

---

# **Simulationsgestützte Landmarkendetektion, Lokalisierung und Modellgenerierung für mobile Systeme**

---

Von der Fakultät für Elektrotechnik und Informationstechnik der  
Rheinisch-Westfälischen Technischen Hochschule Aachen  
zur Erlangung des akademischen Grades eines Doktors der  
Ingenieurwissenschaften genehmigte Dissertation

vorgelegt von

Diplom-Informatiker Björn SONDERMANN  
aus El Paso, Texas, U.S.A.

*Berichter:* Universitätsprofessor Dr.-Ing. Jürgen ROSSMANN  
Universitätsprofessor Dr. rer. nat. Dominik HENRICH

Tag der mündlichen Prüfung: 4. Mai 2018

**Diese Dissertation ist auf den Internetseiten der  
Hochschulbibliothek online verfügbar.**



# Zusammenfassung

Grundlegend für eine sinnvolle Interaktion von mobilen Systemen mit ihrer Umgebung ist eine sowohl geometrische als auch semantische Umgebungserfassung und -beschreibung. Kern der vorliegenden Dissertation ist ein Konzept zur modellbasierten Umgebungserfassung, gepaart mit einer objektbasierten Lokalisierung. Ziel ist es, die Umgebung in semantischen Umweltmodellen zu beschreiben und in Verbindung mit einem Simulationssystem schließlich als anwendungsübergreifendes Semantisches Weltmodell (ASW) zu repräsentieren. Das Konzept der vorliegenden Dissertation beinhaltet die drei Hauptkomponenten Umgebungserfassung, Lokalisierung und Umweltmodellierung. Die Erkennung umgebender Objekte und die Zuordnung einer geeigneten Semantik wird in der vorliegenden Dissertation anhand der Entwicklung und Umsetzung sensordatenverarbeitender Algorithmen für verschiedene Anwendungsszenarien vorgestellt. Die realisierte landmarkenbasierte Lokalisierung ermöglicht es, die erkannten Objekte in einem gemeinsamen Bezugssystem zu verorten, um anschließend darin unterschiedliche Datenquellen miteinander verknüpfen zu können. Das breite Spektrum an Anwendungen beginnt mit einfachen Szenarien der mobilen Robotik und reicht bis hin zur Beteiligung vollautonomer Automobile am Straßenverkehr. Mit steigender Komplexität des Anwendungsgebiets steigen auch die Anforderungen an die Umgebungserfassung und -beschreibung. Zudem findet eine Fokussierung auf das Konzept der Situationserfassung statt, begonnen bei der Betrachtung von Daten einzelner Sensoren, über die semantische Interpretation der Messungen, bis hin zur anwendungsübergreifenden, gesamtheitlichen Modellerzeugung im Simulationssystem. Die vorliegende Arbeit erweitert die momentan vorherrschende Vorstellung einer in erster Linie sensorischen Umgebungserfassung und -modellierung um die wesentlichen Aspekte der semantischen Erkennung, Beschreibung und Verwaltung. Durch die semantische Modellierung - ähnlich der menschlichen Umgebungswahrnehmung - können komplexe Anwendungen realisiert werden, die ihre Umgebung „verstehen“ und so in der Lage sind, auch komplexe Zusammenhänge zu erfassen, zu klassifizieren und ggfs. auch zu abstrahieren - auf jeden Fall aber sinnvoll und sicher zu interagieren. Durch die Einführung eines formalen und modularen Konzepts, das die drei zentralen Aufgabenfelder miteinander sinnvoll verknüpft, wird eine einheitliche und übergreifende Lösung für die verschiedenen Anwendungsklassen geschaffen. Die vorliegende Arbeit legt den Grundstein für weitere Entwicklungen auf dem Gebiet der kollektiv nutzbaren, anwendungsübergreifenden Weltmodelle und die realisierten Anwendungen zeigen das beeindruckende Potential dieser Forschung.



## *Abstract*

For the meaningful interaction of mobile systems with their environment, a geometric and semantic environment detection and description is fundamental. The quintessence of this dissertation is a concept for model-based environment detection, coupled with object-based localization. The goal is to describe the environment in semantic environmental models and, in conjunction with a simulation system, to represent it as a comprehensive semantic world model. The concept of the present dissertation includes the three main components environment detection, localization and environmental modeling. The recognition of surrounding objects and the assignment of suitable semantics is presented on the basis of the development and implementation of sensor data processing algorithms for different application scenarios. The implemented landmark-based localization makes it possible to locate the detected objects in a common frame of reference so that they can subsequently link additional data sources. The broad spectrum of applications starts with simple scenarios of mobile robotics and extends to the participation of fully autonomous automobiles on the road. With increasing complexity of the field of application, the requirements for the environmental detection and description also increase. In addition, a focus on the concept of situational awareness takes place, starting with the observation of data from individual sensors, through the semantic interpretation of the measurements, up to application-spanning, holistic modeling in the simulation system. The present work extends the currently predominant notion of primarily sensory environment detection and modeling to the essential aspects of semantic recognition, description and management. Semantic modeling - similar to human perception of the environment - enables the implementation of complex applications that "understand" their environment. Thus, they need to be able to capture, classify and, in some cases, to abstract complex relationships. Eventually, they are able to interact safely and reasonably. The introduction of a formal and modular concept, which interconnects the three domains, creates a uniform and comprehensive solution for various applications. The present work lays the foundation for further developments in the field of collectively usable, multi-application world models and the realized applications show the impressive potential of this research.



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
<b>2</b>	<b>Grundlagen und Stand der Technik</b>	<b>7</b>
2.1	Sensorik . . . . .	7
2.1.1	Propriozeptive Sensoren . . . . .	8
2.1.2	Entfernungsmesssensorik . . . . .	12
2.1.3	Optische Abbildungssensorik . . . . .	21
2.1.4	Gegenüberstellung der verschiedenen Sensoren . . . . .	26
2.2	Umgebungserfassung . . . . .	29
2.2.1	Einzelaufnahmen von Sensoren . . . . .	29
2.2.2	Spatial Mapping . . . . .	30
2.2.3	Features . . . . .	31
2.2.4	Hinderniserkennung . . . . .	35
2.2.5	Occupancy Grids . . . . .	36
2.2.6	Klassifiziertes Spatial Mapping . . . . .	37
2.2.7	Vom Spatial Mapping zum semantischen Umgebungsmodell . . . . .	40
2.3	Lokalisierungsverfahren . . . . .	42
2.3.1	Kartenlose Lokalisierung . . . . .	42
2.3.2	Kartenbasierte Lokalisierung . . . . .	46
<b>3</b>	<b>Konzept</b>	<b>55</b>
3.1	Umgebungserfassung, Lokalisierung und Umweltmodellierung . . . . .	56
3.2	Anwendungsübergreifende Semantische Weltmodelle . . . . .	58
3.3	Die Simulationsplattform VEROSIM™ . . . . .	64
3.3.1	Virtuelle Testbeds . . . . .	65
3.3.2	Simulationsgestützte Entwicklung . . . . .	67
3.3.3	Simulationsgestützte Steuerung . . . . .	68
3.4	Die mobile Roboterplattform SeekurJr™ . . . . .	69

3.5	Die Umgebungserfassungsplattform „SELMA“ . . . . .	71
3.6	Das VEROSIM™-Visual4-Konzept . . . . .	73
<b>4</b>	<b>Semantische Umgebungserfassung</b>	<b>77</b>
4.1	Anforderungen der Lokalisierung . . . . .	78
4.2	Detektion und Erkennung von Objekten . . . . .	79
4.2.1	Allgemeine Strukturmerkmale . . . . .	80
4.2.2	Erkennung von Bäumen . . . . .	85
4.2.3	Erkennung von Felsen und Steinen . . . . .	91
4.2.4	Erkennung von künstlichen Markern . . . . .	104
4.3	Semantische Beschreibung durch Attribuierung . . . . .	109
4.3.1	Attribuierung von Baumlandmarken . . . . .	109
4.3.2	Attribuierung von Felsen und Steinen . . . . .	117
4.3.3	Attribuierung von künstlichen Markern . . . . .	118
<b>5</b>	<b>Landmarkenbasierte Lokalisierung</b>	<b>121</b>
5.1	Sensorfusion . . . . .	126
5.2	Allgemeines Landmarkentracking . . . . .	131
5.3	Markerbasierte Lokalisierung . . . . .	135
5.3.1	Posenschätzung durch Rückprojektion bekannter Geometrien . . . . .	136
5.3.2	Ergebnisse der markerbasierten Lokalisierung . . . . .	138
5.4	Monte Carlo Lokalisierung (MCL) . . . . .	140
5.4.1	Partikelfilter . . . . .	140
5.4.2	Kalmanfilter . . . . .	141
5.4.3	Ergebnisse der MCL mittels Partikelfilter . . . . .	142
<b>6</b>	<b>Anwendungsübergreifende Umweltmodellierung</b>	<b>151</b>
6.1	Simultane Lokalisierung und Kartenerzeugung (SLAM) . . . . .	151
6.2	Kartenerzeugung und Aktualisierung . . . . .	155
6.3	Ergebnisse der Kartenerstellung . . . . .	156
6.3.1	Wald . . . . .	157
6.3.2	Indoor-Lokalisierung . . . . .	167
6.4	Umsetzung . . . . .	170
<b>7</b>	<b>Anwendungen</b>	<b>173</b>
7.1	Forsteinrichtung und Forstbewirtschaftung . . . . .	173
7.1.1	Automatisierte Waldinventur . . . . .	173
7.1.2	Anwendung in der technischen Produktion . . . . .	178

7.2	Explorationsmissionen . . . . .	181
7.2.1	Anwendung in extraterrestrischen Explorationsmissionen . . . . .	181
7.3	Automatisierte Gebäudevermessung . . . . .	185
7.3.1	Anwendung im Fabrik- und Anlagenbau . . . . .	186
7.3.2	Anwendung im Gebäudemanagement . . . . .	189
7.3.3	Indoorlokalisierung und Navigation . . . . .	190
7.4	Automotive-Anwendungen . . . . .	195
7.4.1	Lokalisierung in Tiefgaragen und Parkhäusern . . . . .	196
7.4.2	Verbesserte Fahrzeuglokalisierung . . . . .	200
<b>8</b>	<b>Zusammenfassung und Ausblick</b>	<b>207</b>



# Abbildungsverzeichnis

1.1	Konzept zur semantischen Umgebungserfassung, Lokalisierung und Umweltmodellierung . . . . .	3
2.1	Messprinzip eines Odometers mit einem bzw. zwei Messrädern . . . . .	11
2.2	Entfernungsmessung mittel Phasenmodulation . . . . .	13
2.3	3D-Laserscanner und aufgenommener Punktwolke . . . . .	15
2.4	Abhängigkeit zwischen Disparität und Entfernung . . . . .	18
2.5	Aus Doppler-Effekt resultierende Frequenzverschiebung beim CW-Radar . . . . .	20
2.6	Die Abbildung eines Weltpunkts in die Bildebene eines Lochkameramodells . . . . .	22
2.7	Skizze der Parameter zur Aufspannung des Bildraums . . . . .	24
2.8	Übersicht verschiedener Methoden zur Umgebungserfassung und Umweltmodellierung . . . . .	29
2.9	Entwicklung der Megapixelanzahl von digitalen Fotoapparaten . . . . .	32
2.10	Basis-Idee des Harris-Detektors . . . . .	33
2.11	Kategorisierung von räumlichen Umgebungsmodellen . . . . .	40
2.12	Übersicht etablierte Lokalisierungsverfahren . . . . .	43
2.13	Kontinuierlich wiederholende Schritte eines Kalman Filters . . . . .	51
2.14	Illustration des Kalman-Filters . . . . .	52
3.1	Grundlegendes Konzept dieser Arbeit . . . . .	57
3.2	Modelldefinition nach VDI . . . . .	59
3.3	Erzeugung anwendungsübergreifender Modelle . . . . .	60
3.4	Definition und Abgrenzung der Begriffe Feature, Landmarke und semantische Landmarke . . . . .	62
3.5	Erweiterung semantischer Objekte um weitere Attribute aus externen Datenquellen . . . . .	63
3.6	Das Grundkonzept des Simulationssystems VEROSIM . . . . .	64

3.7	Virtuelles Testbed . . . . .	66
3.8	Experimentierbare Digitale Zwillinge und ihre realen Gegenstücke . . . . .	67
3.9	Simulationsgestützte Entwicklung reduziert die Entwicklungszeit . . . . .	68
3.10	Die mobile Roboterplattform Seekur Jr in verschiedenen Testumgebungen . . . . .	69
3.11	Simulationsbasierte Steuerung und Evaluierung mit dem Seekur Jr . . . . .	70
3.12	Komponenten des ersten SELMA-Prototypen . . . . .	71
3.13	SELMA-Einheit in der ersten prototypischen Version . . . . .	72
3.14	SELMA-Einheit in der zweiten, miniaturisierten Version . . . . .	73
3.15	VEROSIM-Visual-4 Konzept . . . . .	74
3.16	Simulationsgestützte Prädiktion auf Basis eines Mentalen Modells . . . . .	75
4.1	Scanprofil eines 2D-Laserscanners . . . . .	81
4.2	LaserstrahlenDeltaSkizze . . . . .	82
4.3	Gegenüberstellung von Abstands- und Geometrieansatz . . . . .	83
4.4	Gegenüberstellung von Strahl- bzw. Pixelabstand zwischen 2D-Laserscanner und Stereokamera . . . . .	84
4.5	Analyse der Tiefenberechnung und Tiefengenauigkeit bei Stereokameras . . . . .	86
4.6	RGB- und Tiefenbild einer Stereoaufnahme im Wald . . . . .	89
4.7	Mars-Oberflaeche-mit-Steinen . . . . .	91
4.8	Hinderniserkennung nach Manduchi . . . . .	92
4.9	Tiefenkarte versus Höhenkarte . . . . .	93
4.10	Wichtigkeit einer korrekten Bodenebenenschätzung . . . . .	95
4.11	Ergebnisse der k-Means-Segmentierung und der anschließender Sortierung . . . . .	97
4.12	Simulierte extraterrestrische, planetare Explorationsmission im Virtuellen Testbed . . . . .	100
4.13	Ergebnisse der Bodenschätzung und Höhenkartengenerierung im Testkrater des DFKI Bremen . . . . .	101
4.14	Steindetektion im virtuellen Testbed zur extraterrestrischen Planetenexploration . . . . .	102
4.15	Steindetektion im Steinbruch . . . . .	103
4.16	Projektion eines AR-Markers in den Bildraum der Kamera . . . . .	105

4.17	Verarbeitungsprozess zur AR-Markerdetektion . . . . .	107
4.18	Detektierbarkeit von AR-Markern . . . . .	108
4.19	Analyse einer Durchmesserbestimmung mittels 2D-Laser-scanner . . . . .	110
4.20	Gegenüberstellung der vorgestellten Algorithmen zur Bestimmung von Baumdurchmessern . . . . .	112
4.21	Bestimmung des idealen Werts an Sichtungen bzgl. Ground Truth . . . . .	113
4.22	Absolute Genauigkeitsabweichung der Durchmesserbestimmungen im Testgebiet Zollikofen (CH) . . . . .	114
4.23	Das normalisierte Histogramm sowie die angepasste Wahrscheinlichkeitsdichtefunktion zeigen die annähernde Normalverteilung des Fehlers über alle BHD-Messungen. . .	115
4.24	Absoluter Fehler der BHD-Bestimmung auf Basis von Stereokameradaten . . . . .	116
5.1	Gegenüberstellung zwischen sensorabhängiger und sensorunabhängiger Lokalisierung . . . . .	122
5.2	Anwendungsunabhängiges Lokalisierungskonzept . . .	124
5.3	Adaption des allgemeinen Lokalisierungskonzepts auf zwei Anwendungen . . . . .	125
5.4	Gegenüberstellung der sensorabhängigen und sensorunabhängigen Lokalisierung im Multi-Sensor-Betrieb . . . .	127
5.5	Roboter als Bezugssystem der Umgebungserfassung . . .	128
5.6	3D-Variante eines AR-Markers . . . . .	130
5.7	3D-AR-Marker-Detektion mittel ToF-Kamera . . . . .	130
5.8	Problembeschreibung: Bewegungsrekonstruktion durch Landmarkentracking . . . . .	133
5.9	AR-Marker Posenschätzung: Translatorische Abweichung	138
5.10	AR-Marker Posenschätzung: Rotatorische Abweichung .	139
5.11	Durch Vermessungsingenieure werden die Lokalisierungsergebnisse validiert . . . . .	143
5.12	Trajektorie der MCL mit Partikelfilter im Wald . . . . .	144
5.13	Lokalisierungsgenauigkeit - Translation/Rotation . . . .	145
5.14	Relative Lokalisierungsgenauigkeit - Translation/Rotation	146
5.15	Positionsbestimmung der Roboterplattform anhand zweier Distanzmessungen . . . . .	148

5.16	Ergebnisse der Lokalisierung im Steinbruch anhand von Felsen als Landmarken . . . . .	149
6.1	Kombination aus Umgebungserfassung und Lokalisierungsverfahren . . . . .	153
6.2	Durchmesser-Verteilung eines Baumbestandes . . . . .	156
6.3	Testgebiete für die Anwendung im Wald . . . . .	158
6.4	Evaluation des SLAM-Ergebnisses in Lattenberg (NRW) . . . . .	159
6.5	Evaluation des SLAM-Ergebnisses in Traunstein (Bayern) . . . . .	161
6.6	Evaluation des SLAM-Ergebnisses in Zollikofen (Schweiz) . . . . .	162
6.7	Ergebnis des SLAM-Verfahrens in Zollikofen mit unterlegtem Orthofoto . . . . .	164
6.8	Umgebungsbedingungen für eine zuverlässige Bestandesaufnahme . . . . .	165
6.9	Statistischer Anteil der Waldflächen in Deutschland für die automatisierte Bestandese Erfassung . . . . .	166
6.10	Statistischer Anteil der Waldflächen in Deutschland für die halbautomatisierte Bestandese Erfassung . . . . .	167
6.11	Ergebnistrajektorie des AR-Marker-basierten SLAM . . . . .	168
6.12	Auswertung der rotatorischen/translatorischen Abweichung beim AR-Marker-basierten SLAM . . . . .	169
7.1	Hierarchische Flächeneinteilung und -beschreibung in der Forsteinrichtung . . . . .	176
7.2	Holzerntesimulaton auf Basis einer Einzelbaumbasierten Bestandesbeschreibung . . . . .	179
7.3	Harvester im semantisch erfassten Wald . . . . .	180
7.4	Curiosity-Rover des Mars Science Laboratory (MSL) real und simuliert . . . . .	182
7.5	Planetenexplorationsszenario in der 3D-Simulation . . . . .	183
7.6	Referenzmessungen für das reale Testbed für Planetenexplorationen im Steinbruch . . . . .	184
7.7	Reales Testbed für Planetenexploration in Steinbruch . . . . .	185
7.8	Anwendung im Fabrik- und Anlagenbau . . . . .	188
7.9	Beispiel für eine Augmented Reality Anwendung im Gebäudemanagement . . . . .	191
7.10	Servicerobotik in öffentlichen Gebäuden . . . . .	193
7.11	Das Uniklinikum der RWTH Aachen . . . . .	194
7.12	Modell einer Tiefgarage mit AR-Markern zur Lokalisierung . . . . .	196

7.13	Zustand zum Zeitpunkt während der Lokalisierung mittels AR-Marker . . . . .	197
7.14	Ergebnisstrajektorie der AR-Marker basierten Lokalisierung	198
7.15	Auswertung der translatorischen und rotatorischen Abweichungen bei der AR-Marker-basierten Lokalisierung . . .	199
7.16	Baumdetektion zur Verbesserung der Fahrzeuglokalisierung	202
7.17	Detektion von Bäumen und Straßenlaternen zur Verbesserung der Lokalisierung . . . . .	203
7.18	Bildbasierte Detektion und Entfernungsschätzung von Verkehrsschildern . . . . .	204



# Kapitel 1

## Einleitung

Allgemein versteht man unter einem Roboter einen programmierbaren Apparat, der durch einen frei programmierbaren Bewegungsablauf, für verschiedenste Aufgaben eingesetzt werden kann. Er soll den Menschen bei vielfältigen Aufgaben unterstützen [17]. Gerade in der Industrie sind Roboter schon nicht mehr wegzudenken und jährlich werden neue Rekorde bei deren Umsatzzahlen verzeichnet.<sup>1</sup> Aber nicht nur in industriellen Anwendungen können Roboter den Menschen bei seiner Arbeit unterstützen. Viele Arbeiten die schwere, körperliche Arbeit erfordern oder aufgrund gefährlicher Umgebung riskant für den Menschen sind, können von mobilen Robotern übernommen werden. Im Gegensatz zu Industrierobotern müssen die mobilen Systeme ihren Programmablauf selbstständig auf Veränderungen der Umwelt anpassen.<sup>2</sup>

Ob nun ein frei programmierbarer Bewegungsablauf oder eine selbstständige Anpassung des Programmablaufs an die Umwelt angestrebt wird, ein Roboter muss seine Umgebung wahrnehmen können, um entsprechend damit zu interagieren. Bei einfachen Industrierobotern in Fertigungsanlagen wird die Umwelt künstlich angepasst, sodass diese Wahrnehmung auf ein Minimum reduziert werden kann. Damit ein mobiler Roboter jedoch für alltägliche Aufgaben in einer größtenteils freien Umgebung eingesetzt werden kann, muss die Wahrnehmungsfähigkeit sowie die anschließende Interpretationsgabe entsprechend erhöht werden. Dazu sind komplexe Sensoren erforderlich, die genügend Umgebungsinformationen erfassen, um daraus das weitere Handeln ableiten zu können. Diese Art von Sensoren haben sich in den letzten Jahren mit

---

<sup>1</sup>Berichte werden z.B. regelmäßig von der Robotic Industries Association (RIA) veröffentlicht unter: <https://www.robotics.org/Industry-Statistics>.

<sup>2</sup>Die Japan Robot Association (JRA) <http://www.jara.jp/e/> bezeichnet diese Fähigkeit als wesentliches Merkmal für intelligente Roboter.

rasanter Geschwindigkeit weiter entwickelt und sind mittlerweile bereits in vielen alltäglichen Geräten (PCs, Mobiltelefone, Autos) verbaut.

Neben den Entwicklungen in der Sensorik wurden mit der Durchsetzung von Smartphones, Tablet-PCs und weiteren tragbaren Geräten auch die Geschwindigkeit und Miniaturisierung von Rechnern rasant fortentwickelt. Dadurch ist es seit Kurzem möglich, erschwingliche und leistungsfähige mobile Roboter zu entwickeln, die Aufgaben des Alltags effizient erledigen können wie Staubsaugen, Rasenmähen oder Fensterputzen.

Aber auch die Sensorik von Autos und die Rechenleistung ihrer Bordcomputer nimmt stetig zu. Neben den Sensoren zur Überwachung der internen Abläufe (Lambda-Sonde, Drehzahlsensor, etc.) und Komfortsensorik zur Messung von Helligkeit, Regen, Reifendruck, etc. werden vermehrt Sensoren für die Umgebungserfassung eingesetzt. Diese gibt es für den Nahbereich (z.B. Ultraschallsensoren) oder für den Fernbereich (LIDAR, Kameras, Radar).

Ebenfalls treibende Kraft in der Entwicklung neuer Methoden ist die Sicherung der Unversehrtheit von Menschen, Maschinen und Gütern in einzelnen und zwischen mehreren Produktionsstätten, denn hier bedeuten Schadensfälle sofort auch wirtschaftlichen Schaden. Daher werden hier vermehrt Methoden entwickelt, um ganze Produktionsprozesse sensorisch zu überwachen und abzusichern. Durch die kurze Reaktionszeiten von heutigen Rechnersystemen können hier auch Gefahren in Produktionsprozessen zuverlässig erkannt und abgewendet werden, wozu die menschliche Reaktionsfähigkeit nicht ausreichen würde. Durch die Übertragung dieser Methoden aus dem industriellen Bereich in alltägliche Anwendungsfälle, wie etwa das Steuern eines Fahrzeugs, profitieren nun auch private Bereiche von diesen Entwicklungen. So werden neue Autos mit Assistenzsystemen ausgestattet, die den Fahrweg überwachen oder Kollisionen durch automatisierte Notbremsmanöver abwenden und somit diese, vor einigen Jahrzehnten noch um einiges gefährlichere, Fortbewegungsart sicherer zu machen. Mit der aktuellen Entwicklung autonom fahrender Autos wird dieser Sicherheitsgedanke noch weiter fortgeführt und mit einem ebenfalls nicht zu vernachlässigenden Komfort kombiniert.

Alle diese Anwendungsfälle haben gemeinsam, dass die Umgebung eines mobilen Systems oder eines Objekts sensorisch erfasst und die aktuelle Lage dieses Systems oder Objekts bestimmt wird. Basierend auf

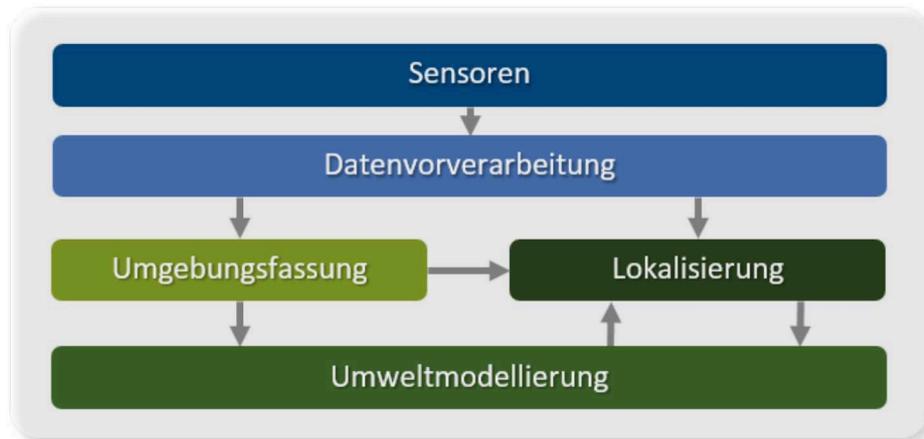


ABBILDUNG 1.1: Eine vereinfachte Darstellung des Grundlegenden Konzepts. Es nutzt die semantischen Objekte der Umgebungserfassung als Landmarken zur Lokalisierung und integriert im Anschluss die Posenschätzung mit den Objekten zu einem semantischen Weltmodell

diesen Daten können direkte Entscheidungen getroffen werden, die aktiv das Verhalten des Systems oder des Objekts unmittelbar beeinflussen oder ein semantisches Umgebungsmodell angefertigt werden, um die Daten zu einem späteren Zeitpunkt auszuwerten.

Zentrale Elemente für die Umsetzung dieser Anwendungsfälle sind die sensorische Umgebungserfassung und Lokalisierung, sowie die Erzeugung eines Weltmodells, das neben der Funktion als Navigationskarte weitere Analysen der Umgebung ermöglicht, z.B. eine anwendungsbezogene Attribuierung oder Beschreibung. Abbildung 1.1 skizziert grob dieses Konzept, das in Kapitel 3 näher ausgeführt wird.

Im Mittelpunkt der vorliegenden Arbeit stehen die drei Aufgabenfelder *Umgebungserfassung*, *Lokalisierung* und *Umweltmodellierung*, die unabhängig voneinander im Kontext verschiedenster Anwendungsfelder bereits vielfach untersucht wurden und auch aktuell noch Gegenstand zahlreicher wissenschaftlicher Publikationen sind. Mit der Betrachtung von sogenannten „Simultaneous Localization And Mapping“-Verfahren (SLAM), also der Lokalisierung mit simultaner Kartengenerierung, wurde bereits ein erster Schritt hin zur Nutzung der Synergieeffekte von zwei der drei Aufgabenfelder gemacht, nämlich der Lokalisierung und der Kartenerzeugung. Das große Potential der Verknüpfung aller drei Aufgabenfelder wurde bislang jedoch noch nicht hinreichend erkannt

und ausgeschöpft.

Der Begriff des *Semantischen Weltmodells* als eine gegenständliche Teilrepräsentation der Umwelt stellt hierbei den Kern aller weiteren Überlegungen und Entwicklungen dieser Arbeit dar. Im Gegensatz zu räumlichen Umweltmodellen, die das Ergebnis aktueller SLAM-Verfahren darstellen, augmentieren Semantische Weltmodelle die räumlichen Daten um einen kognitiven Prozess der Abstraktion, bei dem der sensorischen Wahrnehmung ein Begriff bzw. Konzept zugeordnet wird.

In der vorliegenden Arbeit wird das semantische Weltmodell als 4D-Simulationsmodell in einer Simulationsumgebung umgesetzt, um alle Vorteile einer simulationsgestützten Analyse darauf anwenden zu können. Die gegenständlichen Abbildungen real existierender Objekte im Modell werden in der Simulation als *Digitale Zwillinge* [18] dieser Objekte repräsentiert. Die Simulation erlaubt diesbezüglich eine intelligente Erfassung und Analyse von Zusammenhängen unter den jeweiligen Digitalen Zwillingen.

Für die Erzeugung eines simulierbaren Semantischen Weltmodells bedarf es einer diesbezüglich zielorientierten Betrachtung der drei Aufgabenfelder. Die im Rahmen dieser Arbeit entwickelten Methoden zur *Umgebungserfassung*, *Lokalisierung* und *Umweltmodellierung* wurden aus diesem Grund bereits simulationsgestützt entwickelt, d.h. die Umsetzung erfolgte in Form von Modulen (sog. „Plugins“) der in Kapitel 3.3 näher beschriebenen Simulationsplattform VEROSIM. Insbesondere die Umgebungserfassung ist verantwortlich für den eingangs erwähnten kognitiven Prozess der Abstraktion und muss entsprechend umgesetzt werden.

Die Probleme der Umgebungserfassung, Positionsbestimmung und Kartenerzeugung sind nun keine künstlich geschaffenen Probleme, welche erst mit der Entwicklung neuer Sensoren und mobiler Roboter entstanden sind. Sie existieren schon viel länger und werden in verschiedenen Ausprägungen von fast allen sich frei bewegenden Lebewesen auf der Erde von Beginn an bewältigt. Auch der Mensch hat im Laufe der Zeit viele Methoden entwickelt um diese Fähigkeiten auszubauen. Beispielsweise als Entdecker ist neben einer möglichst genauen Lokalisierung auch eine zuverlässige Kartographierung von hoher Wichtigkeit. Denn wurde erst eine Entdeckung gemacht sollte es auch jeder Zeit wiedergefunden werden können. Um sich einen Weg zu merken und eine Position später wiederfinden zu können ist die intuitiv naheliegendste

Möglichkeit optisch markante Merkmale der Umgebung zu erfassen und sich deren Lage untereinander bezüglich der zu merkenden Position abzuspeichern. Eine Sammlung solcher Merkmale oder auch Landmarken bilden ein reduziertes Weltmodell, das unter anderem die wesentlichen Informationen zum Wiederfinden einer Position enthält. Ausgehend von der Frage: „Wo bin ich?“ können viele weitere Fragen aufgeworfen werden, die mit einer reinen Betrachtung des Problems der Lokalisierung nicht beantworten lassen. Dazu gehören Fragen wie:

- „Was kann ich hier tun?“
- „Welche Informationen kann ich aus der Umgebung ziehen?“
- „Welche Wege sind sicher oder befahrbar?“

Gegenstand dieser Arbeit soll nun aber nicht die Entwicklung eines weiteren, neuen Lokalisierungsalgorithmus sein, sondern es soll ein allgemeingültiges Konzept einer semantischen Umgebungserfassung- Beschreibung und Modellierung erarbeitet und umgesetzt werden. Ziel ist es eine Möglichkeit zu schaffen, die Umwelt als semantisches Modell vorliegen zu haben, das neben einer Lokalisierung auch weitere, komplexere Anwendungen zulässt, ähnlich wie bei der Einführung geographischer Karten, die dann dazu genutzt werden konnten auch Flächen zwischen den Wegen darzustellen und Planungen durchzuführen.

Ziel dieser Arbeit ist die Einführung und Umsetzung eines modularen Konzepts, das die drei zentralen Elemente Umgebungserfassung, Lokalisierung und Umweltmodellierung miteinander verknüpft und eine einheitliche und übergreifende Lösung für die verschiedensten Anwendungen schafft.



## Kapitel 2

# Grundlagen und Stand der Technik

Dieses Kapitel gibt einen Überblick über den derzeitigen Stand der Technik auf den wesentlichen Themengebieten dieser Arbeit, der Umgebungserfassung und Umweltmodellierung sowie der Lokalisierung. Durch eine Gegenüberstellung der aktuellen Verfahren in diesen Bereichen soll eine mögliche Tendenz zu den potenziell besten Ansätzen untersucht und identifiziert werden. Darüber hinaus wird an dieser Stelle die fachliche Grundlage geschaffen, auf der die Arbeit im Folgenden aufsetzt.

## 2.1 Sensorik

Die sensorische Umgebungserfassung stellt ein wesentliches Forschungsgebiet der mobilen Robotik dar. Nur wenn ein Roboter seine Umgebung erfassen kann, ist er in der Lage mit seiner Umwelt zu interagieren. Zu diesem Zweck sind mobile Roboter in der Regel mit einer Vielzahl von Sensoren ausgestattet. Dadurch ist es ihnen möglich Hindernisse noch vor einem Zusammenstoß zu erkennen [19, 20] oder notwendige Informationen über die Umgebung zu sammeln, um ihre Aufgaben erfüllen zu können. Mit der Komplexität dieser Aufgaben steigen auch die Anforderungen an die einzusetzende Sensorik. Denn, wo zum Vermeiden von Kollisionen noch einfache Abstandssensoren ausreichen, werden für die Lokalisierung und Interaktion deutlich detailliertere Informationen der Umgebung benötigt [21].

Seit einiger Zeit werden auch im Automobilbereich zunehmend mehr Sensoren zur Umgebungserfassung eingesetzt [22, 23]. In erster Linie sollen die Sensoren hier zur Steigerung der Sicherheit beitragen, indem

sie in Systemen zur Verminderung oder Vermeidung von Unfällen Verwendung finden. Aber auch für weiterführende Aufgaben, von der Aufbereitung von Umgebungsinformationen für den Fahrer [22] bis hin zum automatisierten Fahren finden Sensoren zur Umgebungserfassung ihre Verwendung.

Im Rahmen dieser Arbeit relevante Sensoren lassen sich wie folgt unterteilen:

1. Propriozeptive Sensoren: Sensoren dieses Typs messen interne Eigenschaften, Zustände bzw. Zustandsänderungen des Systems. Sie werden daher auch als *interne Sensoren* bezeichnet.
2. Exterozeptive Sensoren: Diese Sensoren erfassen die Umgebung und werden daher im Folgenden tiefergehend behandelt. Sie können weiter unterteilt werden in:
  - (a) Entfernungsmesssensorik (depth-aware oder depth-sensing sensors): In diese Gruppe fallen alle Sensoren, die eine Tiefenwahrnehmung der Umgebung ermöglichen.
  - (b) Abbildungssensorik (imaging sensors): Sensoren dieser Kategorie erzeugen ein (meist zweidimensionales) Abbild ihrer Umwelt ohne Tiefeninformationen.

### 2.1.1 Propriozeptive Sensoren

Die Klasse der propriozeptiven Sensoren ist zur Umgebungserfassung im Sinne des Konzepts dieser Arbeit, das in Kapitel 3 ausführlich behandelt wird, nicht relevant. Dennoch können diese Sensoren etwa bei der Lokalisierung unterstützend eingesetzt werden, wenn sie mit ihrer Messung Teile der Posenschätzung hilfreich ergänzen können. Daher werden diese Sensoren hier ebenfalls, wenn auch nicht in aller Tiefe, behandelt.

#### Empfänger für Globale Navigationssatellitensysteme (GNSS)

GNSS-Sensoren sind Empfänger für eines oder mehrere der globalen Navigationssatellitensysteme wie GPS, GLONASS oder Galileo. Die jeweiligen GNSS-Systeme bestehen aus mehreren Satelliten, die kontinuierlich ihren Identifikationscode, Informationen zu ihrer Bahnbewegung und

eine unter den Satelliten synchronisierte Uhrzeit aussenden. Anhand der jeweiligen Signallaufzeit für jedes empfangene Satellitensignal, ist es dem Empfänger möglich die eigene Position zu bestimmen. Theoretisch sind zur Bestimmung der eigenen 3D-Position die Signale von drei Satelliten ausreichend. In der Praxis werden jedoch mindestens vier Satelliten benötigt, da die Uhrzeit zwar zwischen den einzelnen Satelliten synchronisiert ist, jedoch zum Empfänger eine unbekannte Abweichung aufweist. Diese Zeitdifferenz ist eine weitere Unbekannte, die in das zu lösende lineare Gleichungssystem eingeht, um daraus eine Position zu bestimmen. Die Positionsinformationen werden im Standardformat NMEA 0183, spezifiziert durch die National Marine Electronics Association (NMEA), ausgegeben.

### **Inertialeinheit oder Inertial Measurement Unit (IMU)**

Inertialeinheiten oder IMUs werden häufig in der mobilen Robotik (sowohl zu Land als auch auf dem Wasser und in der Luft) verwendet. Dabei ist eine IMU nicht nur ein einzelner Sensor sondern eine Kombination mehrerer zum Teil unterschiedlicher Sensoren, die sich gegenseitig ergänzen. Für die Erfassung translatorischer Bewegungen im Raum werden in IMUs drei zu einander orthogonal ausgerichtete Beschleunigungssensoren verwendet. Durch diese Anordnung können Beschleunigungen in den drei Raumachsen  $x$ ,  $y$  und  $z$  erfasst und, durch Integration, eine Bewegung abgeleitet werden. Im Ruhezustand kann die Erdbeschleunigung gemessen und dazu genutzt werden, die relative Neigung der IMU zu erfassen. Für die Detektion von Rotationen um die Raumachsen werden, ebenfalls orthogonal zu einander ausgerichtet, drei Drehratensensoren (Gyroskopsensoren) verwendet, die durch das Messen der Corioliskraft ihre jeweilige Winkelgeschwindigkeit ableiten können.

In manchen IMUs werden zusätzlich Magnetfeldsensoren (Magnetometer) zur Messung des Erdmagnetfeldes verwendet, um neben den relativen Zustandsänderungen auch eine absolute Bezugsgröße zu erhalten. Die Erdbeschleunigung und das Erdmagnetfeld bilden damit ein Bezugssystem für die absolute Orientierung der IMU. Um auch eine absolute Positionsbestimmung zu ermöglichen werden IMUs teilweise ebenfalls mit GNSS-Sensoren versehen. Diese Varianten werden oftmals in sogenannten Trägheitsnavigationssystemen (Inertial Navigation Systems (INS)) verwendet.

### Magnetometer (Kompass)

Ein Kompass kann zur Orientierungsbestimmung genutzt werden, jedoch, im Unterschied zur IMU, nur in einem Freiheitsgrad (1DOF). Die Messung der lokalen Erdmagnetfeldausrichtung ist seit der griechischen Antike bekannt [24] und wird seitdem zur durchaus präzisen<sup>1</sup> Orientierungsbestimmung verwendet. Elektrische Magnetometer erreichen heutzutage Genauigkeiten  $< 1$  deg, vorausgesetzt es befinden sich keine ferro- oder elektromagnetischen Störquellen in der näheren Umgebung des Sensors. Gegenüber diesen Störungen sind die Sensoren überaus empfindlich. Aus diesem Grund werden sie in der Regel nur als ergänzende Sensoren (aiding sensors) beispielsweise für IMUs verwendet. Gerade in dieser Kombination ergänzen sich die verhältnismäßig langsame<sup>2</sup> und störanfällige aber dafür absolute Richtungsbestimmung des Magnetometers mit der vergleichsweise flinken und robusten, aber aufintegrierten, Richtungsbestimmung einer IMU auf ideale Weise.

### Hodometer (Odometer)

Eine einfache Bewegungsmessung für mobile, mit Rädern angetriebene Systeme, sind Hodometer. Unter der Hodometrie versteht man die Streckenmessung anhand von gezählten Radumdrehungen bei bekanntem Raddurchmesser (Abbildung 2.1 oben). Durch die gleichzeitige Messung von Umdrehungen bei zweispurigen Fahrzeugen auf beiden Seiten können neben der zurückgelegten Strecke auch, unter der Vernachlässigung von

Schlupf, Drehwinkel abgeleitet, und somit eine Bewegungsrekonstruktion in der Ebene gewährleistet werden (Abbildung 2.1 unten). Diese Art der relativen Lokalisierung ist sehr einfach aber auch stark fehlerbehaftet. Insbesondere bei ungünstigen Kombinationen aus Radaufhängung, Bodenbeschaffenheit, Antriebsart, Rad- und Bodenmaterial, können schnell gravierende Messfehler entstehen. Aus diesem Grund stellen Hodometer zwar eine einfache Art der Bewegungsmessung, aber ebenso eine

---

<sup>1</sup>Im frühen 12. Jahrhundert waren Kompass mit einer Einteilung in 24 Richtungen, also in Stufen von jeweils  $15^\circ$  bereits üblich.

<sup>2</sup>Ein digitaler Kompass wird meist mit Raten um 100Hz abgetastet, während Drehraten- und Beschleunigungssensoren mit Abtastraten um 10kHz betrieben werden (Quelle: [https://www.sbg-systems.com/docs/Ellipse\\_Series\\_Leaflet.pdf](https://www.sbg-systems.com/docs/Ellipse_Series_Leaflet.pdf)).

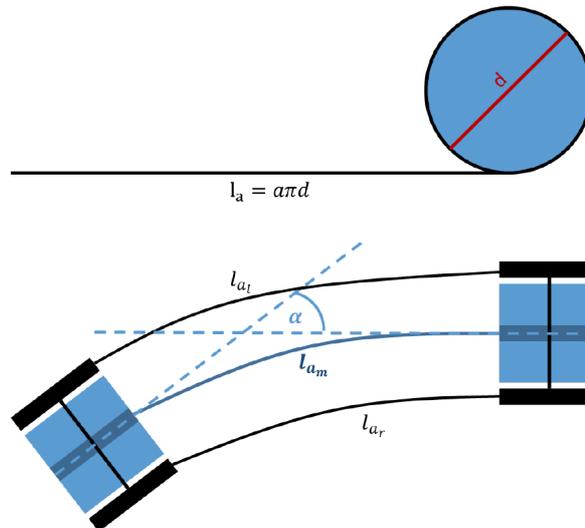


ABBILDUNG 2.1: Messprinzip eines Odometers mit einem bzw. zwei Messrädern. Oben: die Anzahl der vollständigen Radumdrehungen  $a$  wird gemessen; unten: die Radumdrehungen für das linke  $a_l$  und das rechte Rad  $a_r$  werden auf ein gedachtes mittleres Rad umgerechnet.

ungenau, dar. Allerdings reicht die Genauigkeit, entsprechende Umgebungseigenschaften vorausgesetzt, dazu aus, um die Lokalisierung zu unterstützen oder zu ergänzen, wo andere Lokalisierungsverfahren nicht möglich sind. Zum Beispiel scheitert eine landmarkenbasierte Lokalisierung sobald ein Bereich befahren wird, der nicht über ausreichend viele Landmarken verfügt. Dann kann eine einfache Fahrzeughodometrie dazu genutzt werden, die Fahrzeugeigenbewegung solange fortzuführen bis wieder genügend Landmarken ins Sichtfeld der übrigen Sensoren gelangen, um eine landmarkenbasierte Lokalisierung fortzusetzen.

Darüber hinaus wird im Automotivebereich die Fahrzeughodometrie dazu verwendet, um bei fehlendem oder fehlerhaftem GNSS-Signal (z.B. in Tunneln und Unterführungen) die Fahrzeugposition weiter zu schätzen. Diese Schätzung wird meist unterstützt durch eine geeignete Filterung, wie etwa ein Kalmanfilter, dessen Ansatz in Abschnitt 2.3.2 erläutert wird.

## 2.1.2 Entfernungsmesssensorik

Entfernungsmesssensorik nimmt in dieser Arbeit eine wichtige Stellung ein und soll daher etwas eingehender behandelt werden. Da die technische Realisierung aber in der Regel keinen Einfluss auf die Repräsentation der Messergebnisse hat (abgesehen von Genauigkeit und Rauschverhalten), soll der Fokus auf der Art der distanzbasierten Umgebungsabbildung, also den resultierenden Sensordaten, liegen. Zum Beispiel kann ein Landmarkendetektor, der 2,5D-Tiefenkarten als Datengrundlage erwartet gleichermaßen auf Tiefenkarten von Stereokameras, ToF-Kameras und RGB-D Kameras arbeiten, auch wenn sich die Art der Datenerfassung in diesen drei Sensoren grundlegend unterscheidet. Eine (eher seltene) Ausnahme bilden Sensoren oder Sensorsysteme, die ungeordnete Punktwolken aufnehmen und deren Daten dann nicht in Form von 2,5D-Punktwolken oder 1,5D-Tiefenprofilen darstellbar sind (z.B. [25]).

### LiDAR-Sensoren

Eine schnelle und äußerst genaue Art der Entfernungsmessung ist das sogenannte Light Detection And Ranging (LiDAR) Verfahren. Unter diesem Begriff werden allen Sensoren zusammengefasst, die zur Entfernungsmessung Licht aussenden, die von der Umgebung zurückgeworfenen Reflexionen detektieren und, die Entfernung der umgebenden Objekte ermitteln. Dabei unterscheiden sich zwischen den verschiedenen Sensortypen die Art der Lichtquelle genauso wie die Detektionsvorrichtung und das Messverfahren zur Entfernungsbestimmung.

Für die Messung weiter Entfernungen mit einer Genauigkeit im Meter bis Zentimeterbereich wird das Lichtlaufzeitverfahren zur Entfernungsbestimmung verwendet. Dabei wird die Zeitdifferenz zwischen ausgesendetem und empfangenem Licht gemessen und, da die Lichtgeschwindigkeit als konstant angenommen werden kann, wird daraus die vom Licht zurückgelegte Entfernung  $l$  berechnet durch:

$$l = \frac{c}{2} \Delta t \quad (2.1)$$

Da für die Lokalisierung meist kürzere Entfernungen mit höherer Genauigkeit gemessen werden sollen, wird hier typischerweise eher das

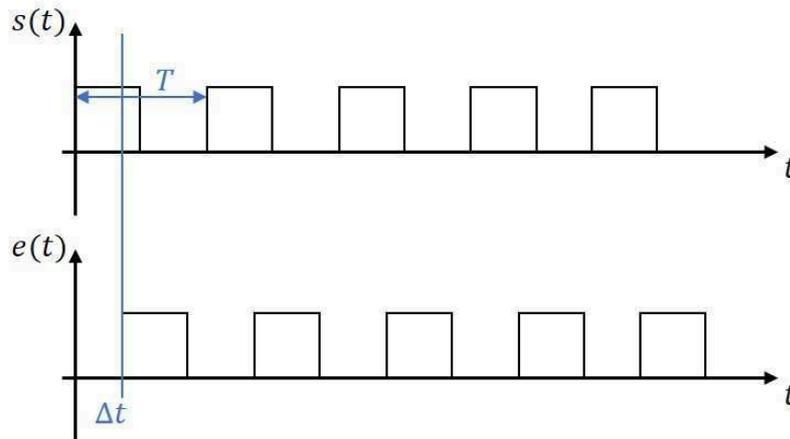


ABBILDUNG 2.2: Entfernungsmessung mittel Phasenmodulation. Ein Strahl wird mit der Frequenz  $f = 1/T$  moduliert und nach einer Zeit  $\Delta t$  empfangen.

Verfahren der Phasenverschiebung angewandt. Dazu wird kein einzelner Lichtimpuls ausgesendet sondern ein kontinuierlicher und mit einer Frequenz  $f = 1/T$  modulierter Strahl (Abbildung 2.2).

Anstelle einer Zeitmessung wird die Phasendifferenz  $\phi$  zwischen ankommendem und ausgesendetem Licht gebildet. Rechnerisch erhält man diese durch die Gleichung:

$$\phi = \frac{\Delta t}{T} 2\pi \quad (2.2)$$

Dann kann die Distanz durch

$$s_n = \frac{cT}{4\pi} \cdot (\phi + 2n\pi) = \frac{c}{2} \cdot (nT + \Delta t) \quad (2.3)$$

berechnet werden.

Zu beachten ist hier, dass die resultierende Entfernungsmessung nicht eindeutig ist, da die Phasenmodulation periodisch erfolgt. In Gleichung 2.3 spiegelt sich dies durch den Term  $\phi + 2n\pi$  wider, der für alle  $n \in \mathbb{N}$  denselben Tangenswert zurückgibt. Dieses Phänomen wird in der Literatur als *phase wrapping* bezeichnet. Somit hat jede Modulationsfrequenz  $f$  eine Maximaldistanz  $d_{max} = \pi$ . Entsprechende Sensoren sind dann nur für Entfernungen bis  $d_{max}$  spezifiziert.

LiDAR-Sensoren liefern in der Regel für jeden Messpunkt eine Entfernung (Tiefe). Diese Messwerte werden entweder als Feld (1-2 dimensional) oder als 3D-Punkte in einer Punktwolke zurückgegeben. Letztere muss nicht zwangsläufig in einer gerasterter Form vorliegen.

**Laserscanner** Ein weit verbreiteter Sensortyp ist der Laserscanner oder LaDAR (Laser Detection And Ranging). Laserscanner senden einen Laserstrahl in eine Richtung aus und detektieren das reflektierte Licht in einer entsprechenden Empfangseinheit. Kombiniert mit der Richtung des ausgesendeten Lichts resultiert die gemessene Entfernung in einem dreidimensionalen Punkt, der in einem Referenzkoordinatensystem, abhängig vom optischen Zentrum des Sensors, angegeben wird. In der Regel wird die Richtung des ausgesendeten Lichts in Winkeln zwischen der Aussenderichtung und der festgelegten Referenzsichtachse in 2D-Polarkoordinaten bzw. in 3D-Kugelkoordinaten angegeben. Für die Anwendung in kartesischen Koordinatensystemen ist eine entsprechende Umrechnung notwendig. Man unterscheidet Laserscanner meist anhand der Art und Weise, wie die Laserstrahlen ausgesendet werden:

Es gibt 2D-Laserscanner, die einen Laserstrahl über einen sich um eine Achse drehenden Spiegel in einer Ebene aussenden. Das Messergebnis ist meist ein Vektor von Tiefenwerten, der anhand des Öffnungswinkels in der Ebene und der Winkelauflösung (dem Winkelabstand zweier benachbarter Messstrahlen) entsprechend in eine Menge von 3D-Punkten umgewandelt werden kann.

Für einen erweiterten vertikalen Scanbereich werden auch 3D-Laserscanner hergestellt, die nicht nur in einer Ebene Scannen, sondern simultan in mehrere Ebenen (Layer oder manchmal auch Channel) scannen. Üblich sind bereits bis zu 64 Ebenen in kompakten Scannern, die z.B. für den Automotivebereich entwickelt und gefertigt werden. Hier werden meist mehrere, in Gruppen angeordnete Emittter und Detektoren verwendet. Das grundsätzliche Scanprinzip ist aber vergleichbar zum 2D-Laserscanner. Die Sensordaten werden auch hier in Form von Tiefenwerten ausgegeben und können als Tiefenkarte oder Punktwolke genutzt werden.

Ein vollständiger 3D-Scan der Umgebung ist mit 3D-Scannern möglich, die einen Laserstrahl über ein senkrecht rotierendes Prisma ablenken und diese Einheit zusätzlich horizontal rotieren (siehe Abbildung 2.3). Diese Anordnung ermöglicht einen nahezu 360° Rundumscan. Lediglich

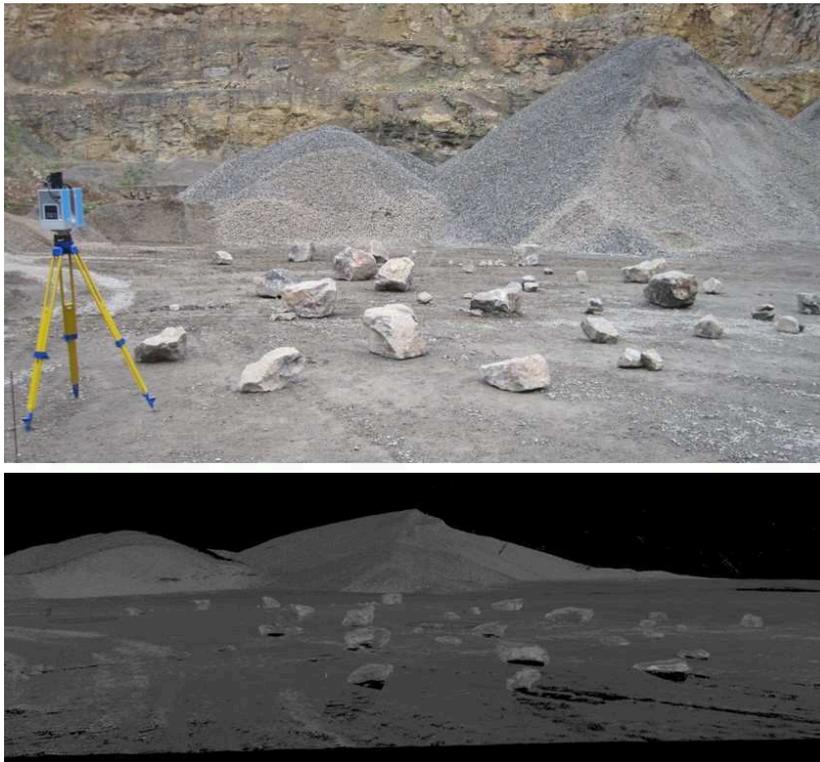


ABBILDUNG 2.3: 3D-Laserscanner „Imager 5006h“ der Firma Zoller+Fröhlich, sowie eine aufgenommene Punktwolke.

die Scannerbasis sorgt für Verschattungen genau unter dem Scanner. Diese Art des Scannens dauert je nach Scanauflösung zwischen einigen Sekunden und mehreren Minuten. Aus diesem Grund sind die 3D-Rundum-Laserscanner für mobile Anwendungen in der Regel nicht gebräuchlich. Stattdessen werden sie zur Aufnahme statischer Szenen wie Gebäude und Produktionsstätten aber auch bei Verkehrsunfällen oder Tatorten verwendet. Durch die hohe Anzahl an Messwerten<sup>3</sup> werden bei diesen Scannertypen meist vollständige Scans auf ein internes Speichermedium geschrieben und anschließend als Punktwolke ausgegeben. Damit ist eine eigene Umrechnung aus Distanzwerten, Scanauflösung und Kugelkoordinaten hin zu kartesischen Messpunkten nicht mehr notwendig.

<sup>3</sup>Der 3D-Laserscanner „Imager 5006“ der Firma Zoller+Fröhlich misst beispielsweise über 1,3 Mio. Punkte bei einem Vollscan höchster Auflösung.

**PMD-Sensoren und Time-of-Flight Kameras** Die Entfernungsmessung durch das Aussenden und Empfangen von Licht funktioniert nicht nur sequentiell mit einzelnen Messstrahlen, wie bei Laserscannern, sondern kann auch mit speziellen Kamerasystemen für ein gesamtes Bild simultan aufgenommen werden. Für diese Sensoren sind zwei Begriffe gebräuchlich: Die „Time-of-Flight-Kamera“ sowie der „PMD-Sensor“. Dabei beschreibt ersteres das grundlegende Messprinzip und letzteres, also das Photonik Mixer Device (PMD) eine konkrete technische Realisierung des Sensors, wo auf dem CMOS-Chip für jeden Pixel die Korrelation zwischen ausgesendeten und einfallendem Licht bestimmt wird. Beide Begriffe werden oftmals synonyme für alle Sensoren dieser Art verwendet, auch wenn sich die jeweilige Technik unterscheidet. Das grundsätzliche Messprinzip ist jedoch bei allen Sensoren identisch. Dazu wird eine große Beleuchtungseinheit verwendet, die Licht in die gesamte aufzunehmende Szene aussendet und ein Sensor, der, ähnlich wie eine gewöhnliche digitale Kamera aufgebaut ist, und das Licht der ausgesendeten Wellenlänge in einem Matrixraster empfängt. Die Entfernung wird wie beim Laserscanner meist nicht durch Lichtlaufzeitverfahren bestimmt sondern über das Verfahren der Phasenverschiebung. Aufgrund des oben erläuterten „phase wrapping“ sind diese Sensoren daher nur bis zu einer, von der verwendeten Wellenlänge abhängigen, maximalen Entfernung spezifiziert (üblicherweise 10m bei der Verwendung von Licht im Infrarot-Wellenlängen-Bereich).

**RGB-D Kameras** Ein anderes Verfahren zur Entfernungsmessung mit Hilfe kameraähnlicher Sensorik sind die als RGB-D Kameras bezeichneten Sensoren. Dabei steht RGB-D für die 3 Kanäle einer gewöhnlichen Farbkamera Rot, Grün, Blau (RGB) ergänzt durch einen weiteren Kanal für die Entfernung (Depth: D). Im Gegensatz zur Entfernungsmessung durch die Detektion von ausgesendeten Lichtimpulsen oder Phasenverschiebungen in der Beleuchtung wird bei der RGB-D Kamera mit Hilfe eines (Infrarot-) Lasers ein Punktmuster in die Szene projiziert und im Kamerabild erfasst. Da das Projektionsmuster im Vorfeld bereits bekannt ist, kann anhand der Position der einzelnen Punkte im Bild per Triangulation die Entfernung des projizierten Punkts ermittelt werden. Diese Art der Entfernungsmessung wurde populär mit der Einführung

der ersten Kinect-Kamera von Microsoft im Jahr 2010<sup>4</sup>. Aufgrund des, im Vergleich zu am Markt erhältlichen ToF-Kameras, günstigen Preises sowie der Möglichkeit zu gleichzeitigen Aufnahme von Farb- und Tiefenwerten erfuhr die Kinect im Bereich der mobilen Robotik eine weite Verbreitung. Es wurden demnach viele Arbeiten zum Thema Lokalisierung und Objekterkennung mit Hilfe von RGB-D Sensoren veröffentlicht. Die Nachteile der Tiefenmessung durch Musterprojektion liegen in der abnehmenden Auflösung des Tiefenkanals mit wachsender Entfernung sowie der Beschränkung auf Indoor-Anwendungen (da das projizierte Muster bei Tageslicht u.U. nicht mehr eindeutig detektiert werden kann). Die maximale Entfernung ist hier abhängig vom Basisabstand zwischen Projektor und Kamera und liegt üblicherweise im Bereich von zehn Metern.

### Stereooptische Kamerasysteme

Für die Tiefenwahrnehmung in einer Szene werden häufig stereooptische Systeme (Stereokameras) verwendet. Sie sind hier in die Kategorie der Entfernungsmesssensorik eingeteilt, obwohl sie nur indirekt über eine sogenannte Tiefen-Awareness verfügen. Sie sind relativ einfach aufgebaut und bestehen im Grunde lediglich aus zwei monooptischen Kameras, mit einem festen relativen Abstand zueinander, der als Basisabstand (oder englisch: baseline)  $b$  bezeichnet wird<sup>5</sup>. Die Tiefenberechnung erfolgt durch eine pixelweise Korrespondenzsuche in den zeitgleich aufgenommenen Einzelbildern des Kameraverbunds. Die Pixelverschiebung korrespondierender Bildpunkte auf den Epipolarlinien (das sind die Schnittlinien der Epipolarebene, aufgespannt durch die beiden Projektionszentren der Kameras und dem betrachteten Objektpunkt, mit der jeweiligen Bildebene der Kamera) beider Kameras wird als Disparität  $d$  angegeben. Die metrische Tiefe  $z$  berechnet sich aus der Disparität in

---

<sup>4</sup>Die im Jahr 2014 eingeführte neue Variante der Kinect verwendet nicht mehr die Musterprojektion der ersten Version zur Tiefenmessung sondern das zuvor behandelte ToF-Verfahren

<sup>5</sup>Eigentlich müsste diese Verschiebung als „horizontale Baseline“ angegeben werden. Da aber in der Praxis gewöhnlich fast nur horizontal zueinander verschoben wird, lässt man das „horizontal“ meist weg. Nur wenn es sich um eine anderweitige Verschiebung handelt wird dies explizit angegeben. Es gibt auch Stereokameras, die eine vertikale Baseline verwenden. Das ändert die Vorgehensweise bei der Tiefenberechnung nur dahingehende, dass Zeilen durch Spalten ersetzt werden.

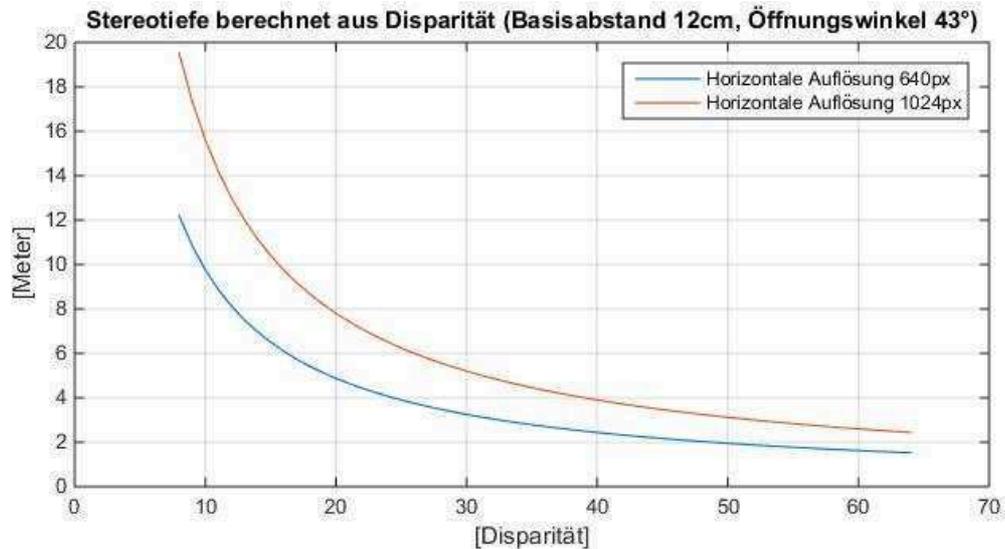


ABBILDUNG 2.4: Darstellung der nichtlinearen Abhängigkeit zwischen Disparität und metrischer Tiefe

Abhängigkeit zur Brennweite  $f$  der Objektivs und der Baseline  $b$  durch:  $z(d) = fb/d$ . Für eine korrekte Korrespondenzsuche werden die beiden Kameras aufeinander kalibriert (siehe hierzu auch Kapitel 2.1.3), d.h. die Einzelbilder werden rektifiziert, so dass eine Bildzeile des einen Bildes auf der Höhe der gleichen Bildzeile des anderen Bildes steht. Damit wird die Korrespondenzsuche auf einen fest definierten Suchbereich eingegrenzt. Zu dieser Suche existieren derzeit zahlreiche Publikationen, die sich im Bereich der Genauigkeit und Laufzeit teils erheblich unterscheiden. Eine Gegenüberstellung und Auswertung zahlreicher Verfahren wird in [26] durchgeführt.

Zu beachten ist, dass die Tiefenauflösung nicht linear mit der Entfernung abnimmt oder gar konstant ist. Abbildung 2.4 veranschaulicht die nichtlineare Abhängigkeit zwischen der Tiefe in Metern und der Disparität in Pixeln für die horizontalen Bildauflösungen von 640 und 1024 Pixeln.

Daher nimmt die Tiefenauflösung mit steigender Entfernung zunächst rapide ab um anschließend gegen eine Disparität von Null zu konvergieren, was einer nicht mehr detektierbaren Verschiebung vom linken zum rechten Bild entspricht.

Für eine sensorunabhängigen Weiterverarbeitung der Daten muss die resultierende Disparitätenkarte in eine metrische Tiefenkarte umgerechnet werden. Zusammen mit dem Referenzbild (meist das linke

Stereobild) stehen anschließend Farb- bzw. Helligkeitsinformationen des Bildes zusammen mit den berechneten Tiefeninformationen zur Verfügung.

### **Nicht-optische Entfernungssensoren**

Neben den bisher behandelten optischen Sensoren zur Entfernungsmessung gibt es weitere Sensoren, die ebenfalls das Laufzeitverfahren verwenden. Im Gegensatz zu LiDAR-Sensoren wird hier jedoch kein Licht ausgesendet und empfangen sondern Schall-, Radar- oder Funkwellen.

**Ultraschallsensoren** Ultraschallsensoren senden Wellen im Frequenzbereich zwischen 20 kHz und 1 GHz [27] (oberhalb des vom Menschen akustisch wahrnehmbaren Frequenzbereichs) aus und detektieren in der Umgebung reflektierte Wellen. Da die Laufzeit von Schall in der Luft (ca. 343 m/s) gering genug ist, um die Verzögerung auch bei geringen Distanzen noch messen zu können, ist der technische Aufwand erheblich geringer als bei LiDAR-Sensoren. Aus diesem Grund sind Ultraschallsensoren einfacher zu fertigen und zu geringeren Preisen zu erhalten. Nachteil dieser Sensoren im Vergleich zu LiDAR ist die kegelförmige Ausbreitung der Schallwellen, d.h. das ausgesendete Signal kann nicht gut gebündelt werden, wie etwa Laserstrahlen, und ermöglicht daher keine punktgenauen Messungen. Gemessen wird stattdessen in sogenannten Schallkeulen mit einem Öffnungswinkel zwischen  $5^\circ$  und  $8^\circ$ . Von dieser Einschränkung abgesehen, können Ultraschallsensoren im Nahbereich aufwandsarm eingesetzt werden und sind daher als Abstandssensoren im Automotive- sowie im industriellen Produktionsbereich weit verbreitet. Eine semantische Objekterkennung nur anhand von Ultraschalldaten ist jedoch (noch) nicht möglich. Daher wird im Folgenden nicht tiefergehend auf diesen Sensortyp eingegangen.

**Radarsensoren** Radar („radio detection and ranging“) verwendet Funkwellen zur Detektion von Objekten und Hindernissen. Neben Entfernungen können ebenfalls Geschwindigkeiten sowie Form und Größe von Objekten bestimmt werden [27]. Wie beim LiDAR unterscheidet man zwischen Sensoren, die Impulse aussenden und mittels Laufzeitverfahren die Entfernung von Hindernissen ermitteln (Impuls-Radar) und solchen, die kontinuierlich phasenmodulierte Funkwellen aussenden und die

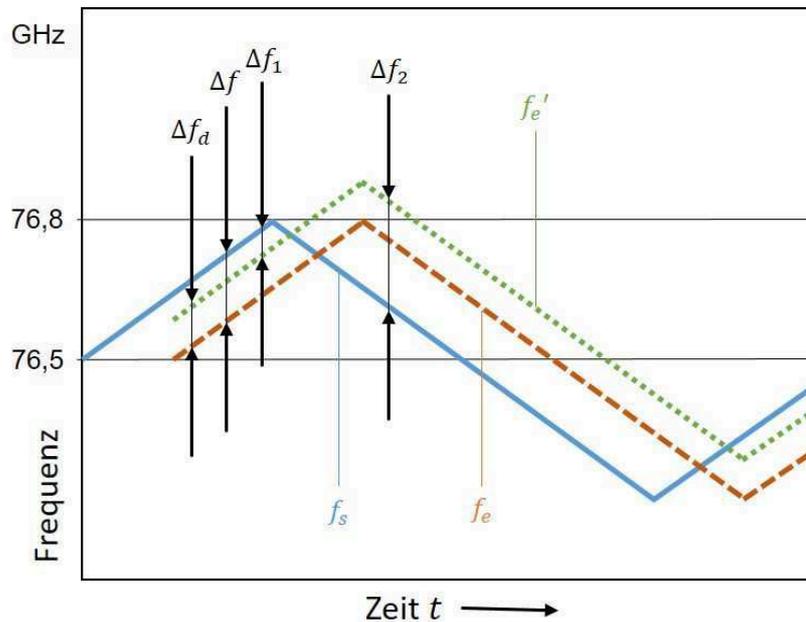


ABBILDUNG 2.5: Aus Doppler-Effekt resultierende Frequenzverschiebung beim CW-Radar. Sendesignal:  $f_s$ , Empfangssignal bei gleicher GESchwindigkeit:  $f_e$ , Empfangssignal bei vorhandener Relativgeschwindigkeit:  $f_e'$ .

Entfernung mittels Phasendifferenzverfahren bestimmen (Continuous Wave (CW) Radar). Die Entfernungsbestimmung unterscheidet sich bei beiden Varianten nicht von der der LiDAR-Sensoren.

Durch die Nutzung des Doppler-Effekts können Radarsensoren mit nur einer Messung neben der Entfernung auch die Relativgeschwindigkeit  $v_{rel}$  des detektierten Objekts bestimmen. Das Signalecho erfährt für ein sich relativ zum Radarsensor bewegendes Objekt gegenüber dem abgestrahlten Signal eine Frequenzverschiebung  $\Delta f_d$  (vgl. Abbildung 2.5). Diese beträgt bei den relevanten Differenzgeschwindigkeiten

$$\Delta f_d = -2f_s \frac{v_{rel}}{c}. \quad (2.4)$$

Bei den im Automotive-Bereich gebräuchlichen Radarfrequenzen von  $f_C = 76,5$  GHz ergibt sich eine Frequenzverschiebung von  $f_D \approx -510 \cdot v_{rel}$  pro Meter [28].

Als dritte Basisgröße ist die seitliche Lage des detektierten Objekts bestimmbar. Dazu muss das Radarsignal in verschiedene Richtungen abgestrahlt, und aus den Echosignalen die Richtung mit der stärksten

Reflektion bestimmt werden. Dazu ist entweder ein schnelles Schwenken („Scannen“) eines Strahls oder eine mehrstrahlige Antennenanordnung notwendig [28]. Das resultierende Sensordatenformat ist dann vergleichbar mit den Scanprofilen eines 2D-Laserscanners. Im direkten Vergleich ist die Winkelauflösung des Radarsensors allerdings deutlich geringer.

### 2.1.3 Optische Abbildungssensorik

In diesem Abschnitt werden Sensoren behandelt, die ihre Umgebung mit Hilfe einer geeigneten Optik auf einen Bildsensor projizieren. Das Ergebnis ist eine zweidimensionale Abbildung der Umgebung in einem Bild. Je nach Bildsensor werden im Bild die Helligkeit, Farbe oder die Infrarotstrahlung der Szene abgebildet. Das Prinzip der optischen Abbildung ist unabhängig vom Bildsensor und beeinflusst hauptsächlich die weitere Verarbeitung der Sensordaten. Aus diesem Grund soll an dieser Stelle die Bildprojektion im Vordergrund stehen und nicht tiefer in die technischen Details der Sensorchips eingestiegen werden.

#### Lochkameramodell und Kameraparameter

Um die Projektion einer dreidimensionalen Szene in eine zweidimensionale Bildebene zu beschreiben, wird kein kompliziertes Kameramodell benötigt. Daher soll an dieser Stelle nur ein kleiner Überblick gegeben werden, der für das Verständnis der Methoden in dieser Arbeit ausreichen soll. Dieser Überblick ist angelehnt an die Arbeit [29] von Ma et.al. sowie an die Bearbeitung des Themas in [30].

Das einfache Modell einer Kamera mit dünner Linse ist schematisch in Abbildung 2.6 aufgezeigt. Dort wird ein Weltpunkt  $X$  auf einen Bildpunkt  $x$  abgebildet. Man sagt auch,  $x$  ist das *Bild* vom Objekt  $X$ . Aus den Seitenverhältnissen der ähnlichen Dreiecke  $A$  und  $A'$  lässt sich die Gleichung

$$\frac{z - f}{y} = \frac{f}{Y} \quad (2.5)$$

aufstellen, und aus den Seitenverhältnissen der ähnlichen Dreiecke  $B$  und  $B'$  die Gleichung

$$\frac{z}{y} = \frac{Z}{Y} \Leftrightarrow Y = \frac{Zy}{z}. \quad (2.6)$$

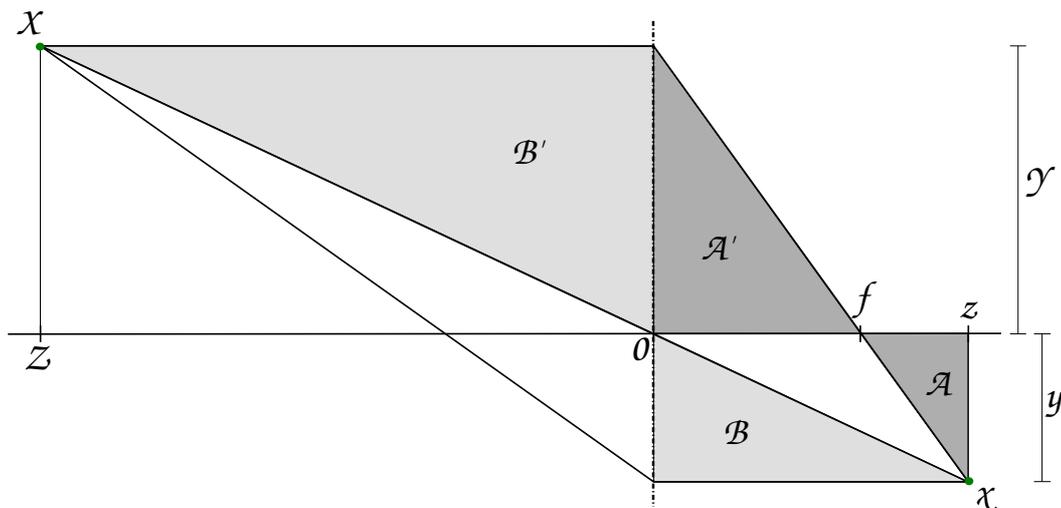


ABBILDUNG 2.6: Ein Weltpunkt  $X$  wird mittels einer dünnen Linse mit Brennpunkt  $f$  auf den Bildpunkt  $x$  abgebildet. Die Linse (gestrichelte Linie) befindet sich auf der Höhe 0 der optischen Achse (waagerechte Linie). In Entfernung  $Z$  zur Linse befindet sich der Weltpunkt  $X$ , und in Entfernung  $z$  der Bildpunkt  $x$ .  $Y$  und  $y$  geben den Abstand des Weltpunkts bzw. des Bildpunkts zur optischen Achse an.

Wird Gleichung (2.6) in Gleichung (2.5) eingesetzt, erhält man

$$\begin{aligned}
 \frac{z-f}{y} &= \frac{fz}{Zy} \\
 \Leftrightarrow z-f &= \frac{fz}{Z} \\
 \Leftrightarrow \frac{z}{f} - 1 &= \frac{z}{Z} \\
 \Leftrightarrow \frac{1}{Z} + \frac{1}{z} &= \frac{1}{f}
 \end{aligned} \tag{2.7}$$

die *Fundamentalgleichung für dünne Linsen*.

Läuft die Größe der Linsenöffnung gegen Null, ergibt sich das einfachste von allen Kameramodellen: das *Lochkameramodell*. In diesem Modell müssen alle Strahlen durch das optische Zentrum 0 laufen und werden daher auch nicht gebrochen. Eine Linse wird nun nicht mehr benötigt und der Abstand der Bildebene zur Linse wird als *Brennweite*  $f$  bezeichnet, da

dies ein kamerainterner Parameter ist, und alle Bildpunkte den selben Tiefenwert  $z = f$  besitzen. Damit lässt sich Gleichung (2.6) umformen zu

$$y = f \frac{Y}{Z}.$$

Für zweidimensionale Bilder erhält man die Koordinaten des Bildpunkts  $\vec{x} = (x, y)^T$  durch die Gleichung

$$x = f \frac{X}{Z}, y = f \frac{Y}{Z}, \quad (2.8)$$

die als *perspektivische Projektion* bezeichnet wird. Damit lässt sich die Projektion des Punkts  $\vec{X} = (X, Y, Z)^T$  in den Bildpunkt  $\vec{x} = (x, y)^T$  kompakter schreiben als:

$$\vec{x} = \begin{pmatrix} x \\ y \end{pmatrix} = \frac{f}{Z} \begin{pmatrix} X \\ Y \end{pmatrix}.$$

In homogenen Koordinaten umgeformt lautet die Gleichung dann

$$Z \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \underbrace{\begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_P \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}. \quad (2.9)$$

Da die Tiefe, also der  $Z$ -Wert des Punkts  $\vec{X}$  in der Regel nicht bekannt ist, ersetzt man ihn durch ein positives Skalar  $\lambda \in \mathbb{R}_{>0}$ , und schreibt so kurz:  $\lambda \vec{x} = P \vec{X}$  mit der Projektionsmatrix  $P$  aus Gleichung (2.9).

Die Matrix  $P$  wird aufgeteilt in eine Kalibrierungsmatrix  $K_f$  und die *Kanonische Projektionsmatrix*  $P_0$  der Form

$$K_f := \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{3 \times 3}, P_0 := \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{(I_3 | 0_3)} \in \mathbb{R}^{3 \times 4}. \quad (2.10)$$

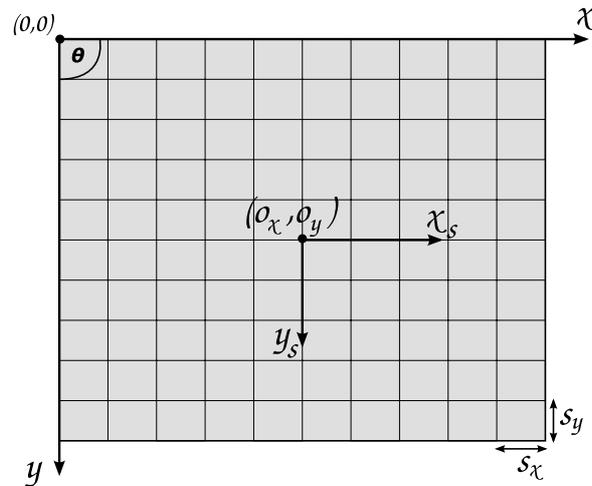


ABBILDUNG 2.7: Das lokale Bildkoordinatensystem  $(x_s, y_s)$  wird relativ zum Koordinatensystem  $(x, y)$  durch den Offset  $(o_x, o_y)$  und die Pixelbreite  $s_x$  und Pixelhöhe  $s_y$  beschrieben. Der Winkel  $\theta$  gibt den Winkel zwischen den Koordinatenachsen an.

Dann gilt  $P = K_f P_0$ , und die perspektivische Projektion kann letztendlich geschrieben werden als

$$\lambda \vec{x} = K_f P_0 \vec{X}. \quad (2.11)$$

Nun ist die Brennweite nicht der einzige intrinsische Parameter von Interesse. Das lokale Koordinatensystem des Bildes selbst, wird auch von der Kamera festgelegt und gehört deshalb ebenso zu den intrinsischen Parametern. Abbildung 2.7 zeigt dieses Bildkoordinatensystem, samt seinen Parametern.

Dazu gehört die Pixelbreite  $s_x$  und Pixelhöhe  $s_y$  sowie die lokalen Koordinaten  $o := (o_x, o_y)$  des Hauptpunkts der optischen Achse (*principal point*), das ist der Schnittpunkt der optischen Achse mit der Bildebene. Damit lassen sich die tatsächlichen Koordinaten  $\mathbf{x}$  des Bildpunkts ermitteln durch:

$$\mathbf{x} = \vec{x} + o.$$

Zur Unterscheidung soll  $\vec{x}$  den Punkt in Koordinaten, relativ zum principle point, darstellen und  $\mathbf{x}$  den tatsächlichen Bildpunkt.

Wenn zuletzt nicht immer davon ausgegangen werden soll, dass die Bildachsen senkrecht aufeinander stehen, so lässt sich der sogenannte *Skew Faktor*  $s_\theta := \cot(\theta)$ , mit dem Winkel  $\theta$  zwischen den Bildachsen

angeben. Meistens ist der Winkel  $\theta$  sehr nah an  $90^\circ$ , und somit  $s_\theta$  sehr nah an Null. Damit ergibt sich die Kalibrierungsmatrix

$$K_s := \begin{bmatrix} s_x & s_\theta & o_x \\ 0 & s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{3 \times 3}. \quad (2.12)$$

Kombiniert man die Kalibrierungsmatrizen  $K_f$  und  $K_s$  mit der Projektionsmatrix  $P$  erhält man für die perspektivische Projektion insgesamt  $\lambda \mathbf{x} = K_s K_f P_0 \vec{\mathbf{X}}$  oder ausführlich:

$$\lambda \mathbf{x} = \begin{bmatrix} s_x & s_\theta & o_x \\ 0 & s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \vec{\mathbf{X}}.$$

Die Kalibrierungsmatrix, die durch Multiplikation der Matrizen  $K_s$  und  $K_f$  entsteht wird schlicht mit  $K$  bezeichnet. Sie hat die Form

$$K := K_s K_f = \begin{pmatrix} s_x & s_\theta & o_x \\ 0 & s_y & o_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} f s_x & f s_\theta & o_x \\ 0 & f s_y & o_y \\ 0 & 0 & 1 \end{pmatrix}, \quad (2.13)$$

und wird, da sie alle intrinsischen Parameter der Kamera beinhaltet, als intrinsische Parametermatrix (*intrinsic parameter matrix*) oder häufig auch als Kalibrierungsmatrix (*calibration matrix*) der Kamera bezeichnet.

Es existieren bereits viele zuverlässige Verfahren, um aus einer Reihe von Kameraaufnahmen die intrinsischen Parameter der Kamera zu schätzen. Meist wird dazu ein Kalibrierungsobjekt verwendet, das aus verschiedenen Perspektiven und unter konstanter Kamerakonfiguration aufgenommen wird. Aufgrund der Korrespondenzenbildung unter allen Aufnahmen können dann die intrinsischen Kameraparameter geschätzt werden, wie etwa in [31, 32].

Als Kalibrierungsobjekt kommt meist ein Schachbrettmuster in einer Ebene zum Einsatz, dessen Ausrichtung zur Kamera über die verschiedenen Aufnahmen verändert wird. Aber auch andere Objekte können zur Kalibrierung verwendet werden, wie etwa die in Kapitel 4.2.4 verwendeten Augmented Reality Marker, die durch ihrer einheitliche und bekannte Geometrie ebenso dazu geeignet sind, eine Kamera anhand mehrerer Aufnahmen zu kalibrieren.

Ein weiteres Kalibrierungsverfahren stellt die in [33, 34] beschriebene lineare Selbstkalibrierung anhand des sogenannten absoluten Kegelschnitt (*absolute conic*) dar. Eine Aufarbeitung dieses Verfahrens befindet sich in [30].

Da mittlerweile zahlreiche Verfahren zur Bestimmung von intrinsischen Kameraparametern, gepaart mit der Möglichkeit zur einfachen Rektifizierung von Kamerabildern anhand der identifizierten Parameter, existieren (z.B. in der OpenCV library [35]) soll das Thema der Kalibrierung an dieser Stelle abgeschlossen werden. Im Folgenden wird davon ausgegangen, dass die Kameraparameter der verwendeten Kameras bekannt bzw. einfach bestimmbar sind und die Eingangsbilder der Kameras vor der Weiterverarbeitung gemäß der angesprochenen Verfahren rektifiziert wurden. Das erlaubt die Anwendung des oben beschriebenen Lochkameramodells für alle weiteren Betrachtungen.

### 2.1.4 Gegenüberstellung der verschiedenen Sensoren

Alle in diesem Abschnitt behandelten Sensoren sollen an dieser Stelle noch einmal gegenübergestellt und verglichen werden. Dazu gibt Tabelle 2.1 eine Übersicht. Da die Sensordatenverarbeitung im stärkeren Fokus dieser Arbeit steht, als die jeweilige Funktionsweise der Sensoren, wurden die Einträge von Tabelle 2.1 nach dem Datenformat sortiert, in dem die jeweiligen Sensoren ihre gemessenen Daten typischerweise ausgeben. Sensoren, deren Daten für diese Arbeit von besonderem Interesse sind, wurden entsprechend hervorgehoben (fette Buchstaben). Darüber hinaus wird ebenfalls angegeben, ob die von den einzelnen Sensoren gemessene Energie aktiv in die Szene ausgesendet wird, oder ob die Sensoren passiv arbeiten, d.h. ausschließlich die vorhandene Energie messen. Insbesondere bei der Verwendung von aktiven Sensoren sollte stets berücksichtigt werden, mehrere dieser Sensoren in der Regel nicht ohne weiteres parallel betrieben werden können, da sie durch die Aussendung von Energie die Empfangsenergien der anderen Sensoren beeinflussen können. Offensichtlich ist dies bei Sensoren gleichen Typs, da sie dieselben Frequenzen für die Aussendung nutzen. Schwieriger ist die Identifikation von Interferenzen zwischen Sensoren unterschiedlichen Typs, da sich ggf. nur in Spezialfällen Fehler ergeben, die bei fehlender Berücksichtigung entscheidenden Einfluss in die Umgebungserfassung haben können.

	Datenformat	Sensoren	Energiezuführung	Übliche Aufnahme- frequenz
Skalare Zustands- werte	Beschleunigung	Accelerometer (IMU)	Passiv	~10kHz
	Entfernung	Hodometer	Passiv	geschwindigkeits- proportional
	Winkelgeschwindigkeit	Gyroskop (IMU)	Passiv	~10kHz
	Position (3D)	GNSS-Empfänger	Passiv	~1kHz
	Relativgeschwindigkeit	Radarsensor (1D)	Aktiv	~20Hz
	Richtung/Orientierung	Magnetometer	Passiv	~100Hz
Tiefen- informationen	Skalarer Tiefenwert	Laserdistanzmesser Ultraschallsensor Radarsensor (1D)	Aktiv Aktiv Aktiv	n.A.. ~100Hz ~20Hz
	<b>Tiefenvektor (1D)</b>	<b>Laserscanner (2D)</b> <b>Radarsensor (2D)</b>	<b>Aktiv</b> <b>Aktiv</b>	<b>~25Hz</b> <b>~20Hz</b>
	<b>Tiefenkarte (2D)</b>	<b>ToF-Kamera</b> <b>Strukturlichtkamera</b> <b>Stereokamera</b>	<b>Aktiv</b> <b>Aktiv</b> <b>Passiv</b>	<b>~50Hz</b> <b>~25Hz</b> <b>~10-100Hz</b>
	Punktwolke (3D)	3D-Laserscanner	Aktiv	<b>1/15s – 1/30min</b>
Projiziertes Bild	<b>RGB-Farbbild</b>	<b>Monooptische Kamera</b> <b>RGB-D Kamera</b> <b>Stereokamera</b>	<b>Passiv</b> <b>Aktiv</b> <b>Passiv</b>	<b>~25-100Hz</b> <b>~15-30Hz</b> <b>~10-100Hz</b>
	<b>Graustufenbild</b>	<b>Monooptische Kamera</b> <b>RGB-D Kamera</b> <b>Stereokamera</b>	<b>Passiv</b> <b>Aktiv</b> <b>Passiv</b>	<b>~25-100Hz</b> <b>~15-30Hz</b> <b>~10-100Hz</b>
	<b>Falschfarbbild</b>	<b>Infrarotkamera</b>	<b>Aktiv/Passiv</b>	<b>~25-100Hz</b>
Sonstiges	Geschwindigkeiten (1D)	Radarsensor (2D)	Aktiv	~20Hz

TABELLE 2.1: Auflistung unterschiedlicher Sensortypen sortiert nach ihrem Sensordatenformat und ergänzt durch die Wahrnehmungsrichtung und die Herkunft der zu messenden Energie.

Für die semantische Umgebungserfassung, die im Fokus dieser Arbeit steht, sind exterozeptive Sensoren erforderlich, die berührungslos Informationen ihrer Umgebung erfassen können. Bevorzugt werden Laserscanner, Radar- und Ultraschallsensoren verwendet, da sie schnell und kontaktlos Abstände zu Hindernissen und Objekten bestimmen können. Ultraschallsensoren werden aufgrund ihrer begrenzten Auflösung und Reichweite hauptsächlich zur Hinderniserkennung und Kollisionsvermeidung verwendet. Es gibt aber auch Untersuchungen zur fortgeschrittenen Umgebungserfassung und -modellierung durch die parallele Verwendung mehrerer Ultraschallsensoren [36, 37].

Laserscanner oder LiDAR-Sensoren im Allgemeinen bieten im Gegensatz dazu bereits alleine eine hohe Auflösung kombiniert mit einer ebenfalls hohen Genauigkeit, und das noch in Arbeitsradien von mehreren 10 Metern [27]. Aus diesem Grund existiert eine Vielzahl an Ansätzen zur Umgebungserfassung und -modellierung auf Basis von LiDAR-Sensoren. Ein Nachteil von aktuellen LiDAR-Sensoren sind die, im Vergleich zu den zuvor genannten Sensoren, hohen Anschaffungskosten und die Mechanik des rotierenden Sensorkopfs. Alle aktuell erhältlichen Laserscanner, mit einer Aufnahmedimension größer als eins, besitzen einen mechanisch rotierenden Sensorkopf, worin sich entweder die Laserdiode(n) selbst befinden (patentiert durch die Firma Velodyne LiDAR [38]) oder ein Spiegel befindet, der den/die Laserstrahl(en) umlenkt. Modelle ohne rotierende Elemente, sogenannte Solid-State-3D-Flash-LiDAR Systeme sind derzeit für den Automotive-Sektor angekündigt [39, 40], sind aber derzeit noch nicht regulär aus Serienproduktion erhältlich.

In vielen Anwendungen reichen reine Tiefeninformationen nicht aus, und müssen mit Bildern von Kameras kombiniert werden. Durch Techniken wie Stereobildverarbeitung, Lichtlaufzeitverfahren (time-of-flight ToF) oder Lichtprojektionssensoren (Kinect) stehen bereits Kamerasysteme zur Verfügung, die gleichzeitig Farb- und Tiefeninformationen aufnehmen können. Seitdem diese Systeme Einzug auf dem Entertainment-Sektor erhalten haben (PS4: Stereokamera [41], Microsoft Kinect v1 (Lichtprojektion) [42], Kinect v2 (ToF) [43]), sind sie auch kostengünstig erhältlich und erfreuen sich dadurch einer hohen Beliebtheit im Forschungsbereich der mobilen Robotik und Lokalisierung [44, 45, 46]. Werden Tiefeninformationen gleichzeitig zu Bildinformationen aufgenommen, oder werden sie, wie im Fall der Stereobildverarbeitung, direkt aus diesen generiert, erleichtert dies die weiteren Aufgaben erheblich im Vergleich zu einer aufwändigen Szenenrekonstruktion mittels monooptischer Kamera und einem Structure-from-Motion-Verfahren. Damit stehen mobilen Robotern eine Vielzahl an Daten zur Verfügung, die allerdings auch durch entsprechend komplexe Verfahren verarbeitet werden müssen, um aus ihnen die relevanten Umgebungsinformationen für die jeweilige Aufgabe extrahieren zu können.

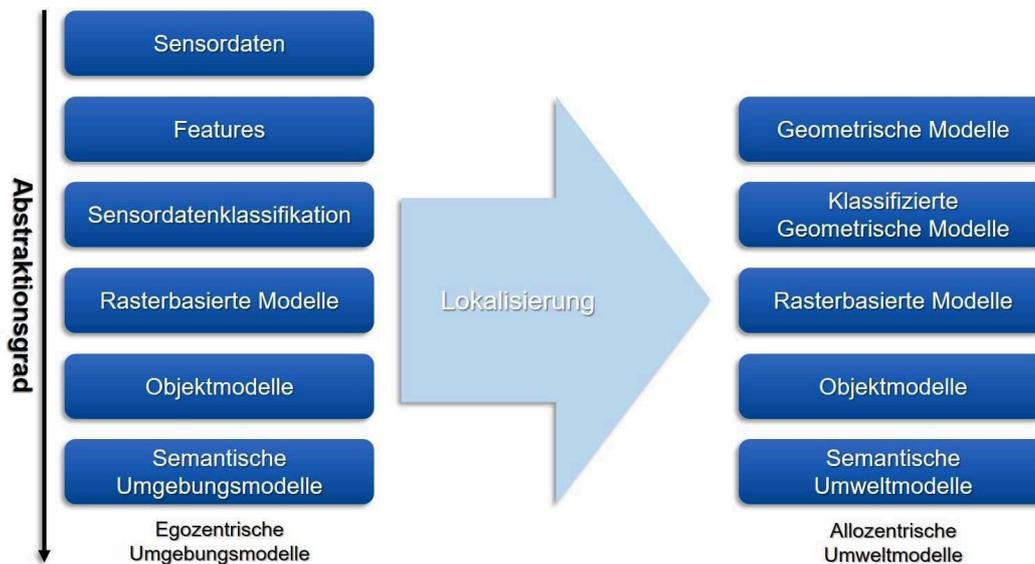


ABBILDUNG 2.8: Übersicht verschiedener Methoden zur Umgebungserfassung und Umweltdellierung

## 2.2 Umgebungserfassung

Im Hinblick auf die Lokalisierung gibt es im Bereich der Umgebungserfassung und Modellierung viele, verschiedene Ansätze, um Umweltinformationen aus den verschiedenen Sensordaten zu generieren und in Modellen zu verwalten. Abbildung 2.8 gibt einen Überblick der häufigsten Arten der Umgebungsmodellierung. Sie werden hier in erster Linie anhand ihres Abstraktionsgrads eingeordnet und anschließend noch einmal bezüglich ihres Bezugssystems, d.h. egozentrisch oder allozentrisch, unterschieden. Der Abstraktionsgrad reicht von Messwerten im Sensorraum, also reinen Sensordaten, bis hin zu einer abstrakten und semantischen Beschreibung der Umgebung.

### 2.2.1 Einzelaufnahmen von Sensoren

Die einfachste Variante eines temporären und egozentrischen Umgebungsmodells ist die direkte Verwendung von Tiefendaten aus entsprechenden Sensoren wie Ultraschallsensoren, Laserscanner, RGB-D-Kamera, etc. um etwa Hindernisse zu erkennen und darauf zu reagieren. Im Gegensatz zu anspruchsvolleren Verfahren, wird hier jeder Scan unabhängig von vorherigen Messungen ausgewertet, sodass keine zeitlichen

Bezüge erstellt werden können. Diese Art der Umgebungserfassung wird im Automobilbereich beispielsweise bei sogenannten Einparkhilfen angewendet [28].

## 2.2.2 Spatial Mapping

Werden die Einzelaufnahmen über die Zeit akkumuliert, erhält man eine dichte Punktwolke (ggf. augmentiert mit Farbinformationen), was als räumliche Karte oder Spatial Mapping bezeichnet werden kann [47]. Zur Erzeugung von solch detaillierten Umgebungsmodellen existieren auf dem Markt sogenannte Mobile Mapping Systeme. Diese werden entweder auf Fahrzeugen montiert, wie etwa Google es zur Aufnahme der Daten für die Dienste „Google StreetView“ und „Google Earth“ [48] verwendet (z.B. [49]), sind in einem Rucksack verbaut (z.B. [50]) oder werden als Handheld-Geräte (z.B. [25]) angeboten. Alle derartigen Mobile Mapping Systeme haben gemeinsam, dass sie die Sensordaten nur aufzeichnen und, aufgrund der hohen Datendichte, erst nachträglich über spezielle Dienste in räumliche Karten umwandeln. Weitere Dienste wie Pix4D [51] haben sich darauf spezialisiert, aus sich überlagernden Luftbildern Geländemodelle zu erzeugen. Die Herkunft der Bilddaten ist hier nicht auf ein spezielles System beschränkt. In der Regel werden hier GPS-gestützte Drohnen verwendet, die nicht nur die gemessene GPS-Position der Drohne zum Zeitpunkt der Aufnahme bestimmen, sondern zudem auch vordefinierte Punkte autonom per GPS anfliegen und dort die Aufnahme auflösen können. Anwendungsbereiche sind laut [50] u.a. die BIM-Kartierung von Gebäuden oder die 3D-Erfassung von Katastrophengebieten bzw. ist nach [51] darüber hinaus auch eine Ernteanalyse in der Landwirtschaft oder eine Verwendung im Immobilienmarkt möglich.

Auch zur Lokalisierung in der mobilen Robotik, werden diese räumlichen Umgebungsmodelle verwendet. Das Lokalisierungsverfahren wird dann als Scan-to-Scan-Matching bezeichnet, wie beispielsweise in [52, 53, 54, 55]. Zur Organisation solch großer Punktwolken werden Verfahren aus der Computergraphik genutzt, die große Szenen in topologische Datenstrukturen unterteilen, wie z.B. Octrees in [56]. Für die simultane Kartenerzeugung bei SLAM, werden ebenfalls oft Punktwolken verwendet, da sie anwendungsunabhängig eingesetzt werden können. Beispielsweise das SLAM Verfahren „Google Cartographer“ [57] baut eine Karte aus 2D-Laserscans auf, um eine Lokalisierung darin zu ermöglichen.

Die Daten werden hier in sogenannte Submaps organisiert, die jeweils aus mehreren gematchten Einzelscans bestehen. Ein Loop-Closing wird auf dieser Sub-Map-Ebene mittels Scan-to-Submap-Matching ermöglicht. Dies erlaubt eine Optimierung bei gefundenen Loops durchzuführen, die die akkumulierte Fehler des SLAM-Verfahrens auflöst und korrigiert. Die Verwaltung großer Punktwolken ist allerdings äußerst speicherintensiv und benötigt besonders leistungsstarke Prozessoren<sup>6</sup>.

Die Augmented Reality Brille „Hololense“ von Microsoft [58] nutzt mehrere Kameras und Strukturlichtprojektoren, um die Umgebung zu erfassen und ein geometrisches Modell zu erzeugen, das anschließend dazu genutzt wird, den realen Raum mit simulierten Komponenten zu überlagern. Die dort angewandte Technik wird von Microsoft zwar selbst auch als Spatial Mapping bezeichnet, beinhaltet aber durch eine zusätzliche Triangulierung des Raumes bereits ein höheres Informationsniveau als eine einfache, akkumulierte Punktwolke, wenngleich das Modellierungsdetail auf diese Weise reduziert wird. Dafür ist diese Technik online nutzbar und benötigt keine offline Datenverarbeitung zur Erzeugung des Umgebungsmodells, wie etwa [49, 50, 25, 51] oder besonders leistungsstarke Prozessoren, wie [57].

### 2.2.3 Features

Eine vielfach verwendete Art der Umweltbeschreibung ist die Detektion und Beschreibung von Merkmalen in den Sensordaten und die Verwaltung dieser sogenannten Features in Datenbanken oder, nach der Rückprojektion in den dreidimensionalen-Raum, in räumlichen Datenstrukturen [59].

Um Sensordaten für ein Umgebungsmodell nutzen zu können, müssen allerdings nicht nur Sensordaten aufgenommen werden. Es müssen auch geeignete Verarbeitungsalgorithmen entwickelt werden, die es ermöglichen, die Daten geeignet nutzen zu können. In der Bildverarbeitung wurden zu diesem Zweck bereits eine Vielzahl von Merkmalsextraktoren und -deskriptoren entwickelt. Die große Anzahl dieser Entwicklungen liegt unter anderem wohl daran, dass Kameras in dieser Zeit durch eine Produktion für den Massenmarkt immer günstiger und durch den erhöhten Wettbewerb gleichzeitig auch hochwertiger geworden sind. Junichi Nakamura gibt in [60] einen Überblick über die Geschichte und

---

<sup>6</sup>Hess et al. verwenden in [57] einen Intel Xeon E5-1650 Prozessor mit 3,2 GHz.

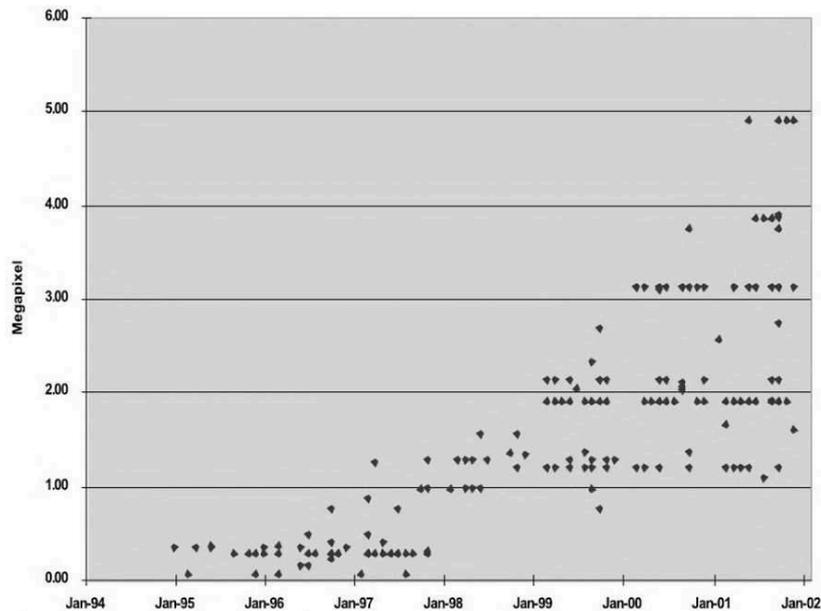


ABBILDUNG 2.9: Entwicklung der Megapixelanzahl von digitalen Fotoapparaten um die letzte Jahrtausendwende (Bildquelle: [60]).

Entwicklung der digitalen Kamera (z.B. Abbildung 2.9). Zudem wurde, aufbauend auf den Detektoren und Deskriptoren eine Vielzahl an Anwendungen entwickelt, die eine fortführende Interpretation und Nutzung der Sensordaten erlauben. Beispiele sind z.B. die Gesichtserkennung [61], Verkehrszeichendetektion [62] oder die 3D-Rekonstruktion aus Kamerabildern [59]. So, wie sich die einzelnen Anwendungen für 2D-Features unterscheiden, so unterscheiden sich auch die Feature-Detektoren, d.h. sie wurden auf bestimmte Anwendungsfelder und deren Anforderungen hin konzipiert und entwickelt. So eignen sich beispielsweise einige Features besser zum Tracking von Strukturen über mehrere Aufnahmen und andere sind besser geeignet für die Erkennung von größeren markanten Bereichen in Bildern. In [63] geben Hassaballah, et. al. einen detaillierten Überblick der zur Zeit am häufigsten benutzten Featuredetektoren und Deskriptoren. Allgemein hat sich eine Unterteilung in Kanten-, Ecken- und Blob-Detektoren etabliert. Kanten- und Ecken-Features beschreiben genau das, was ihr Name vermuten lässt. Zu den verbreitetsten Kandedektoren gehört der Canny-Detektor [64]. Dieser bestimmt nach einer

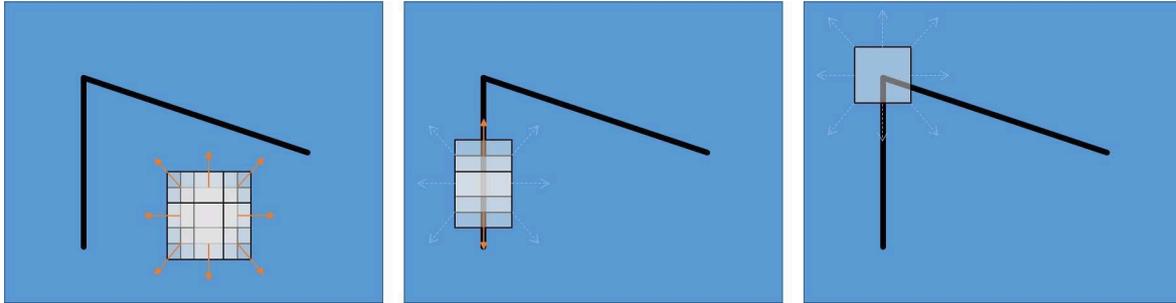


ABBILDUNG 2.10: Basis-Idee des Harris-Detektors: Im Gegensatz zu „flachen“ Regionen und Kanten, können Ecken aufgrund ihrer Intensitätsänderung in alle Richtungen genau lokalisiert werden.

partiellen Ableitung des Bildes in  $x$ - und  $y$ -Richtung den Gradientenbetrag des Bildes

$$|\nabla I(x, y)| = \left| \begin{pmatrix} \partial_x I \\ \partial_y I \end{pmatrix} \right| = \left| \begin{pmatrix} I_x \\ I_y \end{pmatrix} \right| = \sqrt{(I_x)^2 + (I_y)^2}.$$

zur Bestimmung der Kantenstärke  $e_s = \nabla I$ . Durch die Rotationsinvarianz des Bildgradienten können auf diese Weise Kanten aller Orientierungen bestimmt werden sowie die Orientierung der Kante  $e_o = \arctan \frac{f_x}{f_y}$  selbst.

Ein großer Nachteil bei der Verwendung von Kanten ist der, dass entlang der Kante keine Intensitätsänderungen auftreten, sodass sie keine eindeutige Position im Bild aufweisen. Ecken hingegen, können aufgrund signifikanter Änderungen der Intensität in zwei Richtungen genau in ihrer Position bestimmt werden. Der bekannteste Eckendetektor ist der Harris-Detektor [65]. Er basiert auf der Strukturmatrix bzw. dem Strukturtensor:

$$T(x) = G_\sigma * \nabla I \nabla I^T = \int G_\sigma(x - x') \begin{pmatrix} (I_x(x'))^2 & f_x(x') f_y(x') \\ f_x(x') f_y(x') & (I_y(x'))^2 \end{pmatrix} dx',$$

wobei  $x, x' \in \mathbb{R}^2$  und  $G_\sigma$  ein Glättungskern der Breite  $\sigma$  ist.

Ein Eckendetektor, der mit Hilfe von Techniken aus dem maschinellen Lernen eine besonders schnelle Detektion erlaubt ist der FAST-Detektor [66].

Ein großer Nachteil bei der Verwendung von Ecken als Merkmalen ist,

dass sie nicht skalierungsinvariant sind. Schmid et.al. geben in [67] z.B. an, dass die Wiederholbarkeit einer Detektion mit dem Harris-Detektor ab einer Skalierung um den Faktor 2 bereits bei nur noch ca. 20% liegt. Für viele Anwendungsgebiete sind einfache Ecken daher nicht ausreichend.

Daher wurden Feature-Detektoren entwickelt, die das Problem der fehlenden Skalierungsinvarianz damit angehen, parallel auf mehreren Skalierungsstufen nach Merkmalen zu suchen. Diese Art wird als Blob-Features bezeichnet. Unter diesen Begriff fallen nahezu alle weitere bekannte Featuredetektoren, wie etwa SIFT [68] und die Weiterentwicklung SURF [69]. Diese Detektoren basieren meist auf dem sogenannten Laplacian-of-Gaussian (LoG), was einer Kombination einer Gaussfilterung und mit dem Laplace-Operator darstellt. Der Laplace-Operator  $\Delta$  entspricht dem Skalarprodukt des Bildgradienten mit sich selbst, also:

$$\Delta I(x, y) = \nabla^2 I(x, y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} = I_{xx} + I_{yy}.$$

Da die zweite Ableitung eines Bildes besonders anfällig ist für hochfrequentes Bildrauschen, wird das Bild zunächst mit einem Gaussfilter geglättet. Die diskrete Approximation des Laplace-Operators kann, wie der Gaussfilter selbst, durch die Faltung des Bildes mit einer diskreten Filtermaske durchgeführt werden, und weil die Faltungsoperation assoziativ ist kann der Gausskern zunächst nur mit dem Laplace-Filterkern gefaltet werden. Auf diese Weise muss das Bild nur einmal mit dem kombinierten Filter gefaltet werden.

Eine Beschleunigung der Ableitungsberechnung kann mittels Approximation über Difference-of-Gaussian (DoG) erreicht werden, wie in [68] oder alternativ durch die Verwendung der Hesse-Matrix wie in [69]. Für weitere Details sei hier auf [70] verwiesen.

Für die Umweltbeschreibung mittels Features ist neben den Detektionsalgorithmen auch eine Möglichkeit zur anschließenden Beschreibungen notwendig. Blob-Features besitzen daher zumeist ihre eigenen Feature-Deskriptoren, die eine möglichst invariante Beschreibung des Features bezüglich, Rotation, Skalierung, Beleuchtung und affine Transformationen aufweisen. Idealerweise werden dazu Merkmalsvektoren genutzt, die Informationen enthalten, die während des Detektionsprozesses bereits vollständig, oder zumindest zum Teil, berechnet wurden. Der SIFT-Deskriptor z.B. ist ein Vektor, der 128 Elemente zur Beschreibung

eines Features verwendet. Anhand der darin gespeicherten Informationen ist es möglich Features des einen Bildes auch in anderen Bildern, die dieselbe Szene aus leicht veränderter Perspektive zeigen, wiederzufinden. Das Wiederfinden und Matching von Features in zwei Bildern aus unterschiedlichen Aufnahmeperspektiven ist ebenfalls Gegenstand vieler Untersuchungen (z.B. [71, 72, 73]). Was für kleine Kamerabewegungen funktioniert, muss für eine Umweltmodellierung die u.a. auch zur Lokalisierung verwendet wird, auch für beliebig große Bewegungen funktionieren. Dieses Problem haben Verfahren, die lediglich Features und ihre Deskriptoren zur Lokalisierung in Datenbanken verwalten (z.B.[74, 75]).

#### 2.2.4 Hinderniserkennung

In Anwendungsfeldern der autonomen, mobilen Robotik, wo es neben einer zuverlässigen Lokalisierung ebenfalls wichtig ist, sich in der Umgebung zurechtzufinden und auf dynamische Gegebenheiten zu reagieren, reicht eine Menge von Scanpunkten ebenso wenig aus, wie eine Umgebungsbeschreibung im Sensorraum. Um erfolgreich erkennen zu können, ob ein Weg vorhanden und befahrbar ist, müssen die Sensordaten auf eine bestimmte und zweckmäßige Art und Weise interpretiert werden. In vielen Fällen läuft diese Interpretation auf eine Hinderniserkennung hinaus. Auf Hindernisse kann dann reagiert werden, wenn sie einen gewählten Weg blockieren oder aus Sicherheitsgründen eine Weiterfahrt verbieten. Diese Herangehensweise ist insbesondere für Anwendungen autonomer Fahrzeuge „auf freiem Feld“ interessant, d.h. wo vorab kein exakter Weg (off-road) vorgeben ist und große Bereiche frei befahrbar sind (outdoor), wie allgemein in [76, 77, 78, 79] behandelt oder speziell für die DARPA Grand Challenge [80], wie in [81] und [82] beschrieben. In [83] wird die Hinderniserkennung für einen autonomen Rasenmäher genutzt. Zur Lokalisierung werden die erkannten Hindernisse nicht verwendet sondern stattdessen auf satellitengestützte Verfahren zurückgegriffen. Auch werden die Hindernisse ausschließlich dynamisch erkannt und meist nicht in eine Karte übertragen.

## 2.2.5 Occupancy Grids

In [84] wird die Umgebung eines mit einer Stereokamera ausgerüsteten Fahrzeugs in Occupancy Grids modelliert und zusammen mit einem Partikelfilter getrackt. Occupancy Grids unterteilen die Umgebung in ein uniformes Raster aus Zellen um diese anschließend als besetzt (Hindernis erkannt) oder frei zu kennzeichnen. Diese Art der Umgebungsmodellierung ist durch die Verwendung uniformer und diskreter Raster verhältnismäßig schnell und einfach umzusetzen. Laut Danescu et al. in [84] sind Occupancy Grids ideal für die Kartierung von „unstrukturierten Umgebungen“ geeignet. Die Umgebung und deren Bewegung/Geschwindigkeit wird hier relativ zum Fahrzeug in einem festen Gitter modelliert, d.h. das Occupancy Grid bewegt sich mit dem Fahrzeug mit und wird für eine relative Umgebungsmodellierung aus Fahrzeugsicht genutzt. Diese Art der Verwendung von Occupancy Grids ist üblich für schnell fahrende autonome Fahrzeuge zur Überwachung der Fahrzeugumgebung, und wird äquivalent z.B. auch in [85, 86] verwendet.

Für mobile Roboter in räumlich begrenzten Aktionsgebieten werden Occupancy Grids für globale Umgebungsmodelle verwendet (z.B. [87, 88, 89]), und sind nach Kaufmann et al. in [46] eine räumliche Repräsentation der Umgebung. Sie haben hier meist zwei Funktionen. Zum einen werden sie als Karte zur Lokalisierung verwendet und zum anderen als Grundlage zur autonomen Wegplanung. Ein Beispiel hierfür ist [90], auch wenn Kakiuchi et al. selbst nicht den Begriff des Occupancy Grids verwenden. In [46] werden Occupancy Grids sowohl als Umgebungskarte als auch zur Planung der autonomen Exploration unbekannter Gebiete verwendet. Zur Bestimmung der besten Explorationsstrategie wird für alle Grid-Zellen paarweise unabhängige Zufallsvariablen gespeichert, die beinhalten, wie wahrscheinlich die jeweilige Grid-Zelle besetzt ist. Noch nicht oder nur unzureichend erfasste Bereiche werden auf Basis dieser Verteilung von Wahrscheinlichkeiten ermittelt und können gezielt angefahren werden, um die Lücken in der Karte zu schließen. Ähnlich gehen Carillo et al. in [91] vor, nur werden hier die Wahrscheinlichkeiten separat in einem Graph verwaltet und das Occupancy Grid lediglich als Umgebungskarte genutzt.

Die einfache Speicherung von Umgebungskarten in diskreten 2D-Grid-Zellen kann aber auch Nachteile haben, insbesondere in Verbindung mit SLAM-Verfahren, die mit der Zeit Fehler akkumulieren. Mittels „loop-closing“, also gezieltem Betreten bereits besuchter Bereiche, wird versucht diesen als „drift“ bezeichneten Fehler zu detektieren und in der Karte zu korrigieren. Bei der Verwendung von festen Gitterzellen kann diese Korrektur gerade bei großen Modellen sehr rechenintensiv werden. Schmuck et al. schlagen daher in [92] zur Lösung des Problems vor, kein einzelnes, uniformes Grid, sondern eine Reihe von Sub-Maps zu verwenden, die über einen topologischen Graphen verwaltet werden. Dieses Vorgehen erlaubt schnellere Änderungen der Karte und beschleunigt ebenfalls die Suche von Wegen in großen Umgebungen. Das Vorgehen ähnelt dem Verfahren von Hess et al. in [57] (Google Cartographer), das weiter oben bereits behandelt wurde.

### 2.2.6 Klassifiziertes Spatial Mapping

Ziel der Umgebungsmodellierung mittels Occupancy Grids oder Spatial Mapping ist in erster Linie ein räumliches Modell der Umgebung zu erzeugen und es ggf. dazu zu nutzen um Wege zu planen. Sobald aber eine Interaktion mit der Umgebung gefordert ist, reicht eine räumliche Sicht alleine nicht aus. Die räumlichen Informationen müssen interpretiert, d.h. segmentiert und klassifiziert werden, um eine Interaktion zu ermöglichen. Eine ausführliche Übersicht über die am häufigsten genutzten Ansätze zur 3D-Segmentierung geben Nguyen und Le in [93]. Sie identifizieren fünf verschiedene Vorgehensweisen:

- *Kantenbasierte Verfahren:* Hier werden, ähnlich wie bei den 2D-Kantendetektoren in Bildern, Grenzen in den räumlichen Daten gesucht anhand derer einzelne Segmente von einander unterschieden werden können (z.B. [94, 95]). Nachteile dieser Verfahren sind ihre hohen Empfindlichkeiten gegenüber Rauschen und bei inhomogener Dichte in den räumlichen Informationen.
- *Regionsbasierte Verfahren:* Diese Verfahren fassen Regionen anhand von Nachbarschaftsinformationen zusammen oder teilen einzelne Regionen auf, wenn zu hohe Unterschiede existieren. Man unterscheidet hier das Vorgehen, das ausgehend von einer einzigen Region diese immer weiter unterteilt (top-down) [96] und jenes, das

mit einer Verteilung von initialen Punkten (seeds) startet und, von diesen ausgehend, die Regionen wachsen lässt (bottom-up) [97, 98].

- *Attributbasierte Verfahren:* In diesem zweistufigen Verfahren werden die räumlichen Daten zunächst attribuiert und anschließend entsprechend ihrer Attribute zu Segmenten zusammengefasst. Für die Attribuierung können prinzipiell beliebige Eigenschaften gewählt werden. Häufig verwendet werden Oberflächennormalen (z.B. in [99, 100]).
- *Modellbasierte Verfahren:* Modellbasierte Verfahren verwenden meist geometrische Primitive zur Gruppierung von räumlichen Informationen. Das sogenannte „model-fitting“ wird meistens mittels einem RANSAC-Algorithmus durchgeführt. So auch in [101, 102, 103] und [104].
- *Graphbasierte Verfahren:* Das letzte Verfahren nutzt die Graphentheorie und das sogenannte Graph-Cut-Verfahren zur Segmentierung von 3D-Informationen. Dazu wird beispielsweise jeder Vertex als Knoten eines Graphen angesehen, der mit seinen Nachbarn über Kanten verknüpft ist (z.B. in [105, 106, 107])

Für eine tiefergehende Beschreibung und Diskussion der unterschiedlichen Ansätze sei an dieser Stelle auf [93] verwiesen.

Durch die Segmentierung selbst wird der Informationsgehalt des Umgebungsmodell allerdings noch nicht ausreichend erhöht, um eine Interaktion zu erlauben. Erst die Klassifikation der Segmente erlaubt es einem Roboter oder einem autonomen Fahrzeug höherwertige Funktionen auszuüben. Zunächst hilft eine einfache Unterteilung in statische und dynamische Strukturen, sodass beim Abfahren eines geplanten Wegs abgeschätzt werden kann, ob sich Kollisionen ergeben könnten, um dann vorab entsprechend reagieren zu können. In [108] verwenden Schreier et al. für diese Unterteilung zwei separate Karten für dynamische und statische Komponenten. Zunächst nutzen sie für diesen Zweck, ähnlich wie bereits Danescu et al. in [84] ein temporäres Occupancy Grid, um eine Darstellung der Umgebung zu erhalten, die unabhängig vom eingesetzten Sensor-Setup<sup>7</sup> ist. Darüber hinaus dient hier das Occupancy

---

<sup>7</sup>Als Sensor-Setup wird hier die Art und räumliche Verteilung der eingesetzten Sensoren verstanden.

Grid als Bindeglied zur Synchronisierung zwischen den beiden Karten für dynamische bzw. statische Szenenelemente.

Eine weitere Unterteilung von Objekten kann auf unterschiedliche Weisen erfolgen. So schlagen Osdas und Funkhouser in [109] z.B. eine Unterscheidung von Objekten nach ihrer Form vor. Sie bauen dafür eine Datenbank von Objektformen auf und argumentieren, dass Forms Signaturen, nach erfolgter Segmentierung, schnell verglichen werden können und gleichzeitig eine große Diskrepanz zu unähnlichen Formen aufweisen. Allerdings ist diese Art der Klassifikation sehr anfällig für partielle Verdeckungen von Objekten. Deskriptoren für Besonderheiten in Geometrie und Intensität werden in [110] genutzt, um Merkmale für einzelne Segmente zu finden und um sie daraufhin zu unterscheiden. So können beispielsweise anhand von Oberflächennormalen Gebäude leicht von Vegetation wie Büschen unterschieden, oder Bäume anhand ihrer charakteristischen, höhenabhängigen Ausdehnung identifiziert werden. Dieser merkmalsbasierte Ansatz ist nach Himmelsbach und Luettel in [111] bestens für die Aufgabe der Objekterkennung geeignet, hat aber den Nachteil, dass die steigende Anzahl an verschiedener, unterscheidbarer Klassen nur auf Kosten einer gesteigerten Dimensionalität der Merkmalsvektoren möglich ist. Dieses Problem ist bereits aus der Merkmalsbeschreibung von 2D-Features in der Bildverarbeitung bekannt und erhöht sich noch einmal eklatant in der räumlichen Beschreibung von Merkmalen. Himmelsbach und Luettel sprechen von einer Spanne von 200 bis 1000 Featurewerten pro 3D-Punkt und schlagen daher die Nutzung von Methoden des überwachten Lernens vor, um die Klassifizierung in diesen hochdimensionierten Ereignisräumen durchzuführen. In [111] verwenden sie dafür einen Support Vector Machine (SVM) Klassifizierer, der eine Klassifizierung anhand von Histogrammen der einzelnen Merkmale in Realzeit erlaubt. SVM-Klassifizierer verwenden Xu et al. ebenfalls in [112], allerdings nicht auf Basis von Merkmalshistogrammen, sondern auf sogenannten Morphologischen Features. Dafür werden vorsegmentierte 3D-Cluster in mehrere horizontale Schichten unterteilt, und anschließend jeweils senkrecht in den 2D-Raum projiziert werden. Diese 2D-Scheiben werden dann anhand ihrer Form (Morphologie) in Merkmalsvektoren beschrieben. Anschließend erfolgen dann Klassifikation der Scheiben per SVM und die Fusion zusammengehöriger Teile zu ganzen Objekten. Mit erfolgreicher Segmentierung und Klassifikation liegt das räumliche Umgebungsmodell nun, in jeweilige Bedeutungsklassen

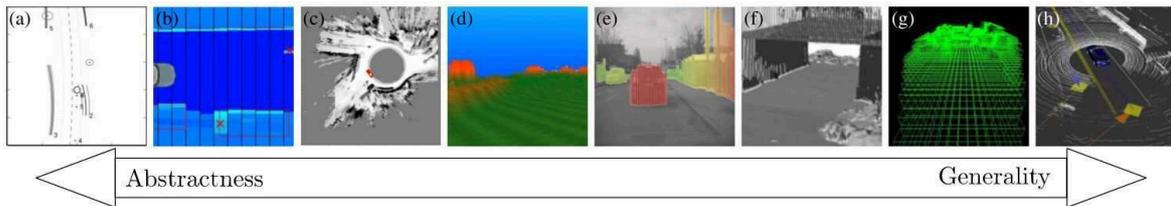


ABBILDUNG 2.11: Kategorisierung von räumlichen Umgebungsmodellen nach [108] in: Featurekarte (a), Intervallkarte (b), 2D-Grid-Karte (c), Steigungskarte (d), Stixel-Welt (e), Multi-Level-Oberflächenkarte (f), Voxel-Grid (g), Sensorrohdaten (h).  
Bildquelle: [108]

unterteilt, vor.

Schreier et al. teilen in [108] die bereits betrachteten, unterschiedlichen Umgebungsmodelle anhand der Repräsentation ihrer Informationen auf einer Skala von „unbestimmt“ (generality) bis zu „abstrakt“ (abstractness) auf (Abbildung 2.11). Was Schreier et al. hier mit Abstraktheit beschreiben, ist der Übergang von reinen Sensordaten bis hin zu abgeleiteten Strukturen, die aus den Sensordaten berechnet und beschrieben werden können. In ihrem Fall stellen Features, die abstrakteste Form der Darstellung dar. Im Zusammenhang mit der Einteilung in Abbildung 2.8 kann man die Abstraktheit als weitere Dimension betrachtet, die unabhängig zum hier verwendeten Informationsniveau und Modellierungsdetail ist.

## 2.2.7 Vom Spatial Mapping zum semantischen Umgebungsmodell

Golovinskiy et al. haben bereits 2009 erkannt welche Wichtigkeit Stadtmodelle mit semantisch ausgezeichneten Objekten (z.B. Autos, Straßenlaternen, Ampelanlagen, etc.) für eine Vielzahl von Anwendungen wie Stadtplanung, Katastrophenschutz, Tourismus, Dokumentation von Kulturgütern u.v.m. darstellen [113]. Ramiya et al. bauen in [114] darauf auf und betonen, dass eine automatisierte Erzeugung dieser Modelle noch immer eine Herausforderung darstellen. Wie schon Golovinskiy et al. geht es ihnen um eine flächendeckende Klassifizierung von räumlichen Modellen im Kontext der Forschungsgebiete um Photogrammetrie, Fernerkundung und Geoinformatik. Das Ergebnis ihrer Arbeiten sind also

großflächige und klassifizierte räumliche Umgebungsmodelle. Rocha et al. gehen bereits einen Schritt weiter und haben erkannt, dass ein semantisches Modell Grundlage für viele verschiedene Anwendungen und Smart-Devices sein kann und beschreiben in [115] die Vorteile, die eine semantische Umweltmodellierung im Kontext der sogenannten Smart Cities bringen kann.

Auch im Zusammenhang mit der Lokalisierung und simultanen Kartenerzeugung (SLAM) wurde die semantische Umweltmodellierung bereits berücksichtigt. So werden in [116] und [117] zwei Arten der semantischen Umweltmodellierung für eine erfolgreiche Lokalisierung und Kartierung aufgezeigt. Wie in den zuvor genannten Publikationen steht auch hier die Anwendung (hier. Lokalisierung) im Vordergrund.

Abgrenzend dazu, steht in der vorliegenden Arbeit die Erzeugung anwendungsübergreifender, semantischer Umgebungs- bzw. Weltmodelle im Vordergrund. Der Unterschied zwischen einem klassifizierten, räumlichen Umgebungsmodell und einem semantischen Umgebungsmodell besteht darin, dass den einzelnen Objekten im semantischen Modell mit einem Datenmodell der jeweiligen Objektklasse begegnet wird, so dass die jeweiligen Objekte (oder Instanzen) anhand ihrer individuellen Attribute weiter beschrieben werden können. Bei einer klassifizierten Punktwolke ist nur gespeichert, welche Messpunkte zu einem Objekt gehören, und zu welcher Objektklasse dieses Objekt gehört. Damit kann zwar auf bestimmte Objekttypen entsprechend reagiert werden, jedoch erlaubt es keine weiterführende Interaktion mit diesen Objekten oder tiefergehende Umgebungsanalysen, da eine Verknüpfung zwischen den Messungen und Instanzen von fundierten Objektklassen fehlt. Dies wird in der Literatur als das „Verankerungsproblem“ (anchoring problem) bezeichnet [118] und ist nach Elfring et al. die erste von vier Anforderungen an einen Algorithmus zur Erzeugung von Weltmodellen für das Anwendungsfeld der mobilen Robotik [119]. Die zweite Anforderung stellt die zuverlässige Attribuierung von Objekten dar, um sie unterscheidbar zu beschreiben. Als weitere Anforderungen nennen Elfring et al. die modellbasierte Objektverfolgung (model-based object tracking) sowie eine mögliche Ausführung des Ganzen in Realzeit. Eine Umsetzung wird in [119] in einem künstlichen Haushaltsszenario gezeigt, wo Objektbewegungen auf einem Küchentisch mit einer Kamera überwacht und über die Zeit getrackt werden.

Alle in diesem Kapitel untersuchten Umgebungsmodelle stehen in

Bezug zu einer spezifischen Anwendung zu deren Zweck sie erzeugt und verwaltet werden. Diese starre Bindung zu einem konkreten Anwendungsbereich verhindert allerdings auch die zusätzliche Nutzung dieser Umgebungsmodelle in weiteren Anwendungen, die eine semantische Weltbeschreibung voraussetzen (vgl. [113, 114]). Um diese Lücke zu schließen bedarf es zunächst einer erschöpfenden Herleitung und Definition der semantischen Umgebungserfassung und Modellierung. Da zum jetzigen Zeitpunkt keine exakte Definition für diese Begrifflichkeiten existieren, bzw. falls doch, sie sich bislang nicht etabliert haben, soll im nächsten Kapitel (Abschnitt 3.2) diese Aufgabe auf Basis des hier gegeben, aktuellen Stand der Technik, nachgeholt werden.

## 2.3 Lokalisierungsverfahren

Eine möglichst exakte und zuverlässige Lokalisierung ist grundlegend für viele Anwendungen der aktuellen Forschung. Insbesondere im Kontext der Automatisierung von Fahrzeugen oder Augmented und Virtual Reality, ist die Bestimmung der exakten Lage von Objekten im dreidimensionalen Raum grundlegende Voraussetzung. Im Zusammenhang mit einer semantischen Umgebungserfassung und Umweltmodellierung ist die Lokalisierung ebenfalls fest verwoben, wie im nächsten Kapitel noch gezeigt wird. Daher soll an dieser Stelle ein kurzer Überblick über aktuelle Lokalisierungsverfahren für mobile Systeme (Autos, Roboter), aber auch für Sensorik (LiDAR, Kamerasysteme) im Allgemeinen, gegeben werden.

Abbildung 2.12 gibt einen Überblick der aktuell etablierten Lokalisierungsverfahren im Bereich der mobilen Robotik, aber auch in artverwandten Anwendungsgebieten wie im Automotive-Bereich oder Welt- raumanwendungen (z.B. Rendezvous&Docking).

### 2.3.1 Kartenlose Lokalisierung

In der mobilen Robotik sind kartenlose Lokalisierungsverfahren beliebt, da sie kostengünstig und mit wenig Aufwand zu realisieren sind. In der Regel reicht hier eine gewöhnliche Kamera aus, wie in [120, 72, 121, 122, 123, 124] Darüber hinaus wurden zahlreiche Verfahren entwickelt, die parallel zu den Farbinformationen der Kamera auch Tiefeninformationen

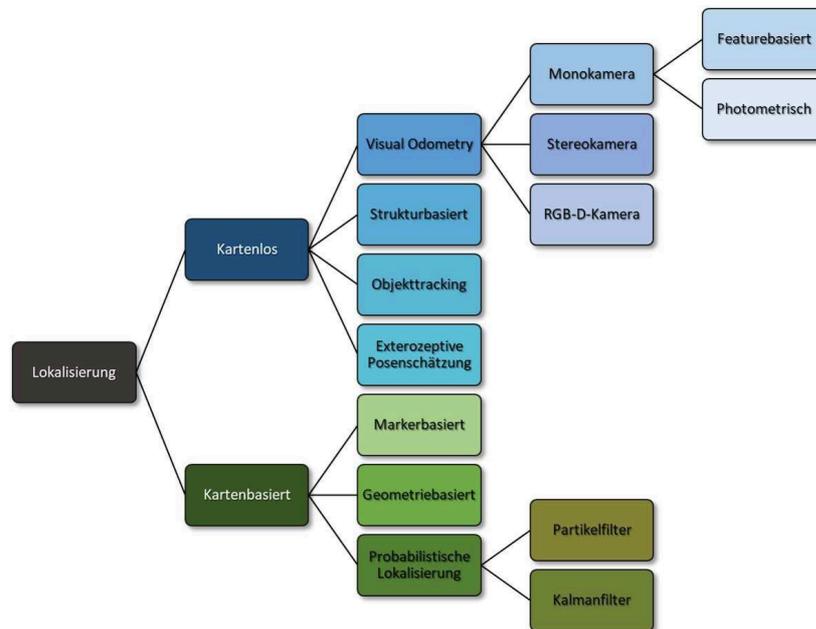


ABBILDUNG 2.12: Übersicht der etablierten Lokalisierungsverfahren im Bereich der mobilen Robotik

verwenden, um die Lokalisierungsergebnisse zu verbessern, wie [125, 126, 127, 128, 129, 130, 131]. Die einzelnen Verfahren der kartenlosen Lokalisierung lassen sich daran unterscheiden, wie die Sensordaten genutzt werden, um daraus eine (Eigen-)Bewegung ableiten zu können.

### Feature Tracking oder Visual Odometry

Der Begriff „Visual Odometry“ wurde maßgeblich von Nistér et al. in [120] als Feature-Tracking-Verfahren für Video-Sequenzen etabliert. In dem Verfahren werden zunächst Features in einer Reihe aufeinanderfolgender Bilder detektiert und anschließend Paarungen zwischen Features benachbarter Bilder gesucht. Diese sogenannten Feature-Matches werden genutzt, um die Kamerabewegung zwischen diesen beiden Aufnahmen rekonstruieren zu können. Im Grunde werden dazu dieselben Algorithmen verwendet wie beim Structure-from-Motion (SfM, siehe [59]), mit dem Unterschied, dass beim SfM die Features genutzt werden, um die betrachtete Szene zu rekonstruieren und die Kameraposen verworfen werden, und bei Visual Odometry die Kameraposen die Grundlage der Bewegungsschätzung darstellen und stattdessen die Features verworfen werden. Aufgrund der vielfältigen Anwendungsmöglichkeiten beider

Verfahren, gab es seitdem eine Vielzahl an Veröffentlichungen, die dieses Thema tiefer untersucht haben ([125, 120, 72, 121]). Ein Nachteil bei der Verwendung von monooptischen Kameras zur Bewegungsschätzung ist in erster Linie die fehlende Information zur Skalierung. So ist es, ohne die Verwendung zusätzlicher Einschränkungen oder externer Sensorik, nicht möglich eine metrische Bewegungsschätzung zu erhalten. Abhilfe für dieses Problem bietet der Einsatz von Stereokameras, wie in [126, 129] veröffentlicht, bzw. patentiert in [128]. Mit der Veröffentlichung der Microsoft Kinect, und dem Zugang zu erschwinglichen RGB-D-Kamerasystemen, wurden auch diese vermehrt dazu genutzt, das Skalierungsproblem zu bewältigen (z.B. in [127]).

### **Semi-dense Visual Odometry**

Alternativ zur Bewegungsschätzung aus Feature-Paarungen zweier aufeinanderfolgender Bilder, kann die Bewegung auch direkt, durch Nutzung photometrischer Verfahren, aus den beiden Bildern berechnet werden. Dazu wird die Bildbewegung im Ganzen berechnet und anschließend auf die Kamerabewegung übertragen. Dieses Verfahren ist einfach umzusetzen, wenn die Kamera eines mobilen Roboters nach unten auf einen planaren Boden ausgerichtet wird, wie in [132, 133, 122]. Damit ist die Tiefe der Szene (Abstand zwischen Kamera und Boden) bekannt und kann zur Lösung des Skalierungsproblems genutzt werden.

Engel et al. argumentieren in [123]: „Videos sind eine günstige Quelle für reichhaltige Informationen, aus diesem Grund sollte auch der Rechenaufwand betrieben werden, um daraus den größtmöglichen Informationsgewinn zu erlangen.“ Daher schlagen sie vor, wie bei sogenannten Semi-dense-SLAM-Verfahren, nach erfolgter Feature-basierter Bewegungsschätzung, Methoden aus der Stereobildverarbeitung zu nutzen, um dichte Tiefenkarten aus den jeweiligen Bildpaaren zu berechnen. Beim SLAM fließen diese Informationen direkt in die räumliche Karte ein (s.o.). In [123] werden die dichten Tiefenkarten dagegen genutzt, um, durch Minimierung der geometrischen und photometrischen Fehler, die initiale Bewegungsschätzung zu verbessern und die Bewegung vorab für folgende Frames zu schätzen. Dieses Verfahren wurde in [124] vereinfacht, sodass die Rechenleistung eines Smartphones zur Berechnung der Bewegung in Realzeit ausreicht. Einen Benchmark-Datensatz für diese

Art von Bewegungsschätzung inklusive Kartenerzeugung wird in [44] vorgestellt.

Werden RGB-D-Kamerasysteme, statt einfachen monokularen Kameras, genutzt, kann die aufwändige Stereobildverarbeitung ausgelassen und direkt auf die Tiefeninformationen zugegriffen werden [130, 131]. Dies erlaubt neben einer schnelleren Bewegungsschätzung, gleichzeitig, durch die exaktere Tiefenmessung, eine höhere Genauigkeit.

### **Strukturbasierte Verfahren**

Bei strukturbasierten Verfahren verzichtet man gänzlich auf die Bildinformationen zur Bewegungsschätzung. Stattdessen wird ausschließlich auf Tiefeninformationen von LiDAR-Sensoren [134, 135], Stereokameras [136] oder RGB-D-Kamerasystemen [137] gerechnet. Grundlegende Komponente bei diesen Verfahren ist der Iterative-Closest-Point-Algorithmus (ICP) [138], der dazu verwendet wird den Abstand zweier Punktwolken zu minimieren. Eine Erweiterung des klassischen ICP-Algorithmus durch die Verbindung mit einem Point-to-Plane-Ansatz wird in [135] vorgestellt.

### **Objektbasierte Lokalisierung**

Der Schritt von featurebasierten Lokalisierungsverfahren zu strukturbasierten Lösungen ist gleichzeitig ein Schritt näher an eine Räumlichkeitsbeschreibung und dadurch auch näher an einer gegenstands-basierten Umweltbeschreibung. Werden Features noch im projizierten Bildraum einer Kamera gesucht und verfolgt, befinden sich räumliche Merkmale bereits in dem Raum, indem man sich orientieren möchte. Ein weiterer Schritt ist nun die Erkennung von zusammenhängenden Objekten und die Orientierung anhand dieser. Diese objektbasierte Lokalisierung kann kartenlos erfolgen [139], ist aber auch ideal für SLAM-Verfahren geeignet [140, 6]. Eine kartenbasierte Betrachtung erfolgt weiter unten.

### **Exterozeptive Posenbestimmung**

Wo bei der objektbasierten Lokalisierung die Eigenbewegung der Kamera bezüglich einer statischen Szene berechnet wird, ist es auch möglich

die Sichtweise vollständig umzukehren. Damit werden dann Objektbewegungen relativ zur Kamera geschätzt. Diese exterozeptive Posenbestimmung kann dazu verwendet werden, einen begrenzten Aktionsraum mit einer (oder mehreren) fest installierten Kamera(s) zu überwachen, und Objektbewegungen in diesem Raum zu überwachen. Z.B. kann auf einem Roboter ein passiver Marker befestigt werden, dessen Lage von der externen Kamera erfasst und bestimmt werden kann. Aus der Augmented Reality ist diese Herangehensweise bereits bekannt [141, 142]. Dort wird mit Hilfe leicht detektierbarer Marker, die Lage von Objekten bestimmt, die anschließend mit virtuellen Objekten überlagert werden. Auch in Weltraumanwendungen werden Marker verwendet, um die Lage von externen Objekten mittels Kamera bestimmen zu können [143, 144].

### 2.3.2 Kartenbasierte Lokalisierung

Ein großer Nachteil bei kartenlosen Lokalisierungsverfahren im Allgemeinen ist die nur relative Posenbestimmung. Die absolute Pose wird hier mittels Pfadintegration aus einer initialen Pose und der Bewegungsschätzung aus mehreren Messungen aufintegriert. Das bedeutet, dass auch bei der Verwendung von Filtern und Verfahren wie Bundle Adjustment zur Absicherung der Ergebnisse, entsteht bei jeder Bewegungsschätzung ein Fehler (u.a. bestehend aus Messfehlern der Sensoren und numerischen Fehlern bei der Berechnung), der sich mit der Zeit ebenfalls aufsummiert. Eine Möglichkeit zur Verringerung des Fehlers ist das sogenannte „loop-closing“, also das gezielte Aufsuchen bereits besuchter Bereiche durch die Bewegung in Kreisen. Damit das funktioniert müssen zwei Bedingungen erfüllt sein:

1. Man muss Einfluss auf den Verlauf des Bewegungspfades haben.
2. Es muss simultan zur Lokalisierung eine Karte aufgebaut werden, um Kreise im Pfad detektieren zu können.

Zur Erfüllung der ersten Bedingung wird in einigen der vorgestellten, kartenlosen Lokalisierungsverfahren daher die Möglichkeit zur simultanen Kartenerzeugung berücksichtigt. Jedoch die Erfüllung der zweiten Bedingung kann oder soll in vielen Anwendungen nicht möglich sein. Alternativ könnte zwar auf externe Positionsbestimmungen (z.B. GNSS)

zurückgegriffen werden, um die aufsummierten Fehler zu eliminieren, aber auch diese Lösung kann nicht in jeder Anwendung genutzt werden. Für eine anwendungsübergreifende Betrachtung des Themas „Lokalisierung“ muss also auch die kartenbasierte Lokalisierung betrachtet werden.

### **Markerbasierte Lokalisierung**

Eine einfache Umsetzung der bereits betrachteten, marker-basierten Lokalisierung auf eine Lösung mit globaler Karte, ist die Dokumentation der Markerposen und der darin gespeicherten Informationen [145, 146, 147, 148]. Aufgrund der relativen Posenbestimmung des Markers relativ zur Kamera, kann mit bekannter, absoluter Pose des Markers auch die absolute Pose der Kamera abgeleitet werden. Eine Umsetzung dieser Art der Lokalisierung wird ausführlich in Kapitel 5.3 der vorliegenden Arbeit behandelt und evaluiert.

### **Geometrische Lokalisierung**

Eine ausschließlich geometrische Herangehensweise wird in den analytischen Verfahren in [149] bzw. in [150] vorgestellt. Die Umgebung wird hier als zweidimensionale Karte angenommen, in der sich ein punktförmiger Roboter bewegen (nur durch Rotation auf der Stelle und vorwärts Fahren) kann. Als Sensorik stehen lediglich Odometer zur Verfügung, die angeben wie weit der Roboter nach vorne gefahren ist und um wie viel Grad er sich dreht. Die Umgebung wird entweder durch Kollisionen mit einer Wand [149] erfasst oder auf Distanz durch einen nicht näher spezifizierten Sensor [150]. Gezeigt wird, dass sich der Roboter, alleine durch die geometrische Analyse seiner Umgebung, global lokalisieren und auf diese Weise das sogenannte „kidnapped robot“-Problem<sup>8</sup> lösen kann. Eine praxisnahe Umsetzung der geometrischen Lokalisierungsmethoden sind in [151, 152, 153, 154] zu finden. Diese eignen sich aufgrund ihrer Umgebungsrepräsentation ausschließlich für eine Indoor-Lokalisierung.

---

<sup>8</sup>Das „kidnapped robot“-Problem beschreibt die Aufgabe für einen mobilen Roboter sich in einer kartierten Umgebung absolut lokalisieren zu können, nachdem er an einer ihm bekannten Position aufgenommen und anschließend an einer willkürlichen Position ausgesetzt wird, ohne dass die internen Lokalisierungsmechanismen entsprechend aktualisiert werden, d.h. der Roboter sich noch an der alten Position wähnt.

## Monte Carlo Lokalisierung (MCL)

Die verbreitetste Form der kartenbasierten Lokalisierung stellt die probabilistische oder Monte Carlo Lokalisierung (MCL) dar. Eine grundlegende Einführung und Übersicht zu diesem Thema geben Thrun et al. in ihrem Buch „Probabilistic Robotics“ [155], das als Standardwerk auf diesem Gebiet zu bezeichnen ist. Basierend auf den dort beschriebenen Methoden sind zahlreiche Veröffentlichung zur Lokalisierung auf Basis von Partikelfiltern [156, 157, 158, 12, 159] oder Kalmanfiltern [160, 161, 162, 163] entstanden. Auch in dieser Arbeit, im Rahmen der semantischen Umgebungserfassung und landmarkenbasierten Lokalisierung, wird ein Lokalisierungsansatz genutzt der auf einem Partikelfilter beruht, wie er in [155] beschrieben ist. Aus diesem Grund werden die Grundlagen dazu in Kapitel 2.3.2 intensiv behandelt.

Da die Lokalisierung auf verschiedene Sensordaten als Beobachtungen oder Messungen angewiesen ist, und diese Beobachtungen aufgrund vielfacher Einflüsse, wie Messrauschen, Messfehler, Fehldetektionen, Verwechslungen und Fehlinterpretationen, zu ungenauen bis zum Teil fehlerbehafteten Daten führen kann, werden die Eingangsdaten häufig Filter verwendet, um auf die fehlerbehafteten Daten entsprechend reagieren zu können. Zwei grundlegende Konzepte, die sich im Kontext der Lokalisierung bereits stark etabliert haben sind der sequentielle Monte-Carlo-Filter, auch als „Partikel-Filter“ bekannt, sowie der erweiterte Kalman-Filter.

**Partikel-Filter** Ein zur Lokalisierung verwendeter Partikel-Filter nimmt die Sensorbeobachtungen entgegen und versucht das Problem nicht analytisch zu lösen, indem ein Gleichungssystem aufgestellt wird, das anschließend minimiert und daraufhin eine Lösung bestimmt werden kann. Stattdessen werden, um einen initialen Startpunkt herum, viele verschiedene Hypothesen aufgestellt, die jeweils einer Position zum aktuellen Zeitpunkt entsprechen. Formal ausgedrückt sind diese Hypothesen Stichproben oder „samples“ einer A-posteriori-Wahrscheinlichkeit und werden als

$$X_t := x_t^{[1]}, x_t^{[2]}, \dots, x_t^{[M]} \quad (2.14)$$

bezeichnet. Dabei stellt jeder Partikel  $x_t^{[m]}$ , mit  $1 \leq m \leq M$ , eine konkrete Instanziierung des Zustands zum Zeitpunkt  $t$  dar [164].

Die Anzahl  $M$  der Partikel ist in der Regel sehr hoch (z.B.  $M = 1000$ ), um den Suchraum möglichst gut abzudecken, d.h. die bayessche Annahme

$$b_t(x) = p(x|z_{1:t}, u_{1:t}) = p(x|z_{1:t}), \quad (2.15)$$

mit dem aktuellen Zustand  $x$  sowie den Steuergrößen  $u_t$  und Messwerten  $z_t$ , zu approximieren.

Zur Bewertung der einzelnen Partikel wird jeweils ein entsprechender Gewichtungsfaktor

$$w_t^{[m]} = p(z_t|x_t^{[m]}) \quad (2.16)$$

anhand der aktuellen Messwerte bestimmt, also die Wahrscheinlichkeit ermittelt, dass die jeweilige Hypothese des aktuellen Zustands zu den Beobachtungen  $z_t$  führen kann. Dazu wird eine im Vorfeld bekannte Karte vorausgesetzt, die eine solche Bewertung der Messwerte erlaubt.

In einem konkreten Beispiel wären die Messwerte die Beobachtung von verschiedenen Landmarken an bestimmten Positionen im lokalen Koordinatensystem des Beobachters. Die Karte ist dann eine im Vorfeld bekannte Verteilung der Landmarken im Gebiet, worin sich der Beobachter zur Zeit aufhält. Die Partikel repräsentieren Annahmen der aktuellen Position bezüglich vorheriger Berechnungen oder, zu Anfang, bezüglich einer initial gegebenen Startposition. Nun werden für alle Annahme die beobachteten Landmarkenpositionen mit den tatsächlich von der angenommenen Position in der Karte sichtbaren Landmarkenpositionen verglichen und der Fehler dazwischen bestimmt. Dieser Fehler spiegelt wider, wie wahrscheinlich die angenommene Position bezüglich der tatsächlich aufgenommenen Beobachtungen ist, siehe Gleichung 2.16.

Nachdem die Partikel verteilt und jeweils einzeln bewertet (gewichtet) wurden, werden im nächsten Schritt alle Partikel deren Gewichtung unterhalb einer festgelegten Grenze liegen verworfen und die Partikel mit ausreichend hoher Gewichtung im Folgenden näher betrachtet. Neue Partikel werden nun, im sogenannten „Resampling“-Schritt gehäuft um solche Partikel verstreut, die im letzten Schritt eine hohe Gewichtung erhalten haben. Durch diese Häufung werden die Partikel in diesem Bereichen dichter verteilt. Ziel ist ein Konvergieren des Filters auf eine Position, die dann mit hoher Wahrscheinlichkeit nahe der gesuchten Position liegt. Formal gesprochen werden die Partikel nun also nicht mehr bezüglich der Annahme  $b_t(x)$  aus Gleichung 2.15 sondern (approximiert)

anhand der neuen Annahme

$$b'_t(x) = p(z_t, x_t^{[m]})b(x_t). \quad (2.17)$$

Eine tiefergehende Betrachtung des Partikelfilters samt zugrundeliegender Theorien soll an dieser Stelle nicht erfolgen. Stattdessen wird an dieser Stelle auf die Quellen [164] verwiesen, worin auch verschiedene Resampling-Strategien tiefergehend untersucht und deren mathematische Grundlagen ebenfalls beschrieben werden. Außerdem sei hier ebenfalls auf [165, 16, 12] verwiesen, wo die Partikelfilter-basierte Monte Carlo Lokalisierung im Kontext der vorliegenden Arbeit umgesetzt wurde.

**Kalman-Filter** Allgemein gesprochen werden Kalman-Filter dazu verwendet, mehrere Messdaten miteinander, unter der Berücksichtigung von Messungenauigkeiten, zu fusionieren und sowohl eine Aussage über eine abgeleitete Größe, als auch über ihre wahrