung trägt, dass die Systemmatrix  $\mathbf{A}$  zeitlich invariant ist. Ebenso ist das Rauschsignal stationär und damit  $\mathbf{Q}$  konstant. Lediglich die Matrix  $\mathbf{B}$ , über die das Eingangssignal auf den Systemzustand wirkt, hängt vom Zustand  $\mathbf{x}$  ab, was sich in der Zeitabhängigkeit von  $\mathbf{W}$  ausdrückt.

Zur Herleitung der Korrektur des Zustandsvektors (6-45) diene die Idee, die Trajektorie des Fahrzeugs um einen Korrekturwinkel  $\Delta\hat{\phi}$  zu drehen. Zugrunde gelegt bei dieser Überlegung wurde, dass die Form der Trajektorie unverändert bleibt. Daher muss, neben dem Einfluss des Eingangsvektors  $\mathbf{u}$  auch der angenommene Rauschvektor um den betreffenden Winkel verdreht werden, um die Kovarianzschätzung zu korrigieren. Ausgangspunkt ist also ein Propagationsschritt der verdrehten (korrigierten) Trajektorie eines VTF, basierend auf dessen Bewegungsgleichung (6-5):

$$\begin{aligned} \mathbf{x}_{k+1}^* = & \mathbf{x}_k^* + \begin{bmatrix} \cos(\Delta\hat{\phi}) & -\sin(\Delta\hat{\phi}) & 0 \\ \sin(\Delta\hat{\phi}) & \cos(\Delta\hat{\phi}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \dots \\ & \dots \left( \begin{bmatrix} \cos(\varphi_k + w_{\varphi,k}) & -\sin(\varphi_k + w_{\varphi,k}) & 0 \\ \sin(\varphi_k + w_{\varphi,k}) & \cos(\varphi_k + w_{\varphi,k}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta x'_k \\ \Delta y'_k \\ \Delta \phi'_k \end{bmatrix} + \begin{bmatrix} w_{x,k} \\ w_{y,k} \\ w_{\varphi,k} \end{bmatrix} \right). \end{aligned} \quad (6-47)$$

Mit der partiellen Ableitung der nichtlinearen Systemfunktion nach den Rauschvariablen

$$\mathbf{W}_{k,[i,j]} = \frac{\partial f(\mathbf{x}_k; \mathbf{u}_k; \mathbf{w}_k)_{[i]}}{\partial \mathbf{w}_{k,[j]}} \bigg|_{\mathbf{x}_k = \hat{\mathbf{x}}_k, \mathbf{u}_k = \mathbf{u}_k, \mathbf{w}_k = 0} = \frac{\partial \mathbf{x}_{k+1}^*_{[i]}}{\partial \mathbf{w}_{k,[j]}} \bigg|_{\mathbf{x}_k = \hat{\mathbf{x}}_k, \mathbf{u}_k = \mathbf{u}_k, \mathbf{w}_k = 0} \quad (6-48)$$

folgt

$$\mathbf{W}_k^* = \begin{bmatrix} \cos(\Delta\hat{\phi}) & -\sin(\Delta\hat{\phi}) & 0 \\ \sin(\Delta\hat{\phi}) & \cos(\Delta\hat{\phi}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -\sin(\hat{\phi}_k) \cdot \Delta x'_k - \cos(\hat{\phi}_k) \cdot \Delta y'_k \\ 0 & 1 & \cos(\hat{\phi}_k) \cdot \Delta x'_k - \sin(\hat{\phi}_k) \cdot \Delta y'_k \\ 0 & 0 & 1 \end{bmatrix} \quad (6-49)$$

$$= \Delta \mathbf{W} \mathbf{W}_k^-$$

Dabei beschreibt (6-49) bereits, wie sich die korrigierte Matrix für die rotierte Trajektorie  $\mathbf{W}_k^*$  aus der ursprünglichen Matrix für die unverdrehte Trajektorie  $\mathbf{W}_k^-$  ableiten lässt.

Analog zu (6-27) für den linearen Fall, gibt (6-50) die Entwicklung von  $\mathbf{P}$  über  $d$  zurückliegende Zeitschritte im nichtlinearen Fall an

$$\mathbf{P}_k^- = \mathbf{A}^d \mathbf{P}_{k-d}^- \mathbf{A}^{d'} + \Delta \mathbf{P}_{k,d}^- \quad (6-50)$$

Dabei gilt

$$\Delta \mathbf{P}_{k,d}^- = \mathbf{A}^{d-1} \mathbf{W}_{k-d}^- \mathbf{Q} \mathbf{W}_{k-d}^{-'} \mathbf{A}^{d-1'} + \dots + \mathbf{A} \mathbf{W}_{k-2}^- \mathbf{Q} \mathbf{W}_{k-2}^{-'} \mathbf{A}' + \mathbf{W}_{k-1}^- \mathbf{Q} \mathbf{W}_{k-1}^{-'} \quad (6-51)$$

Hier ist zu beachten, dass die Systemmatrix  $\mathbf{A}$  im Gegensatz zum allgemeinen nichtlinearen Fall als Einheitsmatrix und konstant vorausgesetzt wird. Damit ergibt sich aus (6-51)

$$\Delta \mathbf{P}_{k,d}^- = \mathbf{W}_{k-d}^- \mathbf{Q} \mathbf{W}_{k-d}^{-'} + \dots + \mathbf{W}_{k-2}^- \mathbf{Q} \mathbf{W}_{k-2}^{-'} + \mathbf{W}_{k-1}^- \mathbf{Q} \mathbf{W}_{k-1}^{-'} \quad (6-52)$$

und aus (6-50) folgt

$$\mathbf{P}_k^- = \mathbf{P}_{k-d}^- + \Delta \mathbf{P}_{k,d}^- \quad (6-53)$$

Wird zur Errechnung der aktuellen Kovarianzmatrix von einem korrigierten Wert des Zustandsvektors  $\hat{\mathbf{x}}_{k-d}^+$  und der Kovarianzmatrix  $\mathbf{P}_{k-d}^+$  ausgegangen, ergibt sich die korrigierte, aktuelle Kovarianzmatrix gemäß

$$\mathbf{P}_k^+ = \mathbf{P}_{k-d}^+ + \Delta \mathbf{P}_{k,d}^+ \quad (6-54)$$

mit

$$\Delta \mathbf{P}_{k,d}^* = \mathbf{W}_{k-d}^* \mathbf{Q} \mathbf{W}_{k-d}^{*'} + \mathbf{W}_{k-d+1}^* \mathbf{Q} \mathbf{W}_{k-d+1}^{*'} + \dots + \mathbf{W}_{k-2}^* \mathbf{Q} \mathbf{W}_{k-2}^{*'} + \mathbf{W}_{k-1}^* \mathbf{Q} \mathbf{W}_{k-1}^{*'} . \quad (6-55)$$

Mit (6-49) lässt sich (6-55) umschreiben zu

$$\begin{aligned} \Delta \mathbf{P}_{k,d}^* &= \Delta \mathbf{W} \mathbf{W}_{k-d}^- \mathbf{Q} \mathbf{W}_{k-d}^{-'} \Delta \mathbf{W}' + \Delta \mathbf{W} \mathbf{W}_{k-d+1}^- \mathbf{Q} \mathbf{W}_{k-d+1}^{-'} \Delta \mathbf{W}' + \dots \\ &\quad \dots + \Delta \mathbf{W} \mathbf{W}_{k-1}^- \mathbf{Q} \mathbf{W}_{k-1}^{-'} \Delta \mathbf{W}' \\ &= \Delta \mathbf{W} \Delta \mathbf{P}_{k,d}^- \Delta \mathbf{W}' \end{aligned} \quad (6-56)$$

und (6-54) ergibt

$$\mathbf{P}_k^* = \mathbf{P}_{k-d}^+ + \Delta \mathbf{W} \Delta \mathbf{P}_{k,d}^- \Delta \mathbf{W}' , \quad (6-57)$$

wobei wegen (6-53) gilt:

$$\Delta \mathbf{P}_{k,d}^- = \mathbf{P}_k^- - \mathbf{P}_{k-d}^- . \quad (6-58)$$

Ebenso lässt sich (6-45) folgendermaßen angeben:

$$\hat{\mathbf{x}}_k^* = \hat{\mathbf{x}}_{k-d}^+ + \Delta \mathbf{W} \Delta \hat{\mathbf{x}}_{k,d} . \quad (6-59)$$

Für das in (6-36) bis (6-41) beschriebene System mit einer Totzeit von  $d$  Abtastschritten im Messsignal lässt sich nach (6-59) und (6-57) eine korrigierte a priori-Schätzung zum Zeitpunkt  $k$  angeben. Um diese Gleichungen lösen zu können ist es notwendig, zum Zeitpunkt der Messung  $k-d$  sowohl die Zustandsschätzung  $\hat{\mathbf{x}}_{k-d}^-$  als auch die Kovarianzmatrix  $\mathbf{P}_{k-d}^-$  zu speichern. Ebenso wie im linearen Fall muss, falls die Totzeit  $d$  des Systems dessen Abtastzeit  $s$  übersteigt, jeder gespeicherte Wert beim Eintreffen neuer Messdaten korrigiert werden.

$$\hat{\mathbf{x}}_{k-d+n}^* = \hat{\mathbf{x}}_{k-d}^+ + \Delta \mathbf{W} \Delta \hat{\mathbf{x}}_{k-d+n,n} \quad \forall n = s, 2s, 3s, \dots < d \quad (6-60)$$

$$\mathbf{P}_{k-d+n}^* = \mathbf{P}_{k-d}^+ + \Delta \mathbf{W} \Delta \mathbf{P}_{k-d+n,n}^- \Delta \mathbf{W}' \quad \forall n = s, 2s, 3s, \dots < d \quad (6-61)$$

Der in Bild 48 dargestellte, zeitliche Ablauf der Zustandsschätzung mit totzeitbehaftetem Messsignal ist auch für den nichtlinearen Fall gültig. Es müssen lediglich andere Gleichungen Anwendung finden. Zum Update der aktuellen Schätzung kommen (6-57) und (6-59), zum Update der gespeicherten Werte (6-60) und (6-61) zur Verwendung.

Zur Errechnung der a posteriori-Schätzung im erweiterten Kalmanfilter sei auf die Literatur [120] verwiesen.

### 6.5 Graphische Benutzerschnittstelle

Zur Bedienung eines komplexen Systems wie des vorgestellten VTS ist der Einsatz einer grafischen Benutzerschnittstelle (graphical user interface, GUI) sinnvoll. Hier werden alle Informationen wie aktuelle Hintergrundschätzung, Kameradaten, Weltmodellzustand (mit allen Weltobjekten), angemeldete Fahrzeuge (IP-Adressen) usw. angezeigt. Außerdem lassen sich ausgewählte Parameter, z. B. für die Bildverarbeitung, verändern.

Bild 49 zeigt die GUI während des Betriebs des VTS. Über die Symbolleiste lassen sich Operationen ausführen wie:

- Starten der Aufnahme einer/mehrerer Kamerabilder (in Abhängigkeit vom Parameter „Stream“)
- Laden eines Hintergrundbilds von der Festplatte
- Speichern eines aktuellen Kamerabilds auf der Festplatte als Hintergrund
- Zurücksetzen des gesamten Weltmodells (Löschen aller Weltobjekte)
- Suche nach Zielmarkern

Um einzelne Softwarekomponenten hinsichtlich ihrer Funktion und Performance zu untersuchen, ist es sinnvoll, gewisse Prozesse an- und abzuschalten. Über die GUI lassen sich diesbezüglich folgende Einstellungen vornehmen:

- Verarbeitung erkannter Visionobjekte im Weltmodell ja / nein (Die Einstellung „nein“ macht dann Sinn, wenn ausschließlich das Ergebnis des Bildverarbeitungsprozesses in der GUI betrachtet werden soll.)
- Kontinuierliches Sampling ja / nein (Durch Betätigen des Symbols „Starte Bildaufnahme“ kann bei Einstellung „nein“ exakt ein Bild verarbeitet werden und dessen Ergebnis z. B. überprüft oder auf der Festplatte gespeichert werden.)
- Bildverarbeitung ja / nein (Auf diese Weise kann die Funktion der GUI überprüft werden, ohne rechenaufwändige Prozesse im Hintergrund.)
- Zeige Daten in GUI an ja / nein (Für den Betrieb des VTS ist die GUI nicht nötig. Um dann maximale Rechenleistung für Bildverarbeitung und Weltmodellschätzung zur Verfügung zu haben, lässt sich die Aktualisierung der GUI ausschalten.)

Schließlich stellt die GUI vier Fenster zur optischen Darstellung von Prozessergebnissen bereit:

- Kamerabild (letztes von der Deckenkamera aufgenommenes Bild)
- Hintergrundbild
- Umschließende Rechtecke aller Vordergrundobjekte (als Ergebnis des Bildverarbeitungsprozesses, bevor die Zielmarker entfernt werden)
- Zustandsschätzung von Weltobjekten (Hindernisobjekte werden durch ihren Mittelpunkt und die geschätzte Geschwindigkeit sowie ihre ID weiß, Zielobjekte

durch ihre Position, Orientierung und ID rot und Fahrzeugobjekte durch ihre Position, Orientierung und ID gelb dargestellt.)

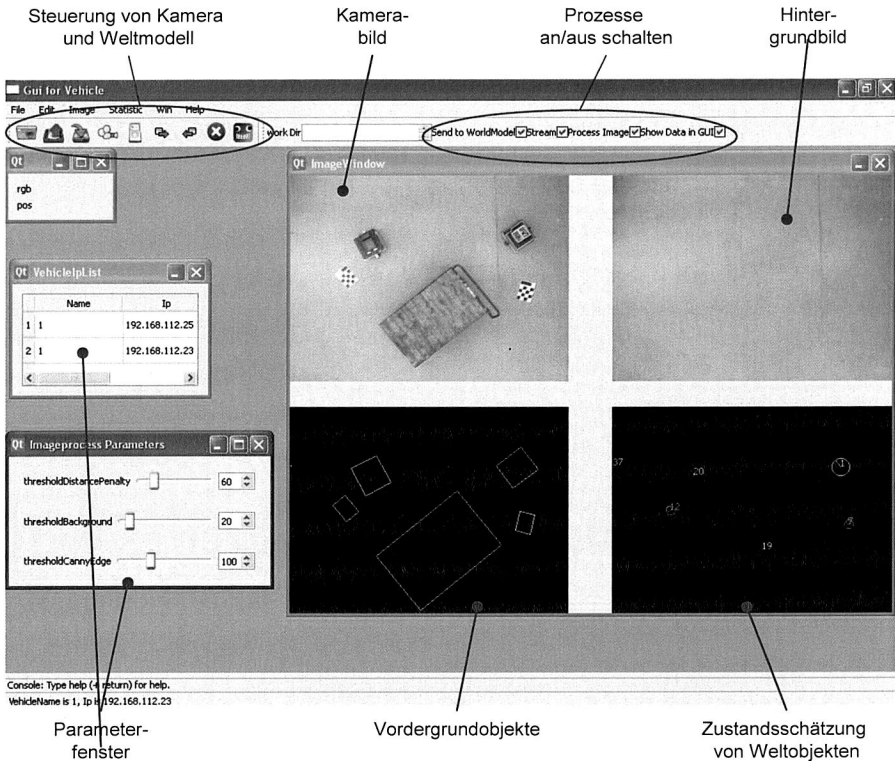


Bild 49: Screenshot der GUI während des Betriebs des VTS.

Neben den online über die GUI einstellbaren Parametern existiert eine Datei, in der alle Grundeinstellungen des Weltmodells hinterlegt sind. Dort erfolgen Einstellungen wie

- Samplefrequenz der Deckenkamera (und damit des Weltmodells)
- Breite und Höhe des von der Kamera erfassten Bereichs
- Auflösung, mit der das Kamerabild verarbeitet wird
- Parameter für die Bildverarbeitung (Schwellwert für die Hintergrundsubtraktion, Hintergrundangleichsrate...)
- Geometrische Parameter von Zielmarkern
- Einstellungen für inter-prozess Kommunikation
- Einstellungen für Debug-Messages
- Dateiname für Logfile
- ...

Die beschriebene GUI stellt ein leistungsfähiges Werkzeug zur Steuerung des VTS und zur Beobachtung ablaufender Prozesse wie Bildverarbeitung und Zustandsschätzung dar. Zusammenfassend umfasst das die optische Erkennung von Vordergrundobjekten mit Hilfe der Deckenkameras sowie deren mathematische Modellierung. Die optische Erkennung basiert auf einer Hintergrundsubtraktion mit anschließender Konturerkennung und Errechnung umschließender Rechtecke für alle Vordergrundobjekte. Während die Zielmarker mittels Mustererkennung anhand ihrer Schachbrettmuster ermittelt werden, dienen die verbleibenden Vordergrundobjekte, die Visionobjekte, als Grundlage für einen modellbasierten Multi-Object Tracking-Algorithmus. Hierzu erfolgt zunächst die mathematische Modellierung von Fahrzeug- oder Hindernisobjekten. Diese Modelle beinhalten neben den geschätzten Positionen weitere Zustandsvariablen wie geschätzte Geschwindigkeit oder geschätzte Orientierung der Objekte. Mit Hilfe dieser Zustandswerte ist es möglich, die Trajektorien der einzelnen Objekte aus einer Serie von Aufnahmen der Deckenkameras zu rekonstruieren. Hierzu erfolgt in jedem Zyklus die Zuweisung von Visionobjekten zu den jeweiligen Fahrzeug- und Hindernisobjekten mit Hilfe der Ungarischen Methode.

## 7 Die prototypische Realisierung des VTS zeigt dessen volle Funktionsfähigkeit

Die vorangegangenen Kapitel stellen eine konzeptionelle Beschreibung eines VTS dar. In diesem Abschnitt erfolgt zunächst eine detaillierte Betrachtung der Komponenten und der Konstruktion des für den Einsatz geplanten VTF. Es besitzt alle mechanischen Komponenten, die entsprechend 5.1 zum Betrieb benötigt werden. Darüber hinaus wird beschrieben, wie die prototypische Versuchsanlage mit mechanisch vereinfachten VTF realisiert wurde und mit Hilfe welcher Softwarekomponenten das beschriebene Konzept umgesetzt worden ist. Daran anschließend wird eine abschließende Diskussion der Performance, insbesondere hinsichtlich Genauigkeit der Positionsbestimmung der einzelnen VTF geführt und es werden weitere Handlungsbedarfe abgeleitet.

### 7.1 Detaillierte Konstruktion des VTF

Um ein VTF umzusetzen, ist es notwendig, dessen Komponenten anforderungsgemäß auszuwählen. Im Folgenden wird eine beispielhafte Realisierung eines VTF vorgestellt. Dabei wird auf Elektromotoren, Rechner und Sensoren eingegangen. Im Zusammenhang mit der ebenfalls beschriebenen Konstruktion ist eine präzise Abschätzung der zu erwartenden Kosten für ein VTF möglich.

#### *Elektromotoren*

Die nötige Antriebsleistung für ein VTF hängt stark von dessen Gewicht ab. Wie in 4.2 dargestellt, ist ein angenommenes Leergewicht von ca. 20 kg realistisch. Die benötigte Leistung beim Fahren in der Ebene wird abgeschätzt nach (7-1). Dabei wird eine Rollreibungszahl  $\mu_f$  von 0,01 (Gummirad auf Betonboden [104]) zugrunde gelegt und von einer Zuladung von 10 kg ausgegangen.

$$P_{Eben,v1} = v_1 \cdot F_N \cdot \mu_f = 1\text{m/s} \cdot 30\text{kg} \cdot 9,81\text{m/s}^2 \cdot 0,01 \approx 3\text{W} \quad (7-1)$$

Die benötigte mechanische Abgabeleistung bei Steigungsfahrten ist neben der Geschwindigkeit insbesondere von der zu überwindenden Steigung abhängig. Zugrunde gelegt wird hier eine angenommene Maximalsteigung von 20 %. Es ergibt sich die erforderliche Leistung aus der in (7-1) formulierten Leistung zur Überwindung der Rollreibung und der zusätzlich zu überwindenden Hangabtriebskraft

$$P_{Hang,v1} = v_1 \cdot F_N \cdot \sin(11,3^\circ) \approx 1\text{m/s} \cdot 30\text{kg} \cdot 9,81\text{m/s}^2 \cdot 0,2 \approx 59\text{W} \quad (7-2)$$

Entsprechend [107] werden Wirkungsgrade der Getriebe je nach Getriebeuntersetzung von ca. 60 % - 70 % angegeben. Bei einem angenommenen Wirkungsgrad von 65 % ergibt sich demnach eine geforderte maximale Motorleistung von ca. 100 W bei einer Fahrgeschwindigkeit von 1 m/s.

Diese mechanische Leistung müssen die beiden Elektromotoren (pro Motor 50 W) bei der Drehzahl abgeben, mit der sie bei einer Geschwindigkeit des VTF von 1 m/s laufen. Mit den gewählten Antriebsrädern mit einem Durchmesser von 75 mm ergibt sich die Motordrehzahl in Abhängigkeit der Getriebeuntersetzung entsprechend

$$n_{Motor,v-1} = \frac{1 \text{ m/s} \cdot g_{Getr}}{\pi \cdot d_{Rad}} \approx 4,24 \frac{1}{s} \cdot g_{Getr} . \quad (7-3)$$

Der Motors PD 4266 [108] der Firma Transmotec leistet bei Verwendung der verfügbaren Getriebeuntersetzung von 17:1 maximal 67 W bei einer Motordrehzahl von 4325 1/min. Bei dieser Drehzahl bewegt sich das Fahrzeug mit 1 m/s vorwärts. Zusätzlich liegt die maximal abrufbare Leistung über den veranschlagten 50 W für das Befahren einer Rampe mit 20 % Steigung.

Die Antriebe für die DTP und die Gurtbänder sind weniger leistungskritisch als die Traktionsantriebe. Für die DTP wird angenommen, dass sie sich mit einer Winkelgeschwindigkeit von mindestens  $2\pi/s$  drehen kann. Diese Forderung ist weniger wichtig für den Individualbetrieb. Vielmehr soll im Schwarmbetrieb das einzelne Fahrzeug schnell unter der gemeinsamen Last seine Orientierung ändern können. Hierzu ist eine zügige Verdrehung der DTP notwendig, wenngleich das notwendige Drehmoment dabei gering ist. Da konstruktiv bedingt eine Untersetzung von Getriebeausgangswelle zu DTP-Achse von ca. 7:1 vorliegt (vgl. 5.1.1), muss die Getriebe-Motor-Einheit eine Drehzahl von 7 1/s bzw. 420 1/min erlauben. Somit wird der Motor PD3246 mit einer Untersetzung von 14:1 verwendet. Diese Kombination besitzt eine Leerlaufdrehzahl von 521 1/min.

Für die optionalen Gurtbänder wird angestrebt, in 3 s einen WST komplett aufnehmen bzw. abgeben zu können. Bei einem Gleitreibungskoeffizient von  $\mu_r = 0,25$  für Trockenreibung zwischen Kunststoffriemen und Stahl [19], ergibt sich bei der angenommenen Nutzlast von 10 kg eine notwendige Leistung von ca. 2,5 W (bei einer Gurtgeschwindigkeit von 0,1 m/s). Mit den Motoren PD3246 und einer Getriebeuntersetzung von 100:1 und den verwendeten Gurtscheiben mit einem Durchmesser von 40 mm lassen sich diese Leistungsdaten erreichen (Drehzahl 73 1/min, Fördergeschwindigkeit ca. 0,15 m/s, max. Motorleistung 7 W).

### **Rechner**

Um eine möglichst lange Einsatzzeit realisieren zu können, ist es sinnvoll, stromsparende Onboard-Rechner beim VTF zu verwenden. Ein hinsichtlich minimaler Energieaufnahme optimierter, IBM-kompatibler PC ist der Wafer-LX2-R11 [46]. Ausgestattet mit einem AMD Geode LX 800 Prozessor, benötigt der gesamte PC ca. 6 W Leistung im Betrieb. Diese geringe Leistungsaufnahme ist unter anderem der Tatsache zu verdanken, dass der Prozessor lediglich mit 500 MHz getaktet ist.



## Sensoren

Als Abstandssensor zur Kollisionsvermeidung kommt der analoge IR-Abstandssensor GP2Y0A21YK0F der Firma Sharp [91] zum Einsatz. Dieser besitzt einen optischen Positionssensor (position sensitive detector, PSD), eine Infrarot emittierende Diode (infrared emitting diode, IRED) sowie eine Signalverarbeitungseinheit. Abhängig vom Abstand eines Objekts zum Sensor wird das von der IRED emittierte Signal im PSD aufgenommen. Der Sensor besitzt einen Messbereich von 10 cm bis 80 cm. Jeweils zwei der Sensoren sind an Vorder- und Rückseite eines VTF installiert.

Zur Detektion von optischen Markierungen auf dem Hallenboden werden ebenfalls IR-Sensoren verwendet. Aufgrund der kompakten Bauweise und der guten Detektion von Farbkanten, sowohl bei Sonneneinstrahlung als auch bei Dunkelheit, kommt hierfür der Sensor CNY70 [119] der Firma Vishay Semiconductors zum Einsatz. Er arbeitet wie der Abstandssensor GP2Y0A21YK0F mit einer IRED. Die vom Objekt reflektierte IR-Strahlung wird beim CNY70 mit einem Phototransistor aufgenommen und erzeugt ein analoges Ausgangssignal in Abhängigkeit von der Intensität der IR-Strahlung.

## Konstruktion

Da die Führungsschiene der Rampe mittig unter dem Fahrzeug verläuft (vgl. Bild 14), muss Mittelbereich des VTF ausreichend Platz bieten. Das hat zur Folge, dass die Antriebsmotoren oberhalb der Welle der Antriebsräder positioniert sind. Die fertige

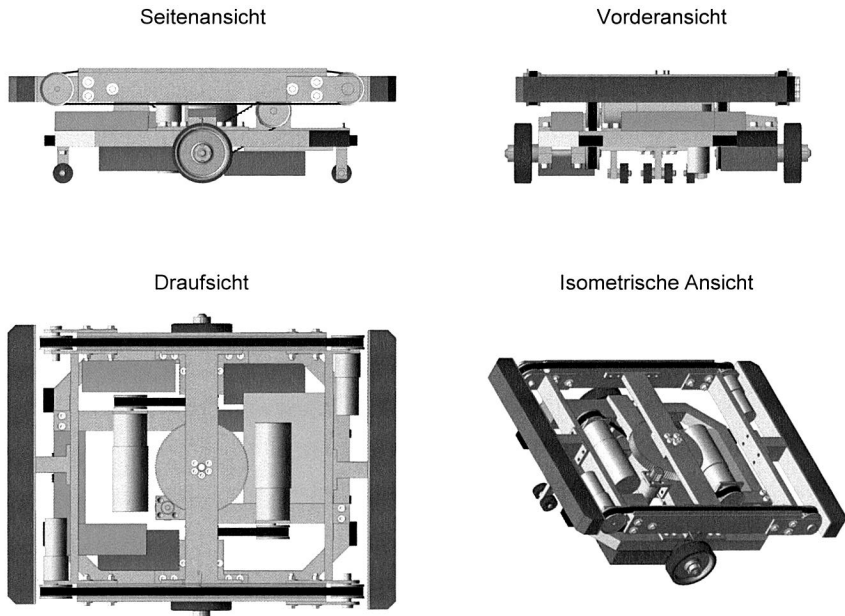


Bild 50: Unbeladenes VTF mit Motorregler (rot), Akkumulatoren (orange), Fahrzeug-PC (gelb) und WLAN-Anschluss (grün)

Konstruktion des VTF ist in Bild 50 dargestellt. Symbolisch sind Volumina für die Motorregler (rot), die LiPo-Akkumulatoren (orange), den Fahrzeug-PC (gelb) und den WLAN-Anschluss (grün) reserviert.

Die abgeschätzten Kosten und das abgeschätzte Gewicht für ein VTF können Tabelle 16 entnommen werden. Dabei ist zu beachten, dass die kalkulierten ca. 1700 € für ein VTF bei massenhafter Beschaffung der Komponenten signifikant sinken werden.

Tabelle 16: Kosten und Gewicht der ausgewählten Komponenten eines VTF (Stand 2013). Für die genannten Preise wird die Anzahl „Stückzahl“ zugrunde gelegt, sodass die Serienpreise um einiges niedriger liegen.

Komponente	Stückpreis in € (ca.)	Stückzahl	Gesamt- gewicht in kg	Gesamt- kosten in €
Antriebsmotor PD4266 [107]	178	2	1,182	290
LIPO-Akku 35C 11,1V [78]	63	3	1,317	189
DTP Motor PD3246 139:1 [107]	109	1	0,256	109
WST-Motor PD3246 100:1 [107]	91	2	0,256	182
Motorregler [96]	50	3	0,05	150
Rechner (incl. 1 GB Ram; 4 GB CF) [46]	370	1	0,3	370
Kugellager, Antriebsräder, Stützräder, Zahnscheiben, Zahnriemen, Wellenmutter, Alu-Rahmen, 5 V DCDC Wandler, Sensoren, Sonstiges	450	1	15	450
Summe			≈18,5	1740

### 7.2 Prototypische Realisierung des VTS

In Bild 51 ist eine typische Szene des prototypisch umgesetzten VTS gezeigt. Darin sind die zwei realisierten Exemplare der in 5.1.2 beschriebenen Fahrzeuge zu sehen. Der verwendete PC zur Bedienung des VTF und zur Bereitstellung des Weltmodells besitzt eine mit 3 GHz getaktete Intel Pentium 4 CPU und verfügt über 1 GB RAM. Als Betriebssystem kommt Windows XP zum Einsatz. Es wurde eine Deckenkamera installiert. Die USB Webcam liefert eine maximale Videoauflösung von 800 x 600 Bildpunkten, wird aber aus Gründen der verfügbaren Rechenleistung mit geringerer Auflösung von 320 x 240 Bildpunkten betrieben.

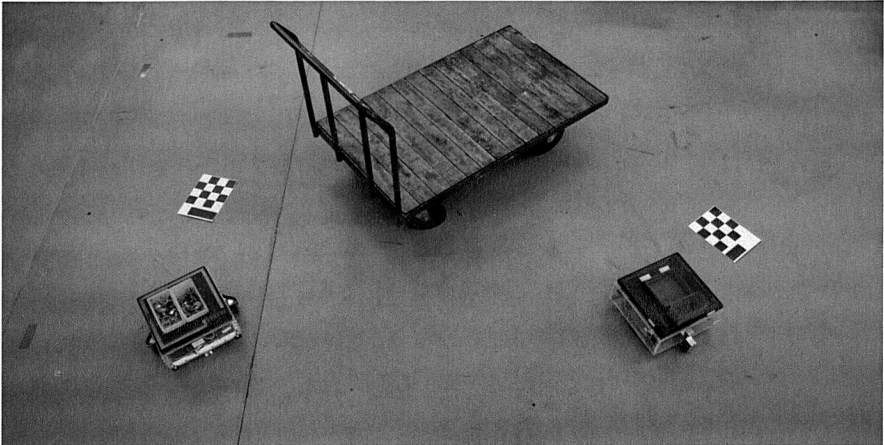


Bild 51: Prototypische Umsetzung des VTS im Versuchslabor. Es sind zwei Fahrzeuge, zwei Zielmarker (ID 7 und 12) und ein Hindernis zu sehen.

Die in Bild 51 dargestellte Szene steht in der GUI (vgl. 6.5) als Kamerabild zur Verfügung. In Bild 52 ist links eine Detailaufnahme der verwendeten Deckenkamera (Hercules Deluxe Optical Glass [38]) dargestellt. Sie ist in einer Höhe von ca. 4 m über dem Hallenboden montiert. Rechts ist das resultierende Kamerabild zu sehen.

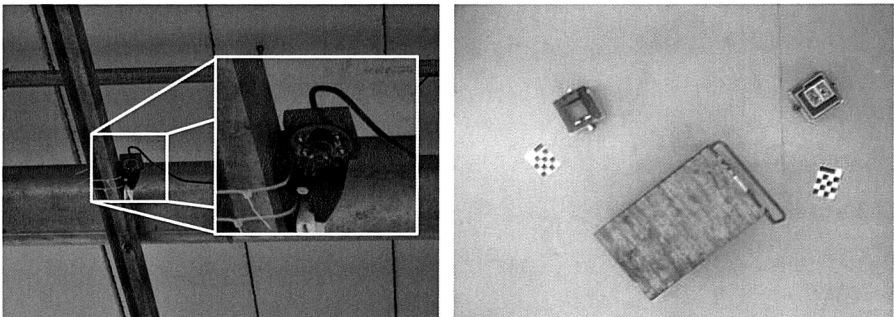


Bild 52: Abbildung der Deckenkamera und Deckenkamerabild der in Bild 51 gezeigten Szene.

Im Gegensatz zu den in Tabelle 16 aufgeführten geplanten Kosten für ein VTF von 1740 € lassen sich die in 5.1.2 beschriebenen, prototypischen Fahrzeuge aufgrund kostengünstigerer Komponenten und dem Verzicht auf eine DTP mit ca. 940 € pro Fahrzeug realisieren. Hinzu kommen Kosten für die Deckenkamera (ca. 18 €), ein Ladegerät für die verwendeten Bleiakkumulatoren der Fahrzeuge (ca. 60 €) und für den stationären PC (ca. 600 €).

Die Umsetzung der in dieser Arbeit beschriebenen Softwarekomponenten erfolgte ausschließlich (bis auf das Betriebssystem Windows) mit open Source Softwaretools.

Während auf diese in 7.3 eingegangen wird, beleuchtet 7.4 die Erprobung des prototypischen Gesamtsystems im Laborbetrieb.

### 7.3 Im Rahmen der Entwicklung eingesetzte Software

In diesem Kapitel erfolgt die Beschreibung von Betriebssystemen, Programmiersprachen, Bildverarbeitungstools, GUI und Middleware (insb. ROS), die bei der prototypischen Umsetzung des beschriebenen VTS-Konzepts zur Verwendung kommen.

#### 7.3.1 Betriebssysteme

Auf den in den Fahrzeugen verbauten PC ist die auf Debian [26] basierende Linux Distribution Ubuntu [109] installiert. Mit Hilfe des Advanced Package Tools (APT), lassen sich alle benötigten Programme (s. u.) in einfachster Weise installieren. Im Rahmen des Projekts sind die Versionen 8.0.4 (Hardy Heron) und 10.04.4 (Lucid Lynx) zum Einsatz gekommen.

Da die Fahrzeugrechner häufig neu aufgesetzt werden müssen, hat sich die Systeminstallation mittels Preboot Execution Environment (PXE) [47] als effizient erwiesen. Der hierfür verwendete Server läuft ebenfalls mit dem Betriebssystem Ubuntu 8.0.4.

Als Betriebssystem des stationären Rechners kommt Windows XP [71] zum Einsatz.

Um die Middleware ROS [122] (siehe 7.3.4) zu nutzen, wurde auch ein zentraler Rechner mit Ubuntu aufgesetzt. Aufgrund von Performanceproblemen ist dieses Projekt jedoch wieder verworfen worden.

#### 7.3.2 Python, Numpy, Cython, PyQt4

Zu Beginn der Entwicklung des VTS musste eine Entscheidung bezüglich der zu verwendenden Programmiersprache getroffen werden. In die engere Auswahl kamen C++ [97] und Python [79]. Dabei liegt ein grundlegender Unterschied darin, dass C++ eine kompilierte Sprache und Python eine interpretierte Sprache darstellt. Daher läuft Python tendenziell langsamer als C++ hat aber den Vorteil, sehr schnelle Softwareentwicklung zu ermöglichen (dynamische Typisierung, keine Pointer, plattformunabhängig (es wird lediglich ein Python Interpreter benötigt), hohe Lesbarkeit (Blöcke durch Einrückung), ...). Zusätzlich arbeitet Python zumeist in Kombination mit low-level Sprachen wie C und C++. Somit ergeben sich, durch Einbindung hocheffizienter Module, leistungsstarke Programme. Darüber hinaus unterstützt Python unter anderem funktionale und objektorientierte Programmierung. Somit fiel die Entscheidung, die Software für das VTS in Python 2 zu erstellen.

Die Gesamtprogramme für den stationären Rechner und für die VTF sind jeweils in einige quasi parallel laufenden Prozessen (Threads) unterteilt. Tabelle 17 und Tabelle 18 zeigen diese mit einer kurzen Erläuterung. Diese Threads sind in Python geschrieben, greifen jedoch häufig auf z. B. in C geschriebene Module zu.

Die Entscheidung, Python als Programmiersprache zu verwenden, bringt es mit sich, dass Source Code für Module wie Pfadplaner oder zur Bildverarbeitung sehr langsam läuft. Es existieren jedoch Möglichkeiten, um mittels kompilierter Sprachen Module zu

erzeugen, die wesentlich effizienter arbeiten als purer Python Code. Im Rahmen des Projekts kommt daher die Sprache Cython [10] zum Einsatz, um Python Code zu kompilieren und damit die benötigte Zeit zur Ausführung eines Algorithmus massiv zu reduzieren. Werden spezielle Softwarekonstrukte wie Schleifen über nicht rechteckige Arrays betrachtet, sind Performanceverbesserungen um den Faktor 1000 gegenüber purem Pythoncode möglich [11]. Der Einsatz von Cython verlangt jedoch statische Typisierung, was einen der Hauptvorteile von Python – dynamische Typisierung – im Hinblick auf gute Lesbarkeit und schnelle Erlernbarkeit zunichtemacht.

Tabelle 17: Threads, die auf dem stationären Rechner laufen

Threadname	Funktion
Camera	Bildaufnahme mit fester Frequenz, Bildverarbeitung (vgl. 6.2)
Chessfinder	Finden der Zielmarker mit Orientierung und ID (vgl. 6.2)
GUI	Grafische Benutzeroberfläche (PyQt4)
WorldModel	Läuft mit fester Frequenz (parallel zu „Camera“). Update der Zustandsschätzungen für alle Weltobjekte in Abhängigkeit der Kameradaten und Fahrzeugdaten (vgl. 6.1, 6.3, 6.4)
WorldModelServer	Kommunikation zwischen WorldModel und VTF

Tabelle 18: Threads, die auf einem VTF laufen

Threadname	Funktion
AutomaticGuidance	Partikelfilter zur Positionsschätzung bei Andockvorgängen (vgl. 5.2.6)
IoHandler	Schnittstelle zu Mikrocontrollern (vgl. 5.2.2)
Pathplanner	Pfadplanung (vgl. 5.2.3)
TrajektorieComparator	Trajektorienvergleich (vgl. 5.2.5)
Tracker	Generierung von Fahrbefehlen um einen geplanten Pfad abzufahren (vgl. 5.2.4)
UserInterface	Möglichkeit zur direkten Fahrzeugkontrolle über Kommandozeile (unabhängig von GUI)
WorldModelClient	Kommunikation zwischen VTF und WorldModel

Zur Lösung numerischer Probleme mit Python existiert das Modul Numpy [75]. Es stellt eine Lösung zur effizienten Handhabung und numerischen Verarbeitung großer, multidimensionaler Arrays dar. Somit lassen sich Algorithmen, die als Operationen auf Arrays ausgedrückt werden können, effizient in Numpy implementieren. Da es jedoch nicht notwendig ist, den Code zu kompilieren, liefert Numpy besser lesbaren und leichter zu verändernden Code als Cython. Hinsichtlich Performance liegt Cython in den meisten Fällen vor Numpy. In [11] wird in einem Vergleich mit großen, nichtquadratischen Arrays ermittelt, dass Numpy bis zu 27-mal länger benötigt als Cython. Im konkreten Fall des Partikelfilters aus 5.2.6 ließ sich die benötigte Berechnungszeit von 0,851 s (mit Numpy) auf 0,141 s (mittels Cython erzeugtes Modul) reduzieren. Neben dem signifikanten Performancegewinn gegenüber Python bei gleichbleibend guter

Lesbarkeit und Erlernbarkeit bringt die Verwendung von Numpy einen weiteren Vorteil mit sich: Es besitzt hinsichtlich der Bedienung viele Parallelen zu Matlab [73]. Somit lassen sich Algorithmen, die in Matlab entwickelt und geprüft worden sind, in einfacher Weise in Numpy umsetzen. Ebenso ist es möglich, auf den Einsatz von Matlab zu verzichten und Numpy direkt, z. B. in Verbindung mit Matplotlib [105], zur wissenschaftlichen Programmierung einzusetzen [60].

Um die in 6.5 vorgestellte GUI zu realisieren, kommt eine der bekanntesten Klassenbibliotheken zur plattformunabhängigen Programmierung grafischer Benutzeroberflächen [105] zum Einsatz. Ursprünglich von der Firma Trolltech entwickelt, wurde Qt [80] im Jahr 2008 von Nokia gekauft, weiterentwickelt, und 2011 unter dem Namen Qt Project als freie Software veröffentlicht. Die im Rahmen dieser Arbeit verwendeten Bindings von Qt für Python werden mit PyQt4 bezeichnet.

Alle externen Module, die im Rahmen des Projekts zum Einsatz kommen, sind in Tabelle 19 angegeben. Zum Teil werden diese in den folgenden Abschnitten detailliert erläutert.

Tabelle 19: Externe Module, die im Rahmen des Projekts verwendet werden

Modul	Einsatzzweck
Numpy [75]	Python Erweiterung für numerische Berechnungen (Weltobjekte, Kalmanfilter ...)
OpenCV [124]	Bildverarbeitungsbibliothek (Vordergrundobjekterkennung, Zielmarkerdetektion, ...)
PyQt4	GUI Bibliothek zum Erstellen der Benutzeroberfläche des stationären Rechners
PySerial [64]	Ansprechen des RS232-Interfaces auf den VTF zur Kommunikation mit zentralem Mikrocontroller
ROS [122]	Framework zur Softwareentwicklung für Roboteranwendungen

### 7.3.3 OpenCV

Die Open-Source-Computer-Vision-Bibliothek (OpenCV) [76] ist eine ursprünglich von Intel entwickelte und aktuell von Willow Garage gewartete Bildverarbeitungsbibliothek. Sie besitzt mehr als 2500 optimierte Algorithmen aus den Bereichen Computervision und maschinellem Lernen. Während auch andere Softwarepakete existieren, um die Bildverarbeitungsprozesse aus 6.2 ablaufen zu lassen (z. B. Roborealm [84], Halcon [74]), hat OpenCV den gravierenden Vorteil, unter der BSD (Berkeley Software Distribution) Lizenz veröffentlichen zu sein. Damit ist diese Programmbibliothek sowohl für Forschungs- als auch für kommerzielle Zwecke kostenfrei verfügbar. Die resultierende, große Anwendergemeinde und die gute Dokumentation [18], auch für die verwendeten Python-Bindings [124], macht die Arbeit mit OpenCV im Zusammenhang mit Python effizient. Nicht umsonst verwenden neben etablierten Unternehmen wie Google, Yahoo, Intel, Sony, Honda und Toyota insbesondere Forschungsinstitute OpenCV. So war OpenCV z. B. ein wichtiger Baustein des Visionssystems von „Stanley“, dem autonom fahrenden VW Touareg R5, der die DARPA (Defense Advanced Research Projects

Agency) Grand Challenge 2005 gewann [103]. Ebenso kommt OpenCV auf vielen Servicerobotern, wie z. B. Care-O-Bot oder PR2 (vgl. 2.5), zur Anwendung. Die Bibliothek unterstützt die Betriebssysteme Windows, Linux, Android und Mac-OS und bietet Programmschnittstellen für C++, C, Python und Java. Geschrieben ist OpenCV in C++.

Zum Einsatz kommt in dieser Arbeit OpenCV-2.4.2. Nach dem kostenlosen Download der Installationsdateien und der einfachen Installation auf dem stationären Windows-Rechner ist es sofort einsatzfähig. Die vorhandenen Beispielanwendungen erleichtern den Einstieg zusätzlich.

#### 7.3.4 Middleware

Um Entwicklungszeit zu reduzieren und die Wiederverwendbarkeit von Software zu erhöhen, bietet sich der Einsatz von Softwareframeworks an, die grundlegende Funktionen für spezifische Aufgabenbereiche bereitstellen. ROS [122] stellt ein solches Framework für den Einsatz bei Roboteranwendungen dar. Es wird, wie OpenCV, von Willow Garage gewartet und ist unter der BSD-Lizenz veröffentlicht. Dabei lassen sich die Design Ziele von ROS nach [81] folgendermaßen zusammenfassen

- Peer-to-peer: Werden mehrere Rechner über Ethernet oder WLAN miteinander verbunden, ist es kritisch, den Datenverkehr über einen zentralen Server ablaufen zu lassen. Durch die Verwendung von peer-to-peer-Kommunikation reduziert sich sowohl der Netzwerktraffic als auch die Auslastung des zentralen Servers, was auch die Skalierbarkeit des Gesamtsystems sicherstellt.
- Tools-based: Die Abkehr von großen monolithischen Systemen hin zu kleineren, modular aufgebauten Softwarepaketen (Tools) erhöht deren Wartbarkeit und damit Robustheit.
- Multi-lingual: Um jeden Entwickler der großen ROS-Nutzer- und Entwicklergemeinschaft in seiner gewohnten Programmierumgebung arbeiten zu lassen, basiert ROS auf einer programmiersprachenneutralen Architektur.
- Thin: Maximale Wiederverwendbarkeit von Treibersoftware und Algorithmen lässt sich erreichen, wenn diese in eigenen Softwarebibliotheken abgelegt werden und keine Abhängigkeit von der verwendeten Middleware besteht. Kleine (thin) Programme, sogenannte Wrapper, sorgen dann dafür, dass die in den Bibliotheken abgelegte, komplexe Logik für ROS aber auch beliebige andere Middleware verfügbar ist.
- Free and Open-Source: Die Veröffentlichung von ROS unter der BSD Lizenz ermöglicht kommerzielle und nichtkommerzielle Entwicklungsprojekte. ROS erlaubt auch die Einbindung von Modulen, die unter anderer Lizenz veröffentlicht sind, jedoch endet diese jeweils bei den Modulgrenzen. Zusätzlich macht es der open-source-Ansatz einfach, Module zu verstehen, Fehler zu finden und Verbesserungen einzubringen.

Basierend auf diesen Designzielen bietet ROS eine umfassende Infrastruktur in Form von Middleware- und Framework-Funktionalitäten, die eine PC-basierte Entwicklung von Roboteranwendungen ermöglicht. Dazu zählen unter anderem eine interprozessua-

le Kommunikationsstruktur, low-level-Ansteuerung von Betriebsmitteln, Hardware-abstraktion, Visualisierung und Debugging-Tools oder grundlegende Bibliotheken für Koordinatentransformation und Kinematikberechnung [31].

Zur Anwendung kommt ROS in einer Vielzahl verschiedener Roboterapplikationen. Beispielhaft seien hier genannt: Serviceroboter wie Care-O-Bot 3 (Fraunhofer IPA), PR2 (WillowGarage) oder TUM-Rosie (TU-München); autonom agierende Fahrzeuge wie Marvin (University of Texas) oder Grizzly (Clearpath Robotics), Lehrroboter wie der Robotino (Festo Didactic) und Flugroboter wie der CityFlyer (CCNY Robotics Lab) [123].

Im vorliegenden Fall des VTS erfolgte die Installation von ROS auf einem stationären Rechner mit Linux-Betriebssystem (Ubuntu). Die Implementierung der einzelnen Prozesse (Bildverarbeitung, Zustandsschätzung für Weltmodelle) ließ sich schnell durchführen. Jedoch konnten die Kameradaten im Vergleich zur Python-Implementierung ohne ROS, nur mit relativ niedriger Frequenz ausgelesen werden. Damit resultierte eine Aktualisierungsrate des Weltmodells von ca. 1/s. Aus Gründen der niedrigen Performance (die nicht unbedingt an ROS liegen müssen, sich jedoch im Rahmen des Projekts nicht beheben ließen) wurde daher die Verwendung von ROS im Zusammenhang mit dem VTS verworfen.

Festzuhalten ist, dass viele, in der reinen Python-Implementierung, fehleranfällige Implementierungsschritte wie die Inter-Prozess-Kommunikation (Queue basierte Kommunikation zwischen Threads) oder die Bereitstellung von Daten für die WLAN-Kommunikation (mittels Stream Sockets) schneller und effektiver mit ROS erledigt werden können. Auch die riesige Fülle an bereitstehenden Softwarekomponenten zur Visualisierung, Pfadplanung, Zustandsschätzung usw. lassen die Verwendung von ROS vielversprechend erscheinen.

### 7.4 Erprobung des Transportsystems im Laborbetrieb

Teilweise werden bereits in den Kapiteln zu den einzelnen Komponenten Ergebnisse aus der prototypischen Systemerprobung gezeigt. So sind in 5.2.3 Details zur Planung von Pfaden und in 5.2.4 Details zum Tracker erläutert.

Dieses Kapitel beschreibt die Genauigkeit der Positionsbestimmung der VTF im prototypischen Gesamtsystem. Als Messgerät zur absoluten Positionserfassung des VTF dient ein Lasertracker der Firma API [6]. Dieser ist in der Lage, die 3D-Position eines Retroreflektors (Target) mit einer Genauigkeit von bis zu  $\pm 10 \mu\text{m}$  zu messen. Um die Messungen durchzuführen, steht ein Standardtarget zur Verfügung, welches sich nicht, wie das Active Target, auf den Laserstrahl des Trackers ausrichten kann. Daher ist er notwendig, vor jeder Testfahrt das Target manuell auf den Tracker auszurichten und dann weitestgehend geradlinig zu fahren, um den erlaubten Einfallswinkel des Laserstrahls nicht zu überschreiten. Als Experiment wird daher sowohl in x- als auch in y-Richtung jeweils fünfmal geradlinig und parallel versetzt durch den Sichtbereich der Deckenkamera gefahren. Die Ausrichtung des VTF und das Starten der Testfahrt erfolgt hierzu manuell. Bild 53 zeigt sowohl den Lasertracker als auch den Prototyp des VTF mit montiertem und ausgerichtetem Target.



Es war vorgesehen, den Lasertracker mit einem externen Triggersignal zur Durchführung einer Messung zu veranlassen. Dieses Triggersignal erzeugt der stationäre Rechner (vgl. 6) immer gleichzeitig mit der Veranlassung der Aufnahme eines Bilds der Deckenkamera. Es hat sich jedoch gezeigt, dass die Messungen während einer Fahrt des VTF bei externer Triggerung des Messvorgangs zu fehlerhafter Datenverarbeitung im Lasertracker führen. Nach Rücksprache mit API muss daher die Triggerung eines Messvorgangs intern mit einer festen Frequenz von 5 Hz erfolgen. Ebenso loggt der stationäre Rechner die Weltmodelldaten – die geschätzte Position des VTS – mit 5 Hz. Hierzu ist es notwendig, beide Messreihen einzeln manuell zu starten. Anschließend lassen sich beide Positionsmessreihen vergleichen und Genauigkeitsaussagen treffen.

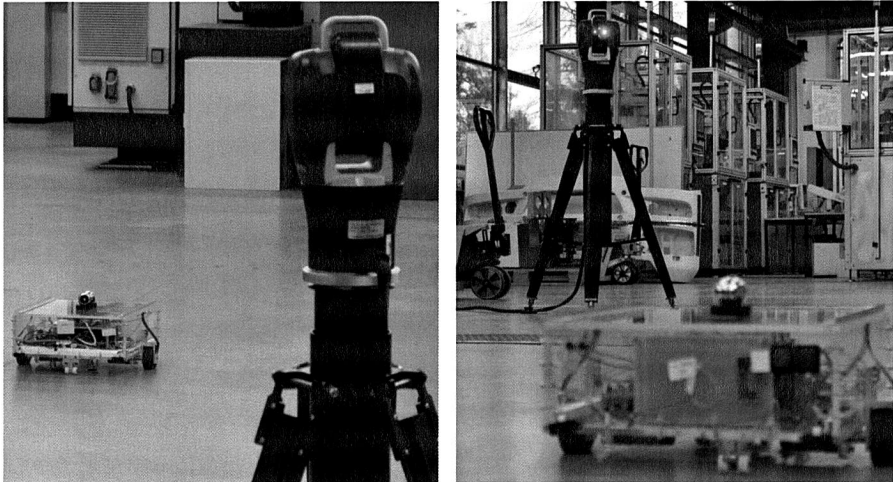


Bild 53: Vermessung der Navigationsgenauigkeit des VTF im Labor des Lehrstuhls FAPS. Links: Retroreflektor (Target) des Lasertrackers, montiert auf einem VTF. Im Vordergrund ist der Lasertracker zu sehen. Rechts: Lasertracker im Betrieb beim Verfolgen des VTF.

Aus genannten Gründen existiert jedoch keine Punkt-zu-Punkt Beziehung zwischen den korrespondierenden Messreihen von Lasertracker und Weltmodell. Um die erfassten Trajektorien miteinander zu vergleichen, wird daher folgendermaßen vorgegangen:

- Ermittlung eines Polynoms sechster Ordnung, welches die mittels Lasertracker erfasste Trajektorie beschreibt (Curve-Fitting). Da das Fahrzeug bei jeder Messung ausschließlich geradeaus fährt, sind die Näherungsfehler sehr gering (vgl. Tabelle 20: Fehler Lasertrackermessung, Polynom).
- Bestimmung der Abstände zwischen dem Polynom sechster Ordnung und der durch das Weltmodell geschätzten Trajektorienpunkte in x- bzw. y-Richtung.
- Berechnung von Varianz und Mittelwert des Fehlers.

Systembedingt ist zu erwarten, dass die Positionsschätzung des VTF vom Betrachtungswinkel der Kamera abhängt. Je nach Höhe des Fahrzeugs inklusive Transportgut verschiebt sich die auf den Hallenboden projizierte Fläche des VTF und damit der

Schwerpunkt des umschließenden Rechtecks umso mehr Richtung Bildrand, je näher sich das Fahrzeug am Bildrand befindet.

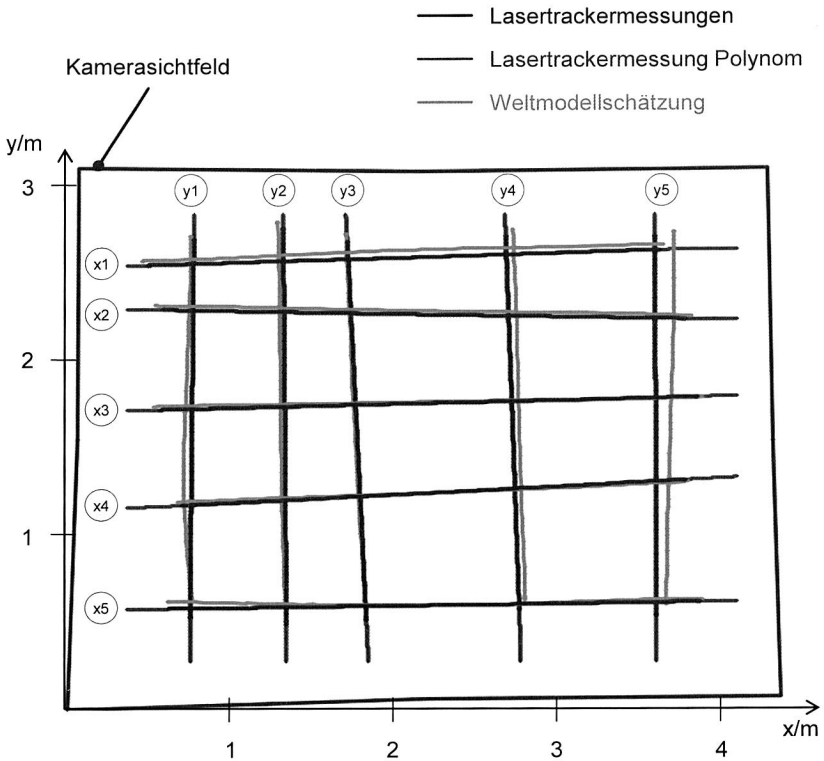


Bild 54: Mittels Lasertracker erfasste und durch das Weltmodell geschätzte Trajektorien von jeweils fünf geradlinigen Fahrten durch den Sichtbereich der Deckenkamera. Die blau dargestellten Trajektorien ergeben sich durch Curve-Fitting aus den mittels Lasertracker erfassten Trajektorien.

Ein zweiter zu erwartender Fehler resultiert aus den optischen Defiziten der eingesetzten low-cost-USB-Kamera. Insbesondere die radiale Verzeichnung tritt bei solchen Kameras häufig in Erscheinung. Dass die optische Achse der Deckenkamera wahrscheinlich nicht exakt lotrecht zum Hallenboden ausgerichtet ist, dürfte zu weiteren Fehlern führen.

Das Ergebnis der Tests zeigt Bild 54. Zunächst fällt auf, dass die Form des mittels Lasertracker erfassten Kamerasichtfelds von einer exakten Rechteckform abweicht. Neben der trapezartigen Verformung, die von einem Ausrichtungsfehler resultiert, lässt sich eine überlagerte Kissenform des Kamerasichtfelds erkennen. Dies erzeugt eine tonnenförmige Verzeichnung der Deckenkamerabilder und damit der VTF-Trajektorien, wie an deren leichter Krümmung hin zum äußeren Rand des Kamerasichtfelds zu erkennen ist.

Die in Tabelle 20 dargestellten Zahlenwerte bestätigen diesen Zusammenhang. Während die im Bereich der Bildmitte verlaufenden Trajektorien  $x_3$  und  $y_3$  geringe mittlere Abweichungen von unter 1 cm aufweisen, zeigt Trajektorie  $y_5$  eine mittlere Abweichung von knapp 9 cm relativ zur mittels Lasertracker erfassten Trajektorie. Auch die Varianz der Abweichung liegt mit  $356,171 \text{ mm}^2$  über allen anderen Fehlerwerten.

Mit der verwendeten Kamera und deren unpräziser Ausrichtung lässt sich demnach keine große Genauigkeit der Positionsschätzung von Weltobjekten erwarten. Obwohl dieser Punkt sicherlich Raum für Verbesserungen bietet, ist eine absolutgenaue Weltobjektpositionsbestimmung von untergeordneter Bedeutung. Da nämlich ein Hindernis oder Zielmarker ebenso wie ein VTF eine fehlerbehaftete Position im Weltmodell besitzt, bleibt der relative Fehler bei Hindernisumfahrungen bzw. Andockvorgängen gering.

Tabelle 20: Differenz zwischen Weltmodellschätzung und mittels Lasertracker erfasster Trajektorien. Zur analytischen Beschreibung der mittels Lasertracker erfassten Trajektorien wird ein Polynom sechster Ordnung verwendet. Die Fehler aller  $x$ -Trajektorien sind in  $y$ -Richtung, die Fehler aller  $y$ -Trajektorien in  $x$ -Richtung angegeben.

	Fehler Lasertrackermessung, Polynom		Fehler Weltmodellschätzung, Polynom	
	Mittelwert in mm	Varianz in $\text{mm}^2$	Mittelwert in mm	Varianz in $\text{mm}^2$
$x_1$	-0,095	0,017	-33,603	40,437
$x_2$	0,052	0,015	-22,494	2,525
$x_3$	-0,076	0,027	-9,239	86,473
$x_4$	-0,166	0,020	-3,945	164,751
$x_5$	-0,050	0,013	-17,179	210,419
$y_1$	-0,038	0,009	-21,955	225,268
$y_2$	0,028	0,009	-22,207	60,242
$y_3$	0,141	0,020	-7,231	12,592
$y_4$	0,099	0,018	42,003	19,079
$y_5$	0,004	0,007	89,206	356,171

Die in Tabelle 6 formulierten Anforderungen an das zu entwickelnde VTS lassen sich zum Teil bereits alleine durch konstruktive Maßnahmen erfüllen. So sind die Transportkapazität und die zur Verfügung stehende Transportfläche sowie Vorrichtungen zur automatischen Lastaufnahme, die Einsatzzeit und die passive Sicherheit durch das in 5.1 beschriebene Vorgehen anforderungsgemäß realisiert. Des Weiteren ist gefordert:

- Fahrgeschwindigkeit von 1 m/s ermöglichen: Die installierte Antriebsleistung und die vorgesehenen Getriebe erfüllen diese Anforderung problemlos. Limitierend wirkt sich hier die zur Verfügung stehende Rechenleistung und die damit verbundene Zykluszeit des Weltmodells von 0,2 s aus. Somit bewegt sich ein Fahrzeug bis zu 20 cm weit, bevor neue Positionsdaten vom Weltmodell verfügbar sind.

- Flexibilität maximieren: Die realisierte automatische Adaption an sich ändernde Umgebungsbedingungen hinsichtlich Hindernispositionen und Hindernisformen sowie sich ändernde Zielpositionen stellt maximale Flexibilität sicher. Ebenso kann die zur Verfügung stehende Transportleistung durch einfachstes Einbringen neuer Fahrzeuge reguliert werden.
- Schwarmbetrieb ermöglichen: Konzeptionell ist das System darauf ausgelegt, mehrere VTF im Schwarm zusammen arbeiten zu lassen. In dieser Arbeit wurde das Thema jedoch nicht adressiert.
- VTS-Kosten extrem niedrig halten: Mit einer Gesamtinvestition von ca. 1600 € für das prototypische VTS und erwarteten Fahrzeugkosten von weniger als 1000 € stellt das System gegenüber den am Markt verfügbaren Lösungen eine extrem günstige Alternative dar.

Zusammengefasst erfüllt das System die Anforderungen, die in Tabelle 6 formuliert sind. Jedoch bringt der Einsatz komplizierter Technik auch Nachteile mit sich. So ist die Absolutgenauigkeit des Systems mit einer mittleren Abweichung von bis zu ca. 9 cm pro Fahrt (vgl. Tabelle 20) verbesserungswürdig. Ebenso stellt die Robustheit der optischen Objekterkennung einen zu optimierenden Punkt dar. Bei weitestgehend konstanten und störungsfreien Lichtverhältnissen können bereits Fehldetektionen von Objekten auftreten (vgl. Tabelle 15). Die Handhabung von sich ändernden Beleuchtungsbedingungen ist bisher lediglich rudimentär (vgl. Bild 40), die Handhabung von Schatten gar nicht realisiert worden. Bedarf für weitere Forschungsarbeiten besteht daher hinsichtlich

- Reduktion der Zykluszeit des Weltmodells. Solange die Deckenkamera mit maximal 5 Hz Bilder aufnimmt, ist eine schnelle Reaktion auf sich bewegende Hindernisse und eine gute Zustandsschätzung der zugehörigen Hindernisobjekte nicht möglich. Neben schnellerer Hardware ist der Einsatz schnellerer Programmiersprachen in Betracht zu ziehen. Die Verwendung der Scriptsprache Python hat auch Studierenden mit geringer Programmiererfahrung die Mitarbeit am VTS ermöglicht. Für performancekritische Teile sollte jedoch verstärkt Gebrauch von kompilierten Sprachen wie C oder C++ gemacht werden
- Verbesserung der optischen Detektion von Weltobjekten. Das umfasst sowohl die adaptive Hintergrundschätzung, um auf Lichtschwankungen reagieren zu können, als auch die sichere Konturerkennung von Objekten, wenn sie sich in geringem Abstand zueinander befinden. Dies lässt sich z. B. durch Verwendung rechenaufwändiger Algorithmen wie die Codebook-Methode oder durch Betrachtung des YUV Farbraums realisieren.
- Verbesserung der mathematischen Modellierung von Hindernis- und Fahrzeugobjekten, um Schwächen der optische Detektion von Weltobjekten zu kompensieren. Eine Möglichkeit besteht in der Verwendung von Zustandsschätzverfahren mit multimodalen Wahrscheinlichkeitsdichteverteilungen, um mehrere Hy-

pothesen über den Zustand eines Weltmodells zu verfolgen, solange die optische Bilderkennung fehlerhafte Daten liefert. Mit dem momentan verwendeten Kalmanfilter ist dies nicht möglich.

- Verbesserung der Absolutgenauigkeit der Positionsbestimmung von VTF. Dies kann zum Teil durch Kompensation der radialen Verzeichnung und des Ausrichtungsfehlers der Deckenkameras erreicht werden. Systembedingt bleibt jedoch auch dann ein Fehler der Positionsbestimmung bestehen. Da nämlich das auf den Boden projizierte Bild eines Objekts als dessen Grundfläche betrachtet wird, ändert sich in Abhängigkeit von der Höhe des VTF und in Abhängigkeit von der Position des VTF relativ zur optischen Achse der Deckenkamera, die erfasste Grundfläche und damit die erfasste Position eines VTF.
- Erweiterung des Einsatzgebiets des VTS. Um eine größere Fläche der Betriebsumgebung überwachen zu können, ist es notwendig, mehrere Deckenkameras zu verwenden. Außerdem bietet der Schwarmbetrieb viele Möglichkeiten, das VTS vielseitiger einzusetzen. Die maximale Transportkapazität lässt sich auf diese Weise beliebig erweitern.



## 8 Zusammenfassung und Ausblick

In der vorliegenden Arbeit wird die Entwicklung eines Vielseitigen Transportsystems (VTS) beschrieben. Ausgehend von einer Studie aktueller Transportaufgaben in der fertigen Industrie und in einem Krankenhaus werden Anforderungen an ein VTS abgeleitet. Es folgt die Ausarbeitung eines Konzepts für ein hinsichtlich der ermittelten Anforderungen optimiertes VTS. Die erfolgreiche Umsetzung des entwickelten Konzepts ermöglicht die Herstellung besonders kostengünstiger, autonom agierender Fahrzeuge für den Transport von Nutzlasten mit einer Masse von ca. 10 kg. Dabei wird als Kernkomponente des Systems ein zentrales Weltmodell erstellt und jedem Fahrzeug mittels WLAN zur Verfügung gestellt. Hierzu sind stationäre Kameras über der Betriebsumgebung notwendig, die sowohl alle VTF als auch alle Hindernisse und alle optisch markierten Zielpositionen detektieren. Auf diese Weise lässt sich der Einsatz kostenintensiver Lasersensorik vermeiden.

Die Realisierung des VTS lässt sich in zwei Themengebiete einteilen. Zum einen sind das die Arbeiten an den Vielseitigen Transportfahrzeugen (VTF). Diese umfassen hardwarespezifische Aspekte wie die Auswahl der Fahrwerkkinematik und deren Komponenten, die Konzeptfindung zur Überwindung von Höhendifferenzen und die computergestützte Konstruktion von Fahrzeug und zugehöriger Rampe. Ebenso sind softwarespezifische Aspekte wie Motorregelung, autonome Navigation, Selbstlokalisierung im Weltmodell und präzise Positionsschätzung bei Andockvorgängen zu berücksichtigen.

Zum anderen sind zur Realisierung des VTS Arbeiten an den stationären Komponenten des VTS notwendig. Dies sind neben der Installation der Deckenkameras insbesondere softwaretechnische Belange zur Generierung des Weltmodells auf einem stationären Rechner. Sie umfassen die Modellierung von Weltobjekten inklusive der Schätzung des jeweiligen Zustandsvektors entsprechend dem vorliegenden Weltobjekttyp. Ebenso erfolgt auf dem stationären Rechner die Verarbeitung der von den Deckenkameras aufgenommenen Bilder, um Visionobjekte und Zielmarker zu detektieren. Um schließlich die Zustandsschätzung der Hindernis- und Fahrzeugobjekte durchführen zu können, ist eine geeignete Zuweisung der detektierten Visionobjekte notwendig. Diese Zuweisung findet ebenfalls mit Hilfe des stationären Rechners statt.

Das Konzept des VTS, basierend auf Deckenkameras und Weltmodell, ist prinzipiell auf Fahrzeuge beliebiger Größe und Masse anwendbar. Entsprechend den ermittelten Anforderungen ist das Konzept jedoch insbesondere dann wirtschaftlich, wenn viele kleinere Fahrzeuge auf engem Raum operieren. Dann kann mit einer geringen Zahl an Deckenkameras eine große Zahl an Fahrzeugen mit Weltmodelldaten versorgt und somit deren autonomer Betrieb ermöglicht werden. Daher befasst sich die Arbeit zwar auch mit der Optimierung der VTF hinsichtlich geringer Kosten (geeigneter Fahrwerktyp, geringe Größe, kostengünstige Antriebsmotoren...), jedoch liegt der Schwerpunkt auf der zuverlässigen Weltmodellbereitstellung. Die Entwicklung der hierzu benötigten

Software auf dem stationären Rechner und auf den Fahrzeug-PCs stellt den mit Abstand arbeitsaufwändigsten Teil der vorliegenden Arbeit dar.

Auf dem stationären Rechner laufen Softwarekomponenten zur Hintergrundschätzung, Hintergrundsubtraktion, Konturerkennung und Errechnung umschließender Rechtecke von Visionobjekten sowie die Zielmarkerdetektion. Ebenso muss die Schätzung der Weltobjektzustände und Zuweisung der ermittelten Visionobjekte zu den geschätzten Weltobjektpositionen mittels Ungarischer Methode durchgeführt werden. Schließlich ist es notwendig, die Daten des Systems mittels GUI zu visualisieren, was auch auf dem stationären Rechner passiert. Auf den Fahrzeugrechnern sind Pfadplaner, Tracker, Trajektorienvergleicher und multimodaler Positionsschätzer zu implementieren.

Die Erstellung der jeweiligen Softwarekomponenten wurde erfolgreich mit Hilfe der Programmiersprache Python durchgeführt. Isoliert funktioniert die Bildverarbeitung zügig. Die Zustandsschätzung für Fahrzeug- und Hindernisobjekte mittels Kalmanfilter liefert gute Ergebnisse. Auch die Ungarische Methode wird erfolgreich und leicht modifiziert implementiert. Für die Fahrzeuge stehen ein schneller, RRT-basierter Pfadplaner, ein stabiler Tracker, schnelle Motorregler und ein zuverlässiges Partikelfilter zur Zustandsschätzung bei Andockprozessen zur Verfügung.

Bei der Integration aller Teilkomponenten in das Gesamtsystem machen sich die niedrige, zur Verfügung stehende Rechengeschwindigkeit und die Verwendung der Scriptsprache Python nachteilig bemerkbar. Es resultiert eine verhältnismäßig hohe Zykluszeit für die Aktualisierung des Weltmodells. Mit Hilfe verschiedener Maßnahmen, wie dem temporären Abschalten der GUI oder der manuellen Auslösung der Funktion zur Detektion von Zielmarkern, lässt sich eine Zykluszeit von 0,2 s realisieren. Zusätzlich wird die Totzeit, die aus der rechenaufwändigen Bildverarbeitung resultiert, explizit bei der Weltmodellerstellung berücksichtigt. Die prototypische Realisierung des Gesamtsystems ist damit voll funktionsfähig und erfüllt alle an das System gestellten Anforderungen.

Aus der in 7.4 erläuterten Praxiserprobung des VTS sind einige Optimierungsmaßnahmen abgeleitet, die in weiterführenden Forschungsprojekten adressiert werden sollten. Dazu zählt zunächst die Reduktion der Zykluszeit des Weltmodells. Diese lässt sich voraussichtlich durch die Verwendung schnellerer Hardware und den Einsatz kompilierter Programmiersprachen erreichen. Des Weiteren ist es notwendig, die optische Detektion und die mathematische Modellierung der Weltobjekte zu verbessern. Somit lässt sich die Robustheit des Systems erhöhen und störende Einflüsse von Sonnenlicht, Schattenwurf oder sich ändernden Beleuchtungssituationen kompensieren. Um die Absolutgenauigkeit der Positionsbestimmung von den VTF zu erhöhen ist es darüber hinaus sinnvoll, systembedingte Fehler der deckenkamerabasierten Positionserfassung zu eliminieren. Neben der radialen Verzeichnung und der ungenauen Ausrichtung der Deckenkamera ist insbesondere die fehlerhafte Projektion des Fahrzeugumrisses auf den Hallenboden zu kompensieren. Schließlich ist die Erweiterung des Einsatzgebiets des VTS sinnvoll, welche sich durch die Installation weiterer Deckenkameras realisieren lässt.



Neben den genannten technischen Weiterentwicklungen des VTS ist insbesondere die Untersuchung weiterer neuartiger Anwendungsfälle durchzuführen. Die in 2.1 erläuterten Aspekte zur Fabrikorganisation bieten viel Raum für Optimierungen. Dazu zählt neben der Kombination der Vorteile verrichtungsorientierter und flussorientierter Organisation die Idee, Fertigungsprozesse auf Bodenniveau abzusenken, da der Materialfluss durch die VTF sowieso auf Bodenniveau erfolgt. In diesem Zusammenhang sind auch Konzepte zu bewerten, um Höhendifferenzen zu überwinden. Der in dieser Arbeit vorgestellten Lösung mittels Rampe, sollten Alternativen wie stationäre Handhabungsgeräte, manuelles Anheben oder mobile Hebeeinrichtung auf dem Fahrzeug gegenüber gestellt werden.



## 9 Summary

In this book, the development of a Versatile Transportation System (VTS) is described. The requirements for a VTS are derived from a study on current transport tasks in the manufacturing industry and in a hospital. It follows the development of a concept for an optimized VTS with respect to the identified requirements. The successful implementation of the developed approach allows the production of very low-cost, autonomously acting vehicles for the transport of payloads with a mass of about 10 kg. The core component of the system is a centrally created world model which is made available to all vehicles by Wi-Fi. For this purpose, stationary cameras are mounted above the operating environment. They are necessary to detect all the Versatile Transportation Vehicles (VTV), all obstacles and all optically marked destination points. In this way, the use of costly laser sensors can be avoided.

The development of the VTS can be divided into two topics. On the one hand, this is the development of the VTV. This includes hardware-specific aspects such as selection of a chassis kinematics and suitable components, finding a concept to overcome height differences and computer-aided design of the vehicle. Similarly, software-specific aspects such as motor control, autonomous navigation, self-localization in the world model and accurate position estimation during docking processes should have to be considered.

On the other hand, for the realization of the VTS, work needs to be done on the stationary components of the VTS. In addition to the installation of the ceiling cameras, in particular software engineering issues concerning the generation of the world model on a stationary computer need to be solved. This includes the modeling of world objects, including the estimation of the respective state vector according to the present world object type. The stationary computer also is processing the pictures taken by the ceiling cameras to detect vision objects and destination markers. Finally, in order to perform the state estimation of the obstacle and vehicle objects, an appropriate allocation of the detected vision objects is necessary. This happens also on the stationary computer.

The concept of the VTS, based on ceiling cameras and world model is in principle applicable to vehicles of any size and mass. However, in accordance with the requirements identified, it is particularly economical when many vehicles operate in confined spaces. Then, a large number of VTV can be supplied with world model data by the use of a small number of ceiling cameras. Therefore, the work deals even with the optimization of VTV in terms of low cost (appropriate chassis type, small size, low cost drive motors ...), but the emphasis is on the reliable world model deployment. The development of the software components required for this purpose on the stationary computer and on the vehicle PCs represents by far the most elaborate part of this thesis work.

On the stationary computer background estimation, background subtraction, contour detection and calculation of enclosing rectangles of vision objects as well as the

destination marker detection has to be performed. Also the estimation of the world object states as well as the assignment of the detected vision objects to the estimated world object positions must be carried out. Finally, it is necessary to visualize the data of the system using a GUI which is also carried out by the stationary computer. The mobile computers of the vehicles need to be equipped with path planner, tracker, trajectory comparator and multimodal position estimators.

The creation of each software component is successfully done. Isolated image processing is quickly executed. The state estimation for vehicle and obstacle objects using Kalman filter provides good results and also the Hungarian method is successfully implemented in a slightly modified manner. For the vehicles a fast, RRT-based, path planner, a stable tracker, fast motor controller and a reliable particle filter for state estimation during docking procedures are available.

However, when integrating all subcomponents into the VTS, the low available computing power and the use of the scripting language Python become noticeable. This results in relatively high cycle times for updating the world model. Using a variety of measures such as the temporary shutdown of the GUI or manually trigger the function for the detection of destination markers, the cycle time can be reduced to 0.2 s. In addition, the dead time that results from executing the complex image processing routines is explicitly included in the world modeling. Thus, the prototypical realization of the overall system is fully operational and meets all set requirements.

As stated in 7.4, the practical testing of the system allows the derivation of measures that should be addressed in further research projects. This includes the reduction of the cycle time of the world model, which is expected to be achieved through the use of faster hardware and the use of compiled programming languages. Furthermore, it is necessary to improve the optical detection and the mathematical modeling of the world objects. Thus, the robustness of the system can be increased by compensation of interference from sunlight, shadows or changing lighting conditions. In order to increase the absolute accuracy of the position determination of the VTV, it is also useful to eliminate systematic errors of the ceiling cameras. In addition to the radial distortion and inaccurate alignment of the ceiling camera, the erroneous projection of the contour of VTV on the floor needs to be compensated. Finally, the extension of the area of operation of the VTS is useful, which can be realized by the installation of additional ceiling cameras.

In addition to these technical developments of the VTS in particular the investigation of novel applications is carried out. The aspects of factory organization stated in 2.1 offer plenty of room for optimizations. This includes combining the advantages of execution-oriented and flow-oriented organization as well as the idea of lowering manufacturing processes to ground level. In this context also concepts are to be assessed in order to overcome differences in height. The solution by means of a ramp, should be compared to alternative approaches using stationary handling robots, manual handling or mobile lifting devices.

The developed system represents a promising starting point. Regarding flexibility and investment costs, it is highly superior to autonomous transportation systems that are currently available on the market.



## 10 Literaturverzeichnis

- [1] F. Adascalitei, Doroftei Ioan, "Practical Applications for Mobile Robots based on Mecanum Wheels - a Systematic Survey," in *Proceedings of the International Conference On Innovations, Recent Trends And Challenges In Mechatronics, Mechanical Engineering And New High-Tech Products Development*, 2011, pp. 112–123.
- [2] Adept Technology GmbH, *Adept Mobile Produkte: Adept Lynx*. Available: <http://www.adept.de/produkte/mobile-roboter/mobile-plattformen/lynx/allgemeines> (2013, Jun. 25).
- [3] A. E. Albert, *Regression and the Moore-Penrose Pseudoinverse*. New York: Academic Press, 1972.
- [4] Allegro, *A3959 SLBT: DMOS Full-Bridge PWM Motor Driver*. Available: <http://www.datasheetcatalog.org/datasheet/allegromicrosystems/3959.pdf> (2012, Sep. 19).
- [5] D. K. Anand, R. B. Zmood, *Introduction to control systems*, 3rd ed. Oxford: Butterworth-Heinemann, 1995.
- [6] API Automated Precision Inc, *Radian: API Product Specifications*. Available: [http://www.apisensor.com/images/ProductLiterature/SpecSheets/Radian%20Spec%20Sheet%20EN0812\\_web.pdf](http://www.apisensor.com/images/ProductLiterature/SpecSheets/Radian%20Spec%20Sheet%20EN0812_web.pdf) (2013, Jul. 26).
- [7] G. Arbeiter, J. Fischer, A. Verl, "3-D-Environment Reconstruction for Mobile Robots using fast- SLAM and Feature Extraction," in *41st International Symposium on Robotics (ISR) and 6th German Conference on Robotics (ROBOTIK)*, 2010, pp. 291–295.
- [8] T. Asfour, K. Regenstien, P. Azad, J. Schroder, A. Bierbaum, N. Vahrenkamp, R. Dillmann, "ARMAR-III: An Integrated Humanoid Platform for Sensory-Motor Control," in *Proceedings of the 6th IEEE-RAS International Conference on Humanoid Robots*, 2006, pp. 169–175.
- [9] S. Avidan, "Ensemble tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 2, pp. 261–271, 2007.
- [10] S. Behnel, R. Bradshaw et al, *Cython: C-Extensions for Python*. Available: <http://cython.org/> (2013, Jan. 11).
- [11] S. Behnel, R. Bradshaw, C. Citro, L. Dalcin, D. Seljebotn, K. Smith, "Cython: The Best of Both Worlds," *Computing in Science & Engineering*, vol. 13, no. 2, pp. 31–39, 2011.
- [12] M. de Berg, *Computational geometry: Algorithms and applications*, 3rd ed. Berlin u. a.: Springer, 2010.
- [13] L. Bergmann, S. Clemens, *Lehrbuch der Experimentalphysik: Optik*, 10th ed. Berlin- New York: Walter de Gruyter, 2004.
- [14] T. Blackburn, D. Vaughan, "Application of linear optimal control and filtering theory to the Saturn V launch vehicle," *IEEE Transactions on Automatic Control*, vol. 16, no. 6, pp. 799–806, 1971.

- [15] J. Bloech, R. Bogaschewsky, U. Buscher, A. Daub, U. Götze, F. Roland, *Einführung in die Produktion*, 6th ed.: Springer, 2008.
- [16] Bosch Rexroth, *Werkstückträger WT 2: 320 x 320 mm*. Available: [www.boschrexroth.com/irj/portal/anonymous/eShop?guest\\_user=anonymousDE&display=catalog&bridgeSelectedCatalog=MTN&bridgePagelId=ts\\_2plus](http://www.boschrexroth.com/irj/portal/anonymous/eShop?guest_user=anonymousDE&display=catalog&bridgeSelectedCatalog=MTN&bridgePagelId=ts_2plus) (2013, May. 18).
- [17] Bosch Rexroth GmbH, *TS 2plus*. Available: [https://www.boschrexroth.com/irj/portal/anonymous/eShop?guest\\_user=anonymousDE&display=catalog&bridgeSelectedCatalog=MTN&bridgePagelId=ts\\_2plus](https://www.boschrexroth.com/irj/portal/anonymous/eShop?guest_user=anonymousDE&display=catalog&bridgeSelectedCatalog=MTN&bridgePagelId=ts_2plus) (2012, Oct. 25).
- [18] G. Bradski, A. Kaehler, *Learning OpenCV*, 1st ed. Beijing u. a.: O'Reilly & Associates, 2008.
- [19] BRETZEL GmbH - Antriebs- und Elektrotechnik, *Lexikon der Antriebstechnik*. Available: <http://www.bretzel-gmbh.de/service/lexikon-der-antriebstechnik.htm> (2012, Oct. 22).
- [20] W. Burgard, D. Fox, D. Hennig, "Estimating the Absolute Position of a Mobile Robot Using Position Probability Grids," in *Proceedings of the 14th National Conference on Artificial Intelligence (AAAI-96)*, 1996, pp. 896–901.
- [21] R. E. Burkard, M. Dell'Amico, S. Martello, *Assignment problems*. Philadelphia: Society for Industrial and Applied Mathematics, 2009.
- [22] G. Campion, G. Bastin, B. Dandrea Novel, "Structural properties and classification of kinematic and dynamic models of wheeled mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 1, pp. 47–62, 1996.
- [23] J. Canny, "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.
- [24] S. Citrin, M. Azimi Sadjadi, "A full-plane block Kalman filter for image restoration," *IEEE Transactions on Image Processing*, vol. 1, no. 4, pp. 488–495, 1992.
- [25] Datenblatt Saugroboter Kobold VR100 von Vorwerk, 21. Available: <http://multivu.prnewswire.com/mnr/prne/vorwerk/52809/docs/52809-Datenblatt.pdf> (2012, Sep. 21).
- [26] Debian Community, *Debian*. Available: <http://www.debian.org/> (2013, Jul. 23).
- [27] R. Demuth, *KATE: Kleine automatische Transporteinheit*. Available: <http://www.goetting.de/news/2012/kate> (2012, Sep. 27).
- [28] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, no. 1, pp. 269–271, 1959.
- [29] *Sicherheit von Flurförderzeugen - Fahrerlose Flurförderzeuge und ihre Systeme*, 1525:1997-12, 1997.
- [30] V. D. Dinh, C. Forgez, K. El Kadri Benkara, G. Friedrich, "Impedance Observer for a Li-Ion Battery Using Kalman Filter," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 8, pp. 3930–3937, 2009.
- [31] S. Edwards, "Leveraging the Open Source Robot Operating System (ROS) for New Industrial Applications," Nov. 2011.
- [32] Europäische Union, *Richtlinie 2006/42/EG Des Europäischen Parlaments und des Rates vom 17. Mai 2006 über Maschinen und zur Änderung der Richtlinie*



- 95/16/EG (Neufassung). Available: [http://eur-lex.europa.eu/LexUriServ/site/de/oj/2006/l\\_157/l\\_15720060609de00240086.pdf](http://eur-lex.europa.eu/LexUriServ/site/de/oj/2006/l_157/l_15720060609de00240086.pdf) (2012, Oct. 09).
- [33] D. Fox, J. Hightower, L. Liao, D. Schulz, G. Borriello, "Bayesian Filters for Location Estimation," *IEEE Pervasive Computing*, vol. 2, no. 3, pp. 24–33, 2009.
- [34] K. Furmans, F. Schönung, C. Nobbe, D. Berbig, "Wandlungsfähiger Materialfluss für die Produktion von morgen," *ATZproduktion*, no. 3, pp. 54–57, 2010.
- [35] J. Gausemeier, T. Schierbaum, R. Dumitrescu, S. Herbrechtsmeier, A. Jungmann, "Miniature robot BeBot: Mechatronic test platform for self-x properties," in *Proceedings of the IEEE International Conference on Industrial Informatics (INDIN)*, 2011, pp. 451–456.
- [36] Gottwald Port Technology GmbH, *Automated Container Transport: Proven Technology from Gottwald*. Available: [http://www.gottwald.com/gottwald/export/gottwaldsite/de/news/pdf/AGV\\_Prospekt\\_uk.pdf](http://www.gottwald.com/gottwald/export/gottwaldsite/de/news/pdf/AGV_Prospekt_uk.pdf) (2013, Jun. 19).
- [37] B. Graf, U. Reiser, M. Hägele, K. Mauz, P. Klein, "Robotic Home Assistant Care-O-bot 3 - Product Vision and Innovation Platform," in *Proceedings of the IEEE Workshop on Advanced Robotics and its Social Impacts - ARSO*, 2009, pp. 139–144.
- [38] Guillemot Corporation, *Hercules Optical Glass*. Available: [http://www.hercules.com/de/webcam/bdd/p/169/hercules-optical-glass-/#product\\_tech\\_carac](http://www.hercules.com/de/webcam/bdd/p/169/hercules-optical-glass-/#product_tech_carac) (2013, Jul. 23).
- [39] W. Günthner, M. ten Hompel, *Internet der Dinge in der Intralogistik*. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 2010.
- [40] J.-S. Gutmann, *Robuste Navigation autonomer mobiler Systeme*. Berlin: Akad. Verl.-Ges. Aka, 2000.
- [41] M. Hägele, N. Blümlein, O. Kleine, *Effirob: Wirtschaftlichkeitsanalysen neuartiger Servicerobotik- Wirtschaftlichkeitsanalysen neuartiger Servicerobotik - Anwendungen und ihre Bedeutung für die Robotik-Entwicklung*. Available: [http://www.ipa.fraunhofer.de/fileadmin/www.ipa.fhg.de/Robotersysteme/Studien/Studie\\_EFFIROB\\_72dpi\\_ol.pdf](http://www.ipa.fraunhofer.de/fileadmin/www.ipa.fhg.de/Robotersysteme/Studien/Studie_EFFIROB_72dpi_ol.pdf) (2013, Oct. 14).
- [42] C. Harris, M. Stephens, "A combined corner and edge detector," in *Proceedings of the 4th Alvey Vision Conference*, 1988, pp. 147–151.
- [43] P. Hart, N. Nilsson, B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Transactions Of Systems Science And Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.
- [44] U. Hilleringmann, *Silizium-Halbleitertechnologie: Grundlagen mikroelektronischer Integrationstechnik ; mit 37 Aufgaben mit Lösungen*, 5th ed. Wiesbaden: Vieweg + Teubner, 2008.
- [45] D. P. Huttenlocher, S. Ullman, *Recognizing Rigid Objects by Aligning Them with an Image*. Ft. Belvoir: Defense Technical Information Center, 1987.
- [46] iEi World, *Single board Computer / Industrial Motherboard: Wafer-LX-R12 Wafer-LX2-R11*. Available: <http://www.comp-mall.de/datenblatt/WAFER-LX.pdf> (2012, Oct. 22).

- [47] Intel Corporation, *Preboot Execution Environment (PXE) Specification: Version 2.1*. Available: <http://download.intel.com/design/archives/wfm/downloads/pxespec.pdf> (2013, Jul. 23).
- [48] International Federation of Robotics, *Service Robots: Provisional definition of Service Robots*. Available: <http://www.ifr.org/service-robots/> (2013, Jun. 26).
- [49] R. Isermann, *Digitale Regelsysteme*, 2nd ed. Berlin, Heidelberg, New York, London, Paris, Tokyo: Springer, 1987.
- [50] T. Jacobs, C. P. Connette, M. Hägele, A. Verl, "Steigerung der Flexibilität mobiler Roboter und Transportsysteme durch den Einsatz kompakter, omnidirektionaler Antriebe," in *Tagungsband Internationales Forum Mechatronik*, 2010.
- [51] M. Jahn, "Ein Beispiel zur Entwicklung kooperierender mobiler Roboter: Konstruktives Design und Steuerungsentwurf," Technische Universität Ilmenau, Ilmenau, 2009.
- [52] B. Jähne, *Digitale Bildverarbeitung*, 6th ed. Berlin u. a.: Springer, 2005.
- [53] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Transaction of the ASME-Journal of Basic Engineering*, vol. 82, no. 4, pp. 35–45, 1960.
- [54] A. Kamagaew, J. Stenzel, A. Nettstrater, M. ten Hompel, "Concept of Cellular Transport Systems in facility logistics," in *Proceedings of the 5th International Conference on Automation, Robotics and Applications (ICARA)*, 2011, pp. 40–45.
- [55] Y. Kanayama, Y. Kimura, F. Miyazaki, Tetsuo Noguchi, "A stable Tracking Control Method for an Autonomous Mobile Robot," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 1990, pp. 384–389.
- [56] P. Kleine-Möllhoff, H. Benad, F. Beilard, M. Esmail, M. Knöll, "Die Batterie als Schlüsseltechnologie für die Elektromobilität der Zukunft Herausforderungen – Potenziale – Ausblick," Hochschule Reutlingen, Reutlingen 2012-3, 2012.
- [57] Y. Koren, J. Borenstein, "Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 1991, pp. 1398–1404.
- [58] E. Kruse, F. Wahl, "Camera-based monitoring system for mobile robot guidance," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, 1998, pp. 1248–1253.
- [59] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, no. 1, pp. 83–97, 1955.
- [60] H. P. Langtangen, *A primer on scientific programming with Python*, 3rd ed. Berlin: Springer, 2012.
- [61] S. M. LaValle, "Rapidly-Exploring Random Trees: A New Tool for Path Planning," Department of Computer Science, Iowa State University, Ames, IA 50011 USA, 1998.
- [62] S. M. LaValle, *Planning algorithms*. Cambridge u. a.: Cambridge Univ. Press, 2006.

- [63] B. Leibe, K. Schindler, L. van Gool, "Coupled Detection and Trajectory Estimation for Multi-Object Tracking," in *Proceedings of the IEEE 11th International Conference on Computer Vision ICCV*, 2007, pp. 1–8.
- [64] C. Liechti, *Welcome to pySerial's documentation — pySerial v2.6 documentation*. Available: <http://pyserial.sourceforge.net/> (2013, Jan. 11).
- [65] R. Löser, "ABC der Batteriesysteme: Informationen rund um die gängigen Akkumulatoren," *Neue Mobilität*, no. 7, pp. 20–21, Apr. 2012.
- [66] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the 7th IEEE International Conference on Computer Vision*, 1999, pp. 1150–1157.
- [67] D. R. Magee, "Tracking multiple vehicles using foreground, background and motion models," *Image and Vision Computing*, vol. 22, no. 2, pp. 143–155, 2004.
- [68] J. Maitin-Shepard, M. Cusumano-Towner, J. Lei, P. Abbeel, "Cloth Grasp Point Detection based on Multiple-View Geometric Cues with Application to Robotic Towel Folding," in *Proceedings of the IEEE Intl. Conference on Robotics and Automation*, 2010.
- [69] R. Mautz, "Application for Venia Legendi in Positioning and Engineering Geodesy," Habilitation Thesis, Institute of Geodesy and Photogrammetry, ETH, Zurich, 2012.
- [70] Mesa Imaging AG. Available: <http://www.mesa-imaging.ch/prodview4k.php?cat=3D%20Camera>.
- [71] Microsoft, *Windows XP*. Available: <http://windows.microsoft.com/de-de/windows/windows-help?os=winxp#windows=windows-xp> (2013, Jul. 23).
- [72] M. L. Mike Phillips, *SIPP: Safe Interval Path Planning for Dynamic Environments*. Available: [http://www.cs.cmu.edu/~maxim/files/sipp\\_icra11.pdf](http://www.cs.cmu.edu/~maxim/files/sipp_icra11.pdf) (2012, Sep. 11).
- [73] C. B. Moler, *Numerical computing with MATLAB*. Philadelphia, Pa: Soc. for Industrial and Applied Mathematics, 2008.
- [74] MVTech Software GmbH, *Halcon: the Power of machine Vision*. Available: <http://www.mvtec.com/halcon/>.
- [75] T. E. Oliphant, *A guide to NumPy*. Trelgol Publishing, 2006.
- [76] OpenCV Developers Team itseez, *OpenCV: Open Source Computer Vision*. Available: <http://opencv.org/> (2013, Jul. 23).
- [77] A. Perera, C. Srinivas, A. Hoogs, G. Brooksby, H. Wensheng, "Multi-Object Tracking Through Simultaneous Long Occlusions and Split-Merge Conditions," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, 2006, pp. 666–673.
- [78] Pichler, *LemonRC LiPo Akkupack 11,1 V; 5,0 AH*. Available: <http://shop.pichler.de/> (2012, Oct. 22).
- [79] M. Pilgrim, *Dive into Python*. Berkeley, Calif: Apress, 2004.
- [80] Qt Project, *Qt*. Available: <http://qt-project.org/> (2013, Jul. 23).

- [81] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, A. Ng, "ROS: an open-source Robot Operating System," in *Proceedings of the Open-Source Software workshop of the International Conference on Robotics and Automation (ICRA)*, 2009.
- [82] B. A. Raichel S. Har-Peled, *The Fréchet distance revisited and extended*. Urbana, IL: University of Illinois, 2011.
- [83] D. B. Reid, "An Algorithm for Tracking Multiple Targets," *IEEE Transactions on Automatic Control*, vol. 24, no. 6, pp. 843–854, 1979.
- [84] Roborealm, *Roborealm: vision for machines*. Available: <http://www.roborealm.com/> (2013, Jul. 24).
- [85] robot-electronics, *EMG30 data*. Available: <http://www.robot-electronics.co.uk/html/emg30.htm> (2012, Sep. 19).
- [86] V. Rückel, "Rechnergestützte Ablaufplanung und Bahngenerierung für kooperierende Industrieroboter," Dissertation, Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik, Friedrich Alexander Universität, Erlangen, 2006.
- [87] Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki, K. Fujimura, "The intelligent ASMIO: System overview and integration," in *Proceedings of the 2002 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems*, 2002, pp. 2478–2483.
- [88] H. Samet, *The design and analysis of spatial data structures*. Reading, Massachusetts: Addison-Wesley, 1990.
- [89] N. X. Semiconductors, *UM10204 I2C-bus specification and user manual*. Available: [http://www.nxp.com/documents/user\\_manual/UM10204.pdf](http://www.nxp.com/documents/user_manual/UM10204.pdf) (2012, Sep. 17).
- [90] M. Sezgin, B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 146–165, 2004.
- [91] Sharp, *GP2Y0A21YK0F: Distance Measuring Sensor Unit Measuring distance: 10 to 80 cm Analog output type*. Available: <http://pdf1.alldatasheet.com/datasheet-pdf/view/412635/SHARP/GP2Y0A21YK0F/+3137WJVPgZSRcITN+tpCziUwy+/datasheet.pdf> (2012, Oct. 22).
- [92] Sick AG. Available: <https://www.mysick.com> (2013, Jun. 14).
- [93] S. Spieker, *Bahnplanung autonomer Transportfahrzeuge*. München: AVM, 2012.
- [94] R. Steinbrecher, *Bildverarbeitung in der Praxis*. München u. a.: Oldenbourg, 1993.
- [95] P. Stephan, I. Dahmann, P. Kraus, G. Frey, "Evaluation of Indoor Positioning Technologies under industrial application conditions in the SmartFactoryKL based on EN ISO 9283," in *Proceedings of the 13th IFAC Symposium on Information Control Problems in Manufacturing (INCOM 09)*, 2009.
- [96] STMicroelectronics, *Automotive fully integrated H-bridge motor driver*. Available: [http://www.st.com/web/en/catalog/sense\\_power/FM1965/SC1039/PF149527?s\\_searchtype=keyword](http://www.st.com/web/en/catalog/sense_power/FM1965/SC1039/PF149527?s_searchtype=keyword) (2012, Oct. 22).

- [97] B. Stroustrup, *The C++ programming language*, 3rd ed. Reading, Massachusetts: Addison-Wesley, 1997.
- [98] I. Sucan, M. Moll, and L. Kavraki, "The Open Motion Planning Library," *IEEE Robotics & Automation Magazine*, vol. 19, no. 4, pp. 72–82, 2012.
- [99] S. Suzuki and K. Abe, "Topological Structural Analysis of Digitized Binary Images by Border Following," *Computer Vision, Graphics and Image Processing*, no. 30, pp. 32–46, 1985.
- [100] Swisslog, *AUTONOMOUS MOBILE ROBOT (AMR) robocourier: HEALTHCARE SOLUTIONS*. Available: <http://www.swisslog.com/hcs-amr-100-amr-overview.pdf> (2012, Sep. 27).
- [101] The Robocup Federation, *Rules 2012: Laws of the RoboCup Small Size League 2011*. Available: [http://robocupssl.cpe.ku.ac.th/\\_media/rules:ssl-rules-2012.pdf](http://robocupssl.cpe.ku.ac.th/_media/rules:ssl-rules-2012.pdf) (11.0.9.2012).
- [102] S. Thrun, W. Burgard, D. Fox, *Probabilistic robotics*. Cambridge, Massachusetts: MIT Press, 2006.
- [103] S. Thrun, M. Montemerlo, D. Stavens, "Stanley, the robot that won the DARPA Grand Challenge," *Journal of Robotik Systems*, vol. 23, no. 9, pp. 661–692, 2006.
- [104] P. A. Tipler, G. Mosca, *Physik für Wissenschaftler und Ingenieure*, 6th ed.: Springer, 2009.
- [105] S. Tosi, *Matplotlib for python developers: Build remarkable publication quality plots the easy way*. Birmingham: Packt Publishing, 2009.
- [106] K. Toyama, J. Krumm, B. Brumitt, B. Meyers, "Wallflower: principles and practice of background maintenance," in *Proceedings of the 7th IEEE International Conference on Computer Vision*, 1999, pp. 255–261.
- [107] Transmotec, *DC Motors, Linear Actuators, Solenoids*. Available: <http://www.transmotec.com/> (2012, Oct. 18).
- [108] Transmotec, *Datasheet PD4266*. Available: <http://www.inovamicro.com/datasheet/PD4266.pdf> (2012, Oct. 19).
- [109] Ubuntu Foundation, *Ubuntu*. Available: <http://www.ubuntu.com/> (2013, Jul. 23).
- [110] G. Ullrich, "Moderne Kliniklogistik im Uni-Klinikum Köln: Jeder Transport unter 15 Minuten," *Krankenhaus Technik+Management*, pp. 42–45, 2004.
- [111] G. Ullrich, *Fahrerlose Transportsysteme: Eine Fibel - mit Praxisanwendungen - zur Technik - für die Planung*. Wiesbaden: Vieweg+Teubner Verlag, 2011.
- [112] G. Ullrich, "Ein Baustein zu fortschrittlicher Kliniklogistik: Fahrerloses Transportsystem rationalisiert Containertransport im Landeskrankenhaus Feldkirch," *f+h Fördern und Heben*, pp. 14–16, 2012.
- [113] G. Ullrich, "Tragende Rolle übernommen: MTU Aero Engines setzt bei der Triebwerksmontage auf FTS," *f+h Fördern und Heben*, no. 5, pp. 34–36, 2013.
- [114] G. Ullrich, *Forum-FTS: Your Partner for Better Logistics*. Available: <http://www.forum-fts.com/> (2013, Jun. 19).
- [115] N. Vahrenkamp, "Bewegungsplanung und sensorgestützte Ausführung für das Greifen auf humanoiden Robotern," KIT, Karlsruhe, Hannover, Karlsruhe, 2011.

- [116] *Fahrerlose Transportsysteme (FTS)*, VDI 2510, 2005.
- [117] VDI-Gesellschaft Produktion und Logistik, *Leitfaden FTS-Sicherheit*. Available: [http://www.vdi.de/fileadmin/vdi\\_de/redakteur\\_dateien/gpl\\_dateien/Leitfaden%20FTS-Sicherheit%20VDI%2020101004%20A%20%282%29.pdf](http://www.vdi.de/fileadmin/vdi_de/redakteur_dateien/gpl_dateien/Leitfaden%20FTS-Sicherheit%20VDI%2020101004%20A%20%282%29.pdf) (2013, Jun. 14).
- [118] R. C. Veltkamp, "Shape Matching: Similarity Measures and Algorithms," in *Proceedings of the International Conference on Shape Modeling and Applications*, 2001, pp. 188–197.
- [119] Vishay, "CNY70: Reflective Optical Sensor with Transistor Output," 83751 Rev 1.8, Aug. 2012.
- [120] G. Welch, G. Bishop, "An Introduction to the Kalman Filter," Department of Computer Science, University of North Carolina at Chapel Hill, 2006.
- [121] M. Weyrich, "Einsatzmöglichkeiten einer 3D-Kamera in der Produktionstechnik am Beispiel der Kinect-Kamera," *Technologien der Fertigungsautomatisierung - Seminarreihe zu ausgewählten Forschungsthemen der industriellen Anwendung*, 2012.
- [122] Willow Garage, *Robot Operating System*. Available: <http://www.ros.org/wiki/> (2013, Jul. 23).
- [123] Willow Garage, *ROS News Blog*. Available: <http://www.ros.org/news/robots> (2013, Jul. 25).
- [124] Willow Garage, *OpenCV Python*. Available: <http://opencv.willowgarage.com/documentation/python/cookbook.html> (2012, Oct. 31).
- [125] W. Wolf, "Entwicklung eines agentenbasierten Steuerungssystems zur Materialflussorganisation im wandelbaren Produktionsumfeld," Dissertation, Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik, Friedrich Alexander Universität, Erlangen, 2008.
- [126] J. P. Womack, D. T. Jones, D. Roos, *The machine that changed the world: How Japan's secret weapon in the global auto wars will revolutionize western industry*, 1st ed. New York, NY: Harper Perennial, 1991.
- [127] S. Zickler, T. Laue, O. Birbach, M. Wongphati, M. Veloso, "SSL-Vision: The Shared Vision System for the RoboCup Small Size League," in *RoboCup International Symposium (RoboCup-09)*, 2009.

# Reihe Fertigungstechnik - Erlangen

[www.mb.uni-erlangen.de/diss](http://www.mb.uni-erlangen.de/diss)

**Band 1 - 52**

**Carl Hanser Verlag, München**

**ab Band 53**

**Meisenbach Verlag, Bamberg**

**45,-- Euro**

**Band 1: Andreas Hemberger**

**Innovationspotentiale in der rechnerintegrierten Produktion  
durch wissensbasierte Systeme**

208 Seiten, 107 Bilder. 1988.

**Band 2: Detlef Classe**

**Beitrag zur Steigerung der Flexibilität automatisierter Montagesysteme  
durch Sensorintegration und erweiterte Steuerungskonzepte**

194 Seiten, 70 Bilder. 1988.

**Band 3: Friedrich-Wilhelm Nolting**

**Projektierung von Montagesystemen**

201 Seiten, 107 Bilder, 1 Tabelle. 1989.

**Band 4: Karsten Schlüter**

**Nutzungsgradsteigerung von Montagesystemen  
durch den Einsatz der Simulationstechnik**

177 Seiten, 97 Bilder. 1989.

**Band 5: Shir-Kuan Lin**

**Aufbau von Modellen zur Lageregelung von Industrierobotern**

168 Seiten, 46 Bilder. 1989.

**Band 6: Rudolf Nuss**

**Untersuchungen zur Bearbeitungsqualität  
im Fertigungssystem Laserstrahlschneiden**

206 Seiten, 115 Bilder, 6 Tabellen. 1989.

**Band 7: Wolfgang Scholz**

**Modell zur datenbankgestützten Planung  
automatisierter Montageanlagen**

194 Seiten, 89 Bilder. 1989.

**Band 8: Hans-Jürgen Wißmeier**

**Beitrag zur Beurteilung des Bruchverhaltens  
von Hartmetall-Fließpreßmatrizen**

179 Seiten, 99 Bilder, 9 Tabellen. 1989.

**Band 9: Rainer Eisele**

**Konzeption und Wirtschaftlichkeit  
von Planungssystemen in der Produktion**

183 Seiten, 86 Bilder. 1990.

**Band 10: Rolf Pfeiffer**

**Technologisch orientierte Montageplanung  
am Beispiel der Schraubtechnik**

216 Seiten, 102 Bilder, 16 Tabellen. 1990.

**Band 11: Herbert Fischer**

**Verteilte Planungssysteme zur Flexibilitätssteigerung  
der rechnerintegrierten Teilefertigung**

201 Seiten, 82 Bilder. 1990.

Band 12: Gerhard Kleineidam  
**CAD/CAP: Rechnergestützte Montagefeinplanung**  
203 Seiten, 107 Bilder. 1990.

Band 13: Frank Vollertsen  
**Pulvermetallurgische Verarbeitung  
eines übereutektoiden verschleißfesten Stahls**  
XIII u. 217 Seiten, 67 Bilder, 34 Tabellen. 1990.

Band 14: Stephan Biermann  
**Untersuchungen zur Anlagen- und Prozeßdiagnostik  
für das Schneiden mit CO<sub>2</sub>-Hochleistungslasern**  
VIII u. 170 Seiten, 93 Bilder, 4 Tabellen. 1991.

Band 15: Uwe Geißler  
**Material- und Datenfluß  
in einer flexiblen Blechbearbeitungszelle**  
124 Seiten, 41 Bilder, 7 Tabellen. 1991.

Band 16: Frank Oswald Hake  
**Entwicklung eines rechnergestützten Diagnosesystems  
für automatisierte Montagezellen**  
XIV u. 166 Seiten, 77 Bilder. 1991.

Band 17: Herbert Reichel  
**Optimierung der Werkzeugbereitstellung  
durch rechnergestützte Arbeitsfolgenbestimmung**  
198 Seiten, 73 Bilder, 2 Tabellen. 1991.

Band 18: Josef Scheller  
**Modellierung und Einsatz von Softwaresystemen  
für rechnergeführte Montagezellen**  
198 Seiten, 65 Bilder. 1991.

Band 19: Arnold vom Ende  
**Untersuchungen zum Biegeumformen mit elastischer Matrize**  
166 Seiten, 55 Bilder, 13 Tabellen. 1991.

Band 20: Joachim Schmid  
**Beitrag zum automatisierten Bearbeiten  
von Keramikguß mit Industrierobotern**  
XIV u. 176 Seiten, 111 Bilder, 6 Tabellen. 1991.

Band 21: Egon Sommer  
**Multiprozessorsteuerung für kooperierende  
Industrieroboter in Montagezellen**  
188 Seiten, 102 Bilder. 1991.

Band 22: Georg Geyer  
**Entwicklung problemspezifischer Verfahrensketten  
in der Montage**  
192 Seiten, 112 Bilder. 1991.

Band 23: Rainer Flohr  
**Beitrag zur optimalen Verbindungstechnik  
in der Oberflächenmontage (SMT)**  
186 Seiten, 79 Bilder. 1991.

Band 24: Alfons Rief  
**Untersuchungen zur Verfahrensfolge Laserstrahlschneiden  
und -schweißen in der Rohkarosseriefertigung**  
VI u. 145 Seiten, 58 Bilder, 5 Tabellen. 1991.

Band 25: Christoph Thim  
**Rechnerunterstützte Optimierung von Materialflußstrukturen  
in der Elektronikmontage durch Simulation**  
188 Seiten, 74 Bilder. 1992.

Band 26: Roland Müller  
**CO<sub>2</sub>-Laserstrahlschneiden  
von kurzglasverstärkten Verbundwerkstoffen**  
141 Seiten, 107 Bilder, 4 Tabellen. 1992.

Band 27: Günther Schäfer  
**Integrierte Informationsverarbeitung bei der Montageplanung**  
195 Seiten, 76 Bilder. 1992.



Band 28: Martin Hoffmann  
**Entwicklung einer CAD/CAM-Prozesskette  
für die Herstellung von Blechbiegeteilen**  
149 Seiten, 89 Bilder. 1992.

Band 29: Peter Hoffmann  
**Verfahrensfolge Laserstrahlschneiden und –schweißen :  
Prozessführung und Systemtechnik in der 3D–Laserstrahlbearbeitung  
von Blechformteilen**  
186 Seiten, 92 Bilder, 10 Tabellen. 1992.

Band 30: Olaf Schrödel  
**Flexible Werkstattsteuerung mit objektorientierten Softwarestrukturen**  
180 Seiten, 84 Bilder. 1992.

Band 31: Hubert Reinisch  
**Planungs– und Steuerungswerkzeuge  
zur impliziten Geräteprogrammierung in Roboterzellen**  
XI u. 212 Seiten, 112 Bilder. 1992.

Band 32: Brigitte Bärnreuther  
**Ein Beitrag zur Bewertung des Kommunikationsverhaltens  
von Automatisierungsgeräten in flexiblen Produktionszellen**  
XI u. 179 Seiten, 71 Bilder. 1992.

Band 33: Joachim Hutfless  
**Laserstrahlregelung und Optikdiagnostik  
in der Strahlführung einer CO<sub>2</sub>-Hochleistungslaseranlage**  
175 Seiten, 70 Bilder, 17 Tabellen. 1993.

Band 34: Uwe Günzel  
**Entwicklung und Einsatz eines Simulationsverfahrens für operative  
und strategische Probleme der Produktionsplanung und –steuerung**  
XIV u. 170 Seiten, 66 Bilder, 5 Tabellen. 1993.

Band 35: Bertram Ehmann  
**Operatives Fertigungscontrolling durch Optimierung  
auftragsbezogener Bearbeitungsabläufe in der Elektronikfertigung**  
XV u. 167 Seiten, 114 Bilder. 1993.

Band 36: Harald Kolléra  
**Entwicklung eines benutzerorientierten Werkstattprogrammiersystems  
für das Laserstrahlschneiden**  
129 Seiten, 66 Bilder, 1 Tabelle. 1993.

Band 37: Stephanie Abels  
**Modellierung und Optimierung von Montageanlagen  
in einem integrierten Simulationssystem**  
188 Seiten, 88 Bilder. 1993.

Band 38: Robert Schmidt–Hebbel  
**Laserstrahlbohren durchflußbestimmender Durchgangslöcher**  
145 Seiten, 63 Bilder, 11 Tabellen. 1993.

Band 39: Norbert Lutz  
**Oberflächenfeinbearbeitung keramischer Werkstoffe  
mit XeCl–Excimerlaserstrahlung**  
187 Seiten, 98 Bilder, 29 Tabellen. 1994.

Band 40: Konrad Grampp  
**Rechnerunterstützung bei Test und Schulung  
an Steuerungssoftware von SMD–Bestücklinien**  
178 Seiten, 88 Bilder. 1995.

Band 41: Martin Koch  
**Wissensbasierte Unterstützung der Angebotsbearbeitung  
in der Investitionsgüterindustrie**  
169 Seiten, 68 Bilder. 1995.

Band 42: Armin Gropp  
**Anlagen– und Prozessdiagnostik  
beim Schneiden mit einem gepulsten Nd:YAG–Laser**  
160 Seiten, 88 Bilder, 7 Tabellen. 1995.

Band 43: Werner Heckel  
**Optische 3D-Konturerfassung und on-line Biegewinkelmessung  
mit dem Lichtschnittverfahren**  
149 Seiten, 43 Bilder, 11 Tabellen. 1995.

Band 44: Armin Rothhaupt  
**Modulares Planungssystem  
zur Optimierung der Elektronikfertigung**  
180 Seiten, 101 Bilder. 1995.

Band 45: Bernd Zöllner  
**Adaptive Diagnose in der Elektronikproduktion**  
195 Seiten, 74 Bilder, 3 Tabellen. 1995.

Band 46: Bodo Vormann  
**Beitrag zur automatisierten Handhabungsplanung  
komplexer Blechbiegeteile**  
126 Seiten, 89 Bilder, 3 Tabellen. 1995.

Band 47: Peter Schnepf  
**Zielkostenorientierte Montageplanung**  
144 Seiten, 75 Bilder. 1995.

Band 48: Rainer Klotzbücher  
**Konzept zur rechnerintegrierten Materialversorgung  
in flexiblen Fertigungssystemen**  
156 Seiten, 62 Bilder. 1995.

Band 49: Wolfgang Greska  
**Wissensbasierte Analyse und Klassifizierung von Blechteilen**  
144 Seiten, 96 Bilder. 1995.

Band 50: Jörg Franke  
**Integrierte Entwicklung neuer Produkt- und Produktionstechnologien  
für räumliche spritzgegossene Schaltungsträger (3-D MID)**  
196 Seiten, 86 Bilder, 4 Tabellen. 1995.

Band 51: Franz-Josef Zeller  
**Sensorplanung und schnelle Sensorregelung für Industrieroboter**  
190 Seiten, 102 Bilder, 9 Tabellen. 1995.

Band 52: Michael Solvie  
**Zeitbehandlung und Multimedia-Unterstützung  
in Feldkommunikationssystemen**  
200 Seiten, 87 Bilder, 35 Tabellen. 1996.

Band 53: Robert Hopperdietzel  
**Reengineering in der Elektro- und Elektronikindustrie**  
180 Seiten, 109 Bilder, 1 Tabelle. 1996.  
ISBN 3-87525-070-2

Band 54: Thomas Rebhan  
**Beitrag zur Mikromaterialbearbeitung mit Excimerlasern –  
Systemkomponenten und Verfahrensoptimierungen**  
148 Seiten, 61 Bilder, 10 Tabellen. 1996.  
ISBN 3-87525-075-3

Band 55: Henning Hanebuth  
**Laserstrahlhartlöten mit Zweistrahlschmelztechnik**  
157 Seiten, 58 Bilder, 11 Tabellen. 1996.  
ISBN 3-87525-074-5

Band 56: Uwe Schönherr  
**Steuerung und Sensordatenintegration für flexible Fertigungszellen  
mit kooperierenden Robotern**  
188 Seiten, 116 Bilder, 3 Tabellen. 1996.  
ISBN 3-87525-076-1

Band 57: Stefan Holzer  
**Berührungslose Formgebung mit Laserstrahlung**  
162 Seiten, 69 Bilder, 11 Tabellen. 1996.  
ISBN 3-87525-079-6

Band 58: Markus Schultz  
**Fertigungsqualität beim 3D–Laserstrahlschweißen  
von Blechformteilen**  
165 Seiten, 88 Bilder, 9 Tabellen. 1997.  
ISBN 3-87525-080-X

Band 59: Thomas Krebs  
**Integration elektromechanischer CA–Anwendungen  
über einem STEP–Produktmodell**  
198 Seiten, 58 Bilder, 8 Tabellen. 1997.  
ISBN 3-87525-081-8

Band 60: Jürgen Sturm  
**Prozeßintegrierte Qualitätssicherung  
in der Elektronikproduktion**  
167 Seiten, 112 Bilder, 5 Tabellen. 1997.  
ISBN 3-87525-082-6

Band 61: Andreas Brand  
**Prozesse und Systeme zur Bestückung  
räumlicher elektronischer Baugruppen (3D-MID)**  
182 Seiten, 100 Bilder. 1997.  
ISBN 3-87525-087-7

Band 62: Michael Kauf  
**Regelung der Laserstrahlleistung und der Fokusparameter  
einer CO<sub>2</sub>-Hochleistungslaseranlage**  
140 Seiten, 70 Bilder, 5 Tabellen. 1997.  
ISBN 3-87525-083-4

Band 63: Peter Steinwasser  
**Modulares Informationsmanagement  
in der integrierten Produkt– und Prozeßplanung**  
190 Seiten, 87 Bilder. 1997.  
ISBN 3-87525-084-2

Band 64: Georg Liedl  
**Integriertes Automatisierungskonzept  
für den flexiblen Materialfluß in der Elektronikproduktion**  
196 Seiten, 96 Bilder, 3 Tabellen. 1997.  
ISBN 3-87525-086-9

Band 65: Andreas Otto  
**Transiente Prozesse beim Laserstrahlschweißen**  
132 Seiten, 62 Bilder, 1 Tabelle. 1997.  
ISBN 3-87525-089-3

Band 66: Wolfgang Blöchl  
**Erweiterte Informationsbereitstellung an offenen CNC–Steuerungen  
zur Prozeß– und Programmoptimierung**  
168 Seiten, 96 Bilder. 1997.  
ISBN 3-87525-091-5

Band 67: Klaus–Uwe Wolf  
**Verbesserte Prozeßführung und Prozeßplanung  
zur Leistungs– und Qualitätssteigerung beim Spulenwickeln**  
186 Seiten, 125 Bilder. 1997.  
ISBN 3-87525-092-3

Band 68: Frank Backes  
**Technologieorientierte Bahnplanung für die 3D–Laserstrahlbearbeitung**  
138 Seiten, 71 Bilder, 2 Tabellen. 1997.  
ISBN 3-87525-093-1

Band 69: Jürgen Kraus  
**Laserstrahlumformen von Profilen**  
137 Seiten, 72 Bilder, 8 Tabellen. 1997.  
ISBN 3-87525-094-X

Band 70: Norbert Neubauer  
**Adaptive Strahlführungen für CO<sub>2</sub>-Lasieranlagen**  
120 Seiten, 50 Bilder, 3 Tabellen. 1997.  
ISBN 3-87525-095-8

Band 71: Michael Steber  
**Prozeßoptimierter Betrieb flexibler Schraubstationen  
in der automatisierten Montage**  
168 Seiten, 78 Bilder, 3 Tabellen. 1997.  
ISBN 3-87525-096-6

Band 72: Markus Pfestorf  
**Funktionale 3D-Oberflächenkenngrößen in der Umformtechnik**  
162 Seiten, 84 Bilder, 15 Tabellen. 1997.  
ISBN 3-87525-097-4

Band 73: Volker Franke  
**Integrierte Planung und Konstruktion  
von Werkzeugen für die Blegebearbeitung**  
143 Seiten, 81 Bilder. 1998.  
ISBN 3-87525-098-2

Band 74: Herbert Scheller  
**Automatisierte Demontagesysteme und recyclinggerechte  
Produktgestaltung elektronischer Baugruppen**  
184 Seiten, 104 Bilder, 17 Tabellen. 1998.  
ISBN 3-87525-099-0

Band 75: Arthur Meßner  
**Kaltmassivumformung metallischer Kleinstteile  
– Werkstoffverhalten, Wirkflächenreibung, Prozeßauslegung**  
164 Seiten, 92 Bilder, 14 Tabellen. 1998.  
ISBN 3-87525-100-8

Band 76: Mathias Glasmacher  
**Prozeß- und Systemtechnik zum Laserstrahl-Mikroschweißen**  
184 Seiten, 104 Bilder, 12 Tabellen. 1998.  
ISBN 3-87525-101-6

Band 77: Michael Schwind  
**Zerstörungsfreie Ermittlung mechanischer Eigenschaften  
von Feinblechen mit dem Wirbelstromverfahren**  
124 Seiten, 68 Bilder, 8 Tabellen. 1998.  
ISBN 3-87525-102-4

Band 78: Manfred Gerhard  
**Qualitätssteigerung in der Elektronikproduktion durch Optimierung  
der Prozeßführung beim Löten komplexer Baugruppen**  
179 Seiten, 113 Bilder, 7 Tabellen. 1998.  
ISBN 3-87525-103-2

Band 79: Elke Rauh  
**Methodische Einbindung der Simulation  
in die betrieblichen Planungs- und Entscheidungsabläufe**  
192 Seiten, 114 Bilder, 4 Tabellen. 1998.  
ISBN 3-87525-104-0

Band 80: Sorin Niederkorn  
**Meßeinrichtung zur Untersuchung der Wirkflächenreibung  
bei umformtechnischen Prozessen**  
99 Seiten, 46 Bilder, 6 Tabellen. 1998.  
ISBN 3-87525-105-9

Band 81: Stefan Schubert  
**Regelung der Fokussage beim Schweißen mit CO<sub>2</sub>-Hochleistungslasern  
unter Einsatz von adaptiven Optiken**  
140 Seiten, 64 Bilder, 3 Tabellen. 1998.  
ISBN 3-87525-106-7

Band 82: Armando Walter Colombo  
**Development and Implementation of Hierarchical Control Structures  
of Flexible Production Systems Using High Level Petri Nets**  
216 Seiten, 86 Bilder. 1998.  
ISBN 3-87525-109-1

Band 83: Otto Meedt  
**Effizienzsteigerung bei Demontage und Recycling  
durch flexible Demontagetechnologien und optimierte Produktgestaltung**  
186 Seiten, 103 Bilder. 1998.  
ISBN 3-87525-108-3

- Band 84: Knuth Götz  
**Modelle und effiziente Modellbildung zur Qualitätssicherung in der Elektronikproduktion**  
 212 Seiten, 129 Bilder, 24 Tabellen. 1998.  
 ISBN 3-87525-112-1
- Band 85: Ralf Luchs  
**Einsatzmöglichkeiten leitender Klebstoffe zur zuverlässigen Kontaktierung elektronischer Bauelemente in der SMT**  
 176 Seiten, 126 Bilder, 30 Tabellen. 1998.  
 ISBN 3-87525-113-7
- Band 86: Frank Pöhlau  
**Entscheidungsgrundlagen zur Einführung räumlicher spritzgegossener Schaltungsträger (3-D MID)**  
 144 Seiten, 99 Bilder. 1999.  
 ISBN 3-87525-114-8
- Band 87: Roland T. A. Kals  
**Fundamentals on the miniaturization of sheet metal working processes**  
 128 Seiten, 58 Bilder, 11 Tabellen. 1999.  
 ISBN 3-87525-115-6
- Band 88: Gerhard Luhn  
**Implizites Wissen und technisches Handeln am Beispiel der Elektronikproduktion**  
 252 Seiten, 61 Bilder, 1 Tabelle. 1999.  
 ISBN 3-87525-116-4
- Band 89: Axel Sprenger  
**Adaptives Streckbiegen von Aluminium-Strangpreßprofilen**  
 114 Seiten, 63 Bilder, 4 Tabellen. 1999.  
 ISBN 3-87525-117-2
- Band 90: Hans-Jörg Pucher  
**Untersuchungen zur Prozeßfolge Umformen, Bestücken und Laserstrahllöten von Mikrokontakten**  
 158 Seiten, 69 Bilder, 9 Tabellen. 1999.  
 ISBN 3-87525-119-9
- Band 91: Horst Arnet  
**Profilbiegen mit kinematischer Gestalterzeugung**  
 128 Seiten, 67 Bilder, 7 Tabellen. 1999.  
 ISBN 3-87525-120-2
- Band 92: Doris Schubart  
**Prozeßmodellierung und Technologieentwicklung beim Abtragen mit CO<sub>2</sub>-Laserstrahlung**  
 133 Seiten, 57 Bilder, 13 Tabellen. 1999.  
 ISBN 3-87525-122-9
- Band 93: Adrianus L. P. Coremans  
**Laserstrahlsintern von Metallpulver – Prozeßmodellierung, Systemtechnik, Eigenschaften laserstrahlgesinterter Metallkörper**  
 184 Seiten, 108 Bilder, 12 Tabellen. 1999.  
 ISBN 3-87525-124-5
- Band 94: Hans-Martin Biehler  
**Optimierungskonzepte für Qualitätsdatenverarbeitung und Informationsbereitstellung in der Elektronikfertigung**  
 194 Seiten, 105 Bilder. 1999.  
 ISBN 3-87525-126-1
- Band 95: Wolfgang Becker  
**Oberflächen Ausbildung und tribologische Eigenschaften excimerlaserstrahlbearbeiteter Hochleistungskeramiken**  
 175 Seiten, 71 Bilder, 3 Tabellen. 1999.  
 ISBN 3-87525-127-X
- Band 96: Philipp Hein  
**Innenhochdruck-Umformen von Blechpaaren: Modellierung, Prozeßauslegung und Prozeßführung**  
 129 Seiten, 57 Bilder, 7 Tabellen. 1999.  
 ISBN 3-87525-128-8

Band 97: Gunter Beitingher  
**Herstellungs- und Prüfverfahren  
für thermoplastische Schaltungsträger**  
169 Seiten, 92 Bilder, 20 Tabellen. 1999.  
ISBN 3-87525-129-6

Band 98: Jürgen Knobloch  
**Beitrag zur rechnerunterstützten verursachungsgerechten Angebotskalkulation von  
Blechteilen mit Hilfe wissensbasierter Methoden**  
155 Seiten, 53 Bilder, 26 Tabellen. 1999.  
ISBN 3-87525-130-X

Band 99: Frank Breitenbach  
**Bildverarbeitungssystem zur Erfassung der Anschlußgeometrie  
elektronischer SMT-Bauelemente**  
147 Seiten, 92 Bilder, 12 Tabellen. 2000.  
ISBN 3-87525-131-8

Band 100: Bernd Falk  
**Simulationsbasierte Lebensdauervorhersage  
für Werkzeuge der Kaltmassivumformung**  
134 Seiten, 44 Bilder, 15 Tabellen. 2000.  
ISBN 3-87525-136-9

Band 101: Wolfgang Schlögl  
**Integriertes Simulationsdaten-Management  
für Maschinenentwicklung und Anlagenplanung**  
169 Seiten, 101 Bilder, 20 Tabellen. 2000.  
ISBN 3-87525-137-7

Band 102: Christian Hinsel  
**Ermüdungsbruchversagen hartstoffbeschichteter  
Werkzeugstähle in der Kaltmassivumformung**  
130 Seiten, 80 Bilder, 14 Tabellen. 2000.  
ISBN 3-87525-138-5

Band 103: Stefan Bobbert  
**Simulationsgestützte Prozessauslegung  
für das Innenhochdruck-Umformen von Blechpaaren**  
123 Seiten, 77 Bilder. 2000.  
ISBN 3-87525-145-8

Band 104: Harald Rottbauer  
**Modulares Planungswerkzeug  
zum Produktionsmanagement in der Elektronikproduktion**  
166 Seiten, 106 Bilder. 2001.  
ISBN 3-87525-139-3

Band 105: Thomas Hennige  
**Flexible Formgebung von Blechen durch Laserstrahlumformen**  
119 Seiten, 50 Bilder. 2001.  
ISBN 3-87525-140-7

Band 106: Thomas Menzel  
**Wissensbasierte Methoden für die rechnergestützte Charakterisierung  
und Bewertung innovativer Fertigungsprozesse**  
152 Seiten, 71 Bilder. 2001.  
ISBN 3-87525-142-3

Band 107: Thomas Stöckel  
**Kommunikationstechnische Integration der Prozeßebe-  
ne in Produktionssysteme durch Middleware-Frameworks**  
147 Seiten, 65 Bilder, 5 Tabellen. 2001.  
ISBN 3-87525-143-1

Band 108: Frank Pitter  
**Verfügbarkeitssteigerung von Werkzeugmaschinen  
durch Einsatz mechatronischer Sensorlösungen**  
158 Seiten, 131 Bilder, 8 Tabellen. 2001.  
ISBN 3-87525-144-X

Band 109: Markus Korneli  
**Integration lokaler CAP-Systeme  
in einen globalen Fertigungsdatenverbund**  
121 Seiten, 53 Bilder, 11 Tabellen. 2001.  
ISBN 3-87525-146-6

Band 110: Burkhard Müller  
**Laserstrahljustieren mit Excimer-Lasern – Prozeßparameter und Modelle zur Aktorkonstruktion**  
128 Seiten, 36 Bilder, 9 Tabellen. 2001  
ISBN 3-87525-159-8

Band 111: Jürgen Göhringer  
**Integrierte Telediagnose via Internet zum effizienten Service von Produktionssystemen**  
178 Seiten, 98 Bilder, 5 Tabellen. 2001.  
ISBN 3-87525-147-4

Band 112: Robert Feuerstein  
**Qualitäts- und kosteneffiziente Integration neuer Bauelementetechnologien in die Flachbaugruppenfertigung**  
161 Seiten, 99 Bilder, 10 Tabellen. 2001.  
ISBN 3-87525-151-2

Band 113: Marcus Reichenberger  
**Eigenschaften und Einsatzmöglichkeiten alternativer Elektroniklote in der Oberflächenmontage (SMT)**  
165 Seiten, 97 Bilder, 18 Tabellen. 2001.  
ISBN 3-87525-152-0

Band 114: Alexander Huber  
**Justieren vormontierter Systeme mit dem Nd:YAG-Laser unter Einsatz von Aktoren**  
122 Seiten, 58 Bilder, 5 Tabellen. 2001.  
ISBN 3-87525-153-9

Band 115: Sami Krimi  
**Analyse und Optimierung von Montagesystemen in der Elektronikproduktion**  
155 Seiten, 88 Bilder, 3 Tabellen. 2001.  
ISBN 3-87525-157-1

Band 116: Marion Merklein  
**Laserstrahlumformen von Aluminiumwerkstoffen - Beeinflussung der Mikrostruktur und der mechanischen Eigenschaften**  
122 Seiten, 65 Bilder, 15 Tabellen. 2001.  
ISBN 3-87525-156-3

Band 117: Thomas Collisi  
**Ein informationslogistisches Architekturkonzept zur Akquisition simulationsrelevanter Daten**  
181 Seiten, 105 Bilder, 7 Tabellen. 2002.  
ISBN 3-87525-164-4

Band 118: Markus Koch  
**Rationalisierung und ergonomische Optimierung im Innenausbau durch den Einsatz moderner Automatisierungstechnik**  
176 Seiten, 98 Bilder, 9 Tabellen. 2002.  
ISBN 3-87525-165-2

Band 119: Michael Schmidt  
**Prozeßregelung für das Laserstrahl-Punktschweißen in der Elektronikproduktion**  
152 Seiten, 71 Bilder, 3 Tabellen. 2002.  
ISBN 3-87525-166-0

Band 120: Nicolas Tiesler  
**Grundlegende Untersuchungen zum Fließpressen metallischer Kleinstteile**  
126 Seiten, 78 Bilder, 12 Tabellen. 2002.  
ISBN 3-87525-175-X

Band 121: Lars Pursche  
**Methoden zur technologieorientierten Programmierung für die 3D-Lasermikrobearbeitung**  
111 Seiten, 39 Bilder, 0 Tabellen. 2002.  
ISBN 3-87525-183-0

Band 122: Jan-Oliver Brassel  
**Prozeßkontrolle beim Laserstrahl-Mikroschweißen**  
148 Seiten, 72 Bilder, 12 Tabellen. 2002.  
ISBN 3-87525-181-4

Band 123: Mark Geisel

**Prozeßkontrolle und –steuerung beim Laserstrahlschweißen  
mit den Methoden der nichtlinearen Dynamik**

135 Seiten, 46 Bilder, 2 Tabellen. 2002.  
ISBN 3-87525-180-6

Band 124: Gerd Eßer

**Laserstrahlunterstützte Erzeugung metallischer Leiterstrukturen auf  
Thermoplastsubstraten für die MID-Technik**

148 Seiten, 60 Bilder, 6 Tabellen. 2002.  
ISBN 3-87525-171-7

Band 125: Marc Fleckenstein

**Qualität laserstrahl-gefügter Mikroverbindungen  
elektronischer Kontakte**

159 Seiten, 77 Bilder, 7 Tabellen. 2002.  
ISBN 3-87525-170-9

Band 126: Stefan Kaufmann

**Grundlegende Untersuchungen zum Nd:YAG- Laserstrahlfügen  
von Silizium für Komponenten der Optoelektronik**

159 Seiten, 100 Bilder, 6 Tabellen. 2002.  
ISBN 3-87525-172-5

Band 127: Thomas Fröhlich

**Simultanes Löten von Anschlußkontakten elektronischer Bauelemente  
mit Diodenlaserstrahlung**

143 Seiten, 75 Bilder, 6 Tabellen. 2002.  
ISBN 3-87525-186-5

Band 128: Achim Hofmann

**Erweiterung der Formgebungsgrenzen beim Umformen von  
Aluminiumwerkstoffen durch den Einsatz prozessangepasster Platinen**

113 Seiten, 58 Bilder, 4 Tabellen  
ISBN 3-87525-182-2

Band 129: Ingo Kriebitzsch

**3 - D MID Technologie in der Automobilelektronik**

129 Seiten, 102 Bilder, 10 Tabellen. 2002.  
ISBN 3-87525-169-5

Band 130: Thomas Pohl

**Fertigungsqualität und Umformbarkeit laserstrahlschweißter  
Formplatinen aus Aluminiumlegierungen**

133 Seiten, 93 Bilder, 12 Tabellen. 2002  
ISBN 3-87525-173-3

Band 131: Matthias Wenk

**Entwicklung eines konfigurierbaren Steuerungssystems für die  
flexible Sensorführung von Industrierobotern**

167 Seiten, 85 Bilder, 1 Tabelle. 2002.  
ISBN 3-87525-174-1

Band 132: Matthias Negendanck

**Neue Sensorik und Aktorik für Bearbeitungsköpfe  
zum Laserstrahlschweißen**

116 Seiten, 60 Bilder, 14 Tabellen  
ISBN 3-87525-184-9

Band 133: Oliver Kreis

**Integrierte Fertigung –**

**Verfahrensintegration durch Innenhochdruck-Umformen, Trennen und  
Laserstrahlschweißen in einem Werkzeug sowie ihre tele- und multimediale Präsentation**

167 Seiten, 90 Bilder, 43 Tabellen  
ISBN 3-87525-176-8

Band 134: Stefan Trautner

**Technische Umsetzung produktbezogener Instrumente der  
Umweltpolitik bei Elektro- und Elektronikgeräten**

179 Seiten, 92 Bilder, 11 Tabellen. 2002.  
ISBN 3-87525-177-6



Band 135: Roland Meier

**Strategien für einen produktorientierten Einsatz räumlicher  
spritzgegossener Schaltungsträger (3-D MID)**

155 Seiten, 88 Bilder, 14 Tabellen. 2002.  
ISBN 3-87525-178-4

Band 136: Jürgen Wunderlich

**Kostensimulation – Simulationsbasierte Wirtschaftlichkeitsregelung  
komplexer Produktionssysteme**

202 Seiten, 119 Bilder, 17 Tabellen. 2002.  
ISBN 3-87525-179-2

Band 137: Stefan Novotny

**Innenhochdruck-Umformen von Blechen aus Aluminium- und  
Magnesiumlegierungen bei erhöhter Temperatur**

132 Seiten, 82 Bilder, 6 Tabellen. 2002.  
ISBN 3-87525-185-7

Band 138: Andreas Licha

**Flexible Montageautomatisierung zur Komplettmontage flächenhafter  
Produktstrukturen durch kooperierende Industrieroboter**

158 Seiten, 87 Bilder, 8 Tabellen. 2003.  
ISBN 3-87525-189-X

Band 139: Michael Eisenbarth

**Beitrag zur Optimierung der Aufbau- und Verbindungstechnik  
für mechatronische Baugruppen**

207 Seiten, 141 Bilder, 9 Tabellen. 2003.  
ISBN 3-87525-190-3

Band 140: Frank Christoph

**Durchgängige simulationsgestützte Planung von  
Fertigungseinrichtungen der Elektronikproduktion**

187 Seiten, 107 Bilder, 9 Tabellen. 2003.  
ISBN 3-87525-191-1

Band 141: Hinnerk Hagenah

**Simulationsbasierte Bestimmung der zu erwartenden  
Mäßhaltigkeit für das Blechbiegen**

131 Seiten, 36 Bilder, 26 Tabellen. 2003.  
ISBN 3-87525-192-X

Band 142: Ralf Eckstein

**Scherschneiden und Biegen metallischer Kleinstteile –  
Materialeinfluss und Materialverhalten**

148 Seiten, 71 Bilder, 19 Tabellen. 2003.  
ISBN 3-87525-193-8

Band 143: Frank H. Meyer-Pittroff

**Excimerlaserstrahlbiegen dünner metallischer Folien  
mit homogener Lichtlinie**

138 Seiten, 60 Bilder, 16 Tabellen. 2003.  
ISBN 3-87525-196-2

Band 144: Andreas Kach

**Rechnergestützte Anpassung von Laserstrahlchneidbahnen  
an Bauteilabweichungen**

139 Seiten, 69 Bilder, 11 Tabellen. 2004.  
ISBN 3-87525-197-0

Band 145: Stefan Hierl

**System- und Prozeßtechnik für das simultane Löten mit  
Diodenlaserstrahlung von elektronischen Bauelementen**

124 Seiten, 66 Bilder, 4 Tabellen. 2004.  
ISBN 3-87525-198-9

Band 146: Thomas Neudecker

**Tribologische Eigenschaften keramischer Blechumformwerkzeuge-  
Einfluss einer Oberflächenendbearbeitung mittels Excimerlaserstrahlung**

166 Seiten, 75 Bilder, 26 Tabellen. 2004.  
ISBN 3-87525-200-4

Band 147: Ulrich Wenger

**Prozessoptimierung in der Wickeltechnik durch innovative maschinenbauliche und regelungstechnische Ansätze**

132 Seiten, 88 Bilder, 0 Tabellen. 2004.  
ISBN 3-87525-203-9

Band 148: Stefan Slama

**Effizienzsteigerung in der Montage durch marktorientierte Montagestrukturen und erweiterte Mitarbeiterkompetenz**

188 Seiten, 125 Bilder, 0 Tabellen. 2004.  
ISBN 3-87525-204-7

Band 149: Thomas Wurm

**Laserstrahljustieren mittels Aktoren –**

**Entwicklung von Konzepten und Methoden für die rechnerunterstützte Modellierung und Optimierung von komplexen Aktorsystemen in der Mikrotechnik**

122 Seiten, 51 Bilder, 9 Tabellen. 2004.  
ISBN 3-87525-206-3

Band 150: Martino Celeghini

**Wirkmedienbasierte Blechumformung:**

**Grundlagenuntersuchungen zum Einfluss von Werkstoff und Bauteilgeometrie**

146 Seiten, 77 Bilder, 6 Tabellen. 2004.  
ISBN 3-87525-207-1

Band 151: Ralph Hohenstein

**Entwurf hochdynamischer Sensor- und Regelsysteme für die adaptive Laserbearbeitung**

282 Seiten, 63 Bilder, 16 Tabellen. 2004.  
ISBN 3-87525-210-1

Band 152: Angelika Hutterer

**Entwicklung prozessüberwachender Regelkreise für flexible Formgebungsprozesse**

149 Seiten, 57 Bilder, 2 Tabellen. 2005.  
ISBN 3-87525-212-8

Band 153: Emil Egerer

**Massivumformen metallischer Kleinstteile bei erhöhter Prozesstemperatur**

158 Seiten, 87 Bilder, 10 Tabellen. 2005.  
ISBN 3-87525-213-6

Band 154: Rüdiger Holzmann

**Strategien zur nachhaltigen Optimierung von Qualität und Zuverlässigkeit in der Fertigung hochintegrierter Flachbaugruppen**

186 Seiten, 99 Bilder, 19 Tabellen. 2005.  
ISBN 3-87525-217-9

Band 155: Marco Nock

**Biegeumformen mit Elastomerwerkzeugen**

Modellierung, Prozessauslegung und Abgrenzung des Verfahrens am Beispiel des Rohrbiegens

164 Seiten, 85 Bilder, 13 Tabellen. 2005.  
ISBN 3-87525-218-7

Band 156: Frank Niebling

**Qualifizierung einer Prozesskette zum Laserstrahlintern metallischer Bauteile**

148 Seiten, 89 Bilder, 3 Tabellen. 2005.  
ISBN 3-87525-219-5

Band 157: Markus Meiler

**Großserientauglichkeit**

**trockenschmierstoffbeschichteter**

**Aluminiumbleche im Presswerk**

Grundlegende Untersuchungen zur Tribologie, zum Umformverhalten und Bauteilversuche

104 Seiten, 57 Bilder, 21 Tabellen. 2005.  
ISBN 3-87525-221-7

Band 158: Agus Sutanto

**Solution Approaches for Planning of Assembly Systems  
in Three-Dimensional Virtual Environments**

169 Seiten, 98 Bilder, 3 Tabellen. 2005.  
ISBN 3-87525-220-9

Band 159: Matthias Boiger

**Hochleistungssysteme für die Fertigung elektronischer Baugruppen  
auf der Basis flexibler Schaltungsträger**

175 Seiten, 111 Bilder, 8 Tabellen. 2005.  
ISBN 3-87525-222-5

Band 160: Matthias Pitz

**Laserunterstütztes Biegen höchstfester Mehrphasenstähle**

120 Seiten, 73 Bilder, 11 Tabellen. 2005.  
ISBN 3-87525-223-3

Band 161: Meik Vahl

**Beitrag zur gezielten Beeinflussung des Werkstoffflusses  
beim Innenhochdruck-Umformen von Blechen**

165 Seiten, 94 Bilder, 15 Tabellen. 2005.  
ISBN 3-87525-224-1

Band 162: Peter K. Kraus

**Plattformstrategien – Realisierung einer varianz- und  
kostenoptimierten Wertschöpfung**

181 Seiten, 95 Bilder, 0 Tabellen. 2005.  
ISBN 3-87525-226-8

Band 163: Adrienn Cser

**Laserstrahlschmelzabtrag – Prozessanalyse und -modellierung**

146 Seiten, 79 Bilder, 3 Tabellen. 2005.  
ISBN 3-87525-227-6

Band 164: Markus C. Hahn

**Grundlegende Untersuchungen zur Herstellung von  
Leichtbauverbundstrukturen mit Aluminiumschaumkern**

143 Seiten, 60 Bilder, 16 Tabellen. 2005.  
ISBN 3-87525-228-4

Band 165: Gordana Michos

**Mechatronische Ansätze zur Optimierung von Vorschubachsen**

146 Seiten, 87 Bilder, 17 Tabellen. 2005.  
ISBN 3-87525-230-6

Band 166: Markus Stark

**Auslegung und Fertigung hochpräziser Faser-Kollimator-Arrays**

158 Seiten, 115 Bilder, 11 Tabellen. 2005.  
ISBN 3-87525-231-4

Band 167: Yurong Zhou

**Kollaboratives Engineering Management in der integrierten virtuellen  
Entwicklung der Anlagen für die Elektronikproduktion**

156 Seiten, 84 Bilder, 6 Tabellen. 2005.  
ISBN 3-87525-232-2

Band 168: Werner Enser

**Neue Formen permanenter und lösbarer elektrischer  
Kontaktierungen für mechatronische Baugruppen**

190 Seiten, 112 Bilder, 5 Tabellen. 2005.  
ISBN 3-87525-233-0

Band 169: Katrin Melzer

**Integrierte Produktpolitik bei elektrischen und elektronischen  
Geräten zur Optimierung des Product-Life-Cycle**

155 Seiten, 91 Bilder, 17 Tabellen. 2005.  
ISBN 3-87525-234-9

Band 170: Alexander Putz

**Grundlegende Untersuchungen zur Erfassung der realen Vorspannung von  
armierten Kaltfließpresswerkzeugen mittels Ultraschall**

137 Seiten, 71 Bilder, 15 Tabellen. 2006.  
ISBN 3-87525-237-3

Band 171: Martin Prechtl  
**Automatisiertes Schichtverfahren für metallische Folien –  
System- und Prozesstechnik**  
154 Seiten, 45 Bilder, 7 Tabellen. 2006.  
ISBN 3-87525-238-1

Band 172: Markus Meidert  
**Beitrag zur deterministischen Lebensdauerabschätzung  
von Werkzeugen der Kaltmassivumformung**  
131 Seiten, 78 Bilder, 9 Tabellen. 2006.  
ISBN 3-87525-239-X

Band 173: Bernd Müller  
**Robuste, automatisierte Montagesysteme durch adaptive Prozessführung  
und montageübergreifende Fehlerprävention am Beispiel flächiger Leichtbauteile**  
147 Seiten, 77 Bilder, 0 Tabellen. 2006.  
ISBN 3-87525-240-3

Band 174: Alexander Hofmann  
**Hybrides Laserdurchstrahlenschweißen von Kunststoffen**  
136 Seiten, 72 Bilder, 4 Tabellen. 2006.  
ISBN 978-3-87525-243-9  
ISBN 3-87525-243-8

Band 175: Peter Wölflick  
**Innovative Substrate und Prozesse mit feinsten Strukturen  
für bleifreie Mechatronik-Anwendungen**  
177 Seiten, 148 Bilder, 24 Tabellen. 2006.  
ISBN 978-3-87525-246-0  
ISBN 3-87525-246-2

Band 176: Attila Kornlődi  
**Detection and Prevention of Hot Cracks during Laser Welding  
of Aluminium Alloys Using Advanced Simulation Methods**  
155 Seiten, 89 Bilder, 14 Tabellen. 2006.  
ISBN 978-3-87525-248-4  
ISBN 3-87525-248-9

Band 177: Uwe Popp  
**Grundlegende Untersuchungen zum Laserstrahlstrukturieren  
von Kaltmassivumformwerkzeugen**  
140 Seiten, 67 Bilder, 16 Tabellen. 2006.  
ISBN 978-3-87525-249-1  
ISBN 3-87525-249-7

Band 178: Veit Rückel  
**Rechnergestützte Ablaufplanung und Bahngenerierung  
Für kooperierende Industrieroboter**  
148 Seiten, 75 Bilder, 7 Tabellen. 2006.  
ISBN 978-3-87525-250-7  
ISBN 3-87525-250-0

Band 179: Manfred Dirscherl  
**Nicht-thermische Mikrojustiertechnik mittels ultrakurzer Laserpulse**  
154 Seiten, 69 Bilder, 10 Tabellen. 2007.  
ISBN 978-3-87525-251-4  
ISBN 3-87525-251-9

Band 180: Yong Zhuo  
**Entwurf eines rechnergestützten integrierten Systems für Konstruktion und  
Fertigungsplanung räumlicher spritzgegossener Schaltungsträger (3D-MID)**  
181 Seiten, 95 Bilder, 5 Tabellen. 2007.  
ISBN 978-3-87525-253-8

Band 181: Stefan Lang  
**Durchgängige Mitarbeiterinformation zur Steigerung von Effizienz und  
Prozesssicherheit in der Produktion**  
172 Seiten, 93 Bilder. 2007.  
ISBN 978-3-87525-257-6

Band 182: Hans-Joachim Krauß  
**Laserstrahlinduzierte Pyrolyse präkeramischer Polymere**  
171 Seiten, 100 Bilder. 2007.  
ISBN 978-3-87525-258-3

Band 183: Stefan Junker  
**Technologien und Systemlösungen für die flexibel automatisierte Bestückung  
permanent erregter Läufer mit oberflächenmontierten Dauermagneten**  
173 Seiten, 75 Bilder. 2007.  
ISBN 978-3-87525-259-0

Band 184: Rainer Kohlbauer  
**Wissensbasierte Methoden für die simulationsgestützte Auslegung  
wirkmedienbasierter Blechumformprozesse**  
135 Seiten, 50 Bilder. 2007.  
ISBN 978-3-87525-260-6

Band 185: Klaus Lamprecht  
**Wirkmedienbasierte Umformung tiefgezogener Vorformen unter besonderer  
Berücksichtigung maßgeschneiderter Halbzeuge**  
137 Seiten, 81 Bilder. 2007.  
ISBN 978-3-87525-265-1

Band 186: Bernd Zolleiß  
**Optimierte Prozesse und Systeme für die Bestückung  
mechatronischer Baugruppen**  
180 Seiten, 117 Bilder. 2007.  
ISBN 978-3-87525-266-8

Band 187: Michael Kerausch  
**Simulationsgestützte Prozessauslegung für das Umformen lokal  
wärmebehandelter Aluminiumplatten**  
146 Seiten, 76 Bilder, 7 Tabellen 2007.  
ISBN 978-3-87525-267-5

Band 188: Matthias Weber  
**Unterstützung der Wandlungsfähigkeit von Produktionsanlagen  
durch innovative Softwaresysteme**  
183 Seiten, 122 Bilder, 3 Tabellen 2007.  
ISBN 978-3-87525-269-9

Band 189: Thomas Frick  
**Untersuchung der prozessbestimmenden Strahl-Stoff-Wechselwirkungen  
beim Laserstrahlschweißen von Kunststoffen**  
104 Seiten, 62 Bilder, 8 Tabellen 2007.  
ISBN 978-3-87525-268-2

Band 190: Joachim Hecht  
**Werkstoffcharakterisierung und Prozessauslegung für die wirkmedienbasierte  
Doppelblech-Umformung von Magnesiumlegierungen**  
107 Seiten, 91 Bilder, 2 Tabellen 2007.  
ISBN 978-3-87525-270-5

Band 191: Ralf Völkl  
**Stochastische Simulation zur Werkzeuglebensdaueroptimierung und  
Präzisionsfertigung in der Kaltmassivumformung**  
178 Seiten, 75 Bilder, 12 Tabellen 2008.  
ISBN 978-3-87525-272-9

Band 192: Massimo Tolazzi  
**Innenhochdruck-Umformen verstärkter Blech-Rahmenstrukturen**  
164 Seiten, 85 Bilder, 7 Tabellen 2008.  
ISBN 978-3-87525-273-6

Band 193: Cornelia Hoff  
**Untersuchung der Prozesseinflussgrößen beim  
Presshärten des höchstfesten Vergütungsstahls 22MnB5**  
133 Seiten, 92 Bilder, 5 Tabellen 2008.  
ISBN 978-3-87525-275-0

Band 194: Christian Alvarez  
**Simulationsgestützte Methoden zur effizienten Gestaltung von  
Lötprozessen in der Elektronikproduktion**  
149 Seiten, 86 Bilder, 8 Tabellen 2008.  
ISBN 978-3-87525-277-4

Band 195: Andreas Kunze  
**Automatisierte Montage von makromechatronischen  
Modulen zur flexiblen Integration in hybride Pkw-Bordnetze**  
160 Seiten, 90 Bilder, 14 Tabellen 2008.  
ISBN 978-3-87525-278-1

Band 196: Wolfgang Hußnätter  
**Grundlegende Untersuchungen zur experimentellen Ermittlung und zur Modellierung von Fließortkurven bei erhöhten Temperaturen**  
152 Seiten, 73 Bilder, 21 Tabellen 2008.  
ISBN 978-3-87525-279-8

Band 197: Thomas Bigl  
**Entwicklung, angepasste Herstellungsverfahren und erweiterte Qualitätssicherung von einsatzgerechten elektronischen Baugruppen**  
175 Seiten, 107 Bilder, 14 Tabellen 2008.  
ISBN 978-3-87525-280-4

Band 198: Stephan Roth  
**Grundlegende Untersuchungen zum Excimerlaserstrahl-Abtragen unter Flüssigkeitsfilmen**  
113 Seiten, 47 Bilder, 14 Tabellen 2008.  
ISBN 978-3-87525-281-1

Band 199: Artur Giera  
**Prozesstechnische Untersuchungen zum Rührreißschweißen metallischer Werkstoffe**  
179 Seiten, 104 Bilder, 36 Tabellen 2008.  
ISBN 978-3-87525-282-8

Band 200: Jürgen Lechler  
**Beschreibung und Modellierung des Werkstoffverhaltens von presshärtbaren Bor-Manganstählen**  
154 Seiten, 75 Bilder, 12 Tabellen 2009.  
ISBN 978-3-87525-286-6

Band 201: Andreas Blankl  
**Untersuchungen zur Erhöhung der Prozessrobustheit bei der Innenhochdruck-Umformung von flächigen Halbzeugen mit vor- bzw. nachgeschalteten Laserstrahlfügeoperationen**  
120 Seiten, 68 Bilder, 9 Tabellen 2009.  
ISBN 978-3-87525-287-3

Band 202: Andreas Schaller  
**Modellierung eines nachfrageorientierten Produktionskonzeptes für mobile Telekommunikationsgeräte**  
120 Seiten, 79 Bilder, 0 Tabellen 2009.  
ISBN 978-3-87525-289-7

Band 203: Claudius Schimpf  
**Optimierung von Zuverlässigkeitsuntersuchungen, Prüfabläufen und Nacharbeitsprozessen in der Elektronikproduktion**  
162 Seiten, 90 Bilder, 14 Tabellen 2009.  
ISBN 978-3-87525-290-3

Band 204: Simon Dietrich  
**Sensoren zur Schwerpunktslagebestimmung der optischen Prozessemissionen beim Laserstrahltiefschweißen**  
138 Seiten, 70 Bilder, 5 Tabellen 2009.  
ISBN 978-3-87525-292-7

Band 205: Wolfgang Wolf  
**Entwicklung eines agentenbasierten Steuerungssystems zur Materialflussorganisation im wandelbaren Produktionsumfeld**  
167 Seiten, 98 Bilder, 2009.  
ISBN 978-3-87525-293-4

Band 206: Steffen Polster  
**Laserdurchstrahlschweißen transparenter Polymerbauteile**  
160 Seiten, 92 Bilder, 13 Tabellen 2009.  
ISBN 978-3-87525-294-1

Band 207: Stephan Manuel Dörfler  
**Rührreibschweißen von walzplattiertem Halbzeug und Aluminiumblech zur Herstellung flächiger Aluminiumschaum-Sandwich-Verbundstrukturen**  
190 Seiten, 98 Bilder, 5 Tabellen 2009.  
ISBN 978-3-87525-295-8

Band 208: Uwe Vogt  
**Seriennahe Auslegung von Aluminium Tailored Heat Treated Blanks**  
151 Seiten, 68 Bilder, 26 Tabellen 2009.  
ISBN 978-3-87525-296-5

Band 209: Till Laumann  
**Qualitative und quantitative Bewertung der Crashtaughigkeit von höchstfesten Stählen**  
117 Seiten, 69 Bilder, 7 Tabellen 2009.  
ISBN 978-3-87525-299-6

Band 210: Alexander Diehl  
**Größeneffekte bei Biegeprozessen- Entwicklung einer Methodik zur Identifikation und Quantifizierung**  
180 Seiten, 92 Bilder, 12 Tabellen 2010.  
ISBN 978-3-87525-302-3

Band 211: Detlev Staud  
**Effiziente Prozesskettenauslegung für das Umformen lokal wärmebehandelter und geschweißter Aluminiumbleche**  
164 Seiten, 72 Bilder, 12 Tabellen 2010.  
ISBN 978-3-87525-303-0

Band 212: Jens Ackermann  
**Prozesssicherung beim Laserdurchstrahlenschweißen thermoplastischer Kunststoffe**  
129 Seiten, 74 Bilder, 13 Tabellen 2010.  
ISBN 978-3-87525-305-4

Band 213: Stephan Weidel  
**Grundlegende Untersuchungen zum Kontaktzustand zwischen Werkstück und Werkzeug bei umformtechnischen Prozessen unter tribologischen Gesichtspunkten**  
144 Seiten, 67 Bilder, 11 Tabellen 2010.  
ISBN 978-3-87525-307-8

Band 214: Stefan Geißdörfer  
**Entwicklung eines mesoskopischen Modells zur Abbildung von Größeneffekten in der Kaltmassivumformung mit Methoden der FE-Simulation**  
133 Seiten, 83 Bilder, 11 Tabellen 2010.  
ISBN 978-3-87525-308-5

Band 215: Christian Matzner  
**Konzeption produktspezifischer Lösungen zur Robustheitssteigerung elektronischer Systeme gegen die Einwirkung von Betauung im Automobil**  
165 Seiten, 93 Bilder, 14 Tabellen 2010.  
ISBN 978-3-87525-309-2

Band 216: Florian Schüßler  
**Verbindungs- und Systemtechnik für thermisch hochbeanspruchte und miniaturisierte elektronische Baugruppen**  
184 Seiten, 83 Bilder, 18 Tabellen 2010.  
ISBN 978-3-87525-310-8

Band 217: Massimo Cojutti  
**Strategien zur Erweiterung der Prozessgrenzen bei der Innhochdruck-Umformung von Rohren und Blechpaaren**  
125 Seiten, 56 Bilder, 9 Tabellen 2010.  
ISBN 978-3-87525-312-2

Band 218: Raoul Plettke  
**Mehrkriterielle Optimierung komplexer Aktorsysteme für das Laserstrahljustieren**  
152 Seiten, 25 Bilder, 3 Tabellen 2010.  
ISBN 978-3-87525-315-3

Band 219: Andreas Dobroschke  
**Flexible Automatisierungslösungen für die Fertigung  
wickeltechnischer Produkte**  
184 Seiten, 109 Bilder, 18 Tabellen 2011.  
ISBN 978-3-87525-317-7

Band 220: Azhar Zam  
**Optical Tissue Differentiation for Sensor-Controlled  
Tissue-Specific Laser Surgery**  
99 Seiten, 45 Bilder, 8 Tabellen 2011.  
ISBN 978-3-87525-318-4

Band 221: Michael Rösch  
**Potenziale und Strategien zur Optimierung des  
Schablonendruckprozesses in der Elektronikproduktion**  
192 Seiten, 127 Bilder, 19 Tabellen 2011.  
ISBN 978-3-87525-319-1

Band 222: Thomas Rechtenwald  
**Quasi-isothermes Laserstrahlsintern von Hochtemperatur-Thermoplasten -  
Eine Betrachtung werkstoff- prozessspezifischer Aspekte am Beispiel PEEK**  
150 Seiten, 62 Bilder, 8 Tabellen 2011.  
ISBN 978-3-87525-320-7

Band 223: Daniel Craiovan  
**Prozesse und Systemlösungen für die SMT-Montage optischer Bauelemente  
auf Substrate mit integrierten Lichtwellenleitern**  
165 Seiten, 85 Bilder, 8 Tabellen 2011.  
ISBN 978-3-87525-324-5

Band 224: Kay Wagner  
**Beanspruchungsangepasste Kaltmassivumformwerkzeuge durch lokal optimierte  
Werkzeugoberflächen**  
147 Seiten, 103 Bilder, 17 Tabellen 2011.  
ISBN 978-3-87525-325-2

Band 225: Martin Brandhuber  
**Verbesserung der Prognosegüte des Versagens von Punktschweißverbindungen bei  
höchstfesten Stahlgüten**  
155 Seiten, 91 Bilder, 19 Tabellen 2011.  
ISBN 978-3-87525-327-6

Band 226: Peter Sebastian Feuser  
**Ein Ansatz zur Herstellung von pressgehärteten Karosseriekomponenten mit maßgeschneiderten  
mechanischen Eigenschaften: Temperierte Umformwerkzeuge. Prozessfenster, Prozesssimulation  
und funktionale Untersuchung**  
195 Seiten, 97 Bilder, 60 Tabellen 2012.  
ISBN 978-3-87525-328-3

Band 227: Murat Arbak  
**Material Adapted Design of Cold Forging Tools Exemplified  
by Powder Metallurgical Tool Steels and Ceramics**  
109 Seiten, 56 Bilder, 8 Tabellen 2012.  
ISBN 978-3-87525-330-6

Band 228: Indra Pitz  
**Beschleunigte Simulation des Laserstrahlumformens von Aluminiumblechen**  
137 Seiten, 45 Bilder, 27 Tabellen 2012.  
ISBN 978-3-87525-333-7

Band 229: Alexander Grimm  
**Prozessanalyse und -überwachung des Laserstrahlhartlötens mittels optischer Sensorik**  
125 Seiten, 61 Bilder, 5 Tabellen 2012.  
ISBN 978-3-87525-334-4

Band 230: Markus Kaupper  
**Biegen von höhenfesten Stahlblechwerkstoffen - Umformverhalten und Grenzen der Biegebarkeit**  
160 Seiten, 57 Bilder, 10 Tabellen 2012.  
ISBN 978-3-87525-339-9



Band 231: Thomas Kroiß

**Modellbasierte Prozessauslegung für die Kaltmassivumformung unter Berücksichtigung der Werkzeug- und Pressenauffederung**

169 Seiten, 50 Bilder, 19 Tabellen 2012.  
ISBN 978-3-87525-341-2

Band 232: Christian Goth

**Analyse und Optimierung der Entwicklung und Zuverlässigkeit räumlicher Schaltungsträger (3D-MID)**

176 Seiten, 102 Bilder, 22 Tabellen 2012.  
ISBN 978-3-87525-340-5

Band 233: Christian Ziegler

**Ganzheitliche Automatisierung mechatronischer Systeme in der Medizin am Beispiel Strahlentherapie**

170 Seiten, 71 Bilder, 19 Tabellen 2012.  
ISBN 978-3-87525-342-9

Band 234: Florian Albert

**Automatisiertes Laserstrahllöten und -reparaturlöten elektronischer Baugruppen**

127 Seiten, 78 Bilder, 11 Tabellen 2012.  
ISBN 978-3-87525-344-3

Band 235: Thomas Stöhr

**Analyse und Beschreibung des mechanischen Werkstoffverhaltens von presshärtbaren Bor-Manganstählen**

118 Seiten, 74 Bilder, 18 Tabellen 2013.  
ISBN 978-3-87525-346-7

Band 236: Christian Kägeler

**Prozessdynamik beim Laserstrahlschweißen verzinkter Stahlbleche im Überlappspstoß**

145 Seiten, 80 Bilder, 3 Tabellen 2013.  
ISBN 978-3-87525-347-4

Band 237: Andreas Sulzberger

**Seriennahe Auslegung der Prozesskette zur wärmeunterstützten Umformung von Aluminiumblechwerkstoffen**

153 Seiten, 87 Bilder, 17 Tabellen 2013.  
ISBN 978-3-87525-349-8

Band 238: Simon Opel

**Herstellung prozessangepasster Halbzeuge mit variabler Blechdicke durch die Anwendung von Verfahren der Blech-massivumformung**

163 Seiten, 108 Bilder, 207 Tabellen 2013.  
ISBN 978-3-87525-350-4

Band 239: Rajesh Kanawade

**In-vivo Monitoring of Epithelium Vessel and Capillary Density for the Application of Detection of Clinical Shock and Early Signs of Cancer Development**

124 Seiten, 58 Bilder, 15 Tabellen 2013.  
ISBN 978-3-87525-351-1

Band 240: Stephan Busse

**Entwicklung und Qualifizierung eines Schneidlinchverfahrens**

119 Seiten, 86 Bilder, 20 Tabellen 2013.  
ISBN 978-3-87525-352-8

Band 241: Karl-Heinz Leitz

**Mikro- und Nanostrukturierung mit kurz und ultrakurz gepulster Laserstrahlung**

154 Seiten, 71 Bilder, 9 Tabellen 2013.  
ISBN 978-3-87525-355-9

Band 242: Markus Michl

**Webbasierte Ansätze zur ganzheitlichen technischen Diagnose**

182 Seiten, 62 Bilder, 20 Tabellen 2013.  
ISBN 978-3-87525-356-6

Band 243: Vera Sturm

**Einfluss von Chargenschwankungen auf die Verarbeitungsgrenzen von Stahlwerkstoffen**

113 Seiten, 58 Bilder, 9 Tabellen.  
ISBN 978-3-87525-357-3

Band 244: Christian Neudel

**Mikrostrukturelle und mechanisch-technologische Eigenschaften widerstandspunktschweißter Aluminium-Stahl-Verbindungen für den Fahrzeugbau**

178 Seiten, 171 Bilder, 31 Tabellen.  
ISBN 978-3-87525-358-0

Band 245: Anja Neumann

**Konzept zur Beherrschung der Prozessschwankungen im Presswerk**

162 Seiten, 68 Bilder, 15 Tabellen.

ISBN 978-3-87525-360-3

Band 246: Ulf-Hermann Quantin

**Laserbasierte Nanostrukturierung mit optisch positionierten Mikrolinsen**

137 Seiten, 89 Bilder, 6 Tabellen.

ISBN 978-3-87525-361-0

Band 247: Erik Lamprecht

**Der Einfluss der Fertigungsverfahren auf die Wirbelstromverluste von Stator-Einzelzahnblechpaketen für den Einsatz in Hybrid- und Elektrofahrzeugen**

148 Seiten, 138 Bilder, 4 Tabellen.

ISBN 978-3-87525-362-7

Band 248: Sebastian Rösel

**Wirkmedienbasierte Umformung von Blechhalbzeugen unter Anwendung magnetorheologischer Flüssigkeiten als kombiniertes Wirk- und Dichtmedium**

148 Seiten, 61 Bilder, 12 Tabellen.

ISBN 978-3-87525-363-4

Band 249: Paul Hippchen

**Simulative Prognose der Geometrie indirekt pressgehalteter Karosseriebauteile für die industrielle Anwendung**

163 Seiten, 89 Bilder, 12 Tabellen.

ISBN 978-3-87525-364-1

Band 250: Martin Zubeil

**Versagensprognose bei der Prozesssimulation von Biegeumform- und Falzverfahren**

171 Seiten, 90 Bilder, 5 Tabellen.

ISBN 978-3-87525-365-8

Band 251: Alexander Kühl

**Flexible Automatisierung der Statorenmontage mit Hilfe einer universellen ambidexteren Kinematik**

142 Seiten, 60 Bilder, 26 Tabellen 2014.

ISBN 978-3-87525-367-2

Band 252: Thomas Albrecht

**Optimierte Fertigungstechnologien für Rotoren getriebeintegrierter PM-Synchronmotoren von Hybridfahrzeugen**

198 Seiten, 130 Bilder, 38 Tabellen 2014.

ISBN 978-3-87525-368-9

Band 253: Florian Risch

**Planning and Production Concepts for Contactless Power Transfer Systems for Electric Vehicles**

185 Seiten, 125 Bilder, 13 Tabellen 2014.

ISBN 978-3-87525-369-6

Band 254: Markus Weigl

**Laserstrahlschweißen von Mischverbindungen aus austenitischen und ferritischen korrosionsbeständigen Stahlwerkstoffen**

184 Seiten, 110 Bilder, 6 Tabellen 2014.

ISBN 978-3-87525-370-2

Band 255: Johannes Noneder

**Beanspruchungserfassung für die Validierung von FE-Modellen zur Auslegung von Massivumformwerkzeugen**

161 Seiten, 65 Bilder, 14 Tabellen 2014.

ISBN 978-3-87525-371-9

Band 256: Andreas Reinhardt

**Ressourceneffiziente Prozess- und Produktionstechnologie für flexible Schaltungsträger**

123 Seiten, 69 Bilder, 19 Tabellen 2014.

ISBN 978-3-87525-373-3

Band 257: Tobias Schmuck

**Ein Beitrag zur effizienten Gestaltung globaler Produktions- und Logistiknetzwerke mittels Simulation**

151 Seiten, 74 Bilder 2014.

ISBN 978-3-87525-374-0

Band 258: Bernd Eichenhüller

**Untersuchungen der Effekte und Wechselwirkungen charakteristischer Einflussgrößen auf das Umformverhalten bei Mikroumformprozessen**

127 Seiten, 29 Bilder, 9 Tabellen 2014.

ISBN 978-3-87525-375-7

Band 259: Felix Lütke

**Vielseitiges autonomes Transportsystem basierend auf Weltmodellerstellung mittels Datenfusion von Deckenkameras und Fahrzeugsensoren**

152 Seiten, 54 Bilder, 20 Tabellen 2014.

ISBN 978-3-87525-376-4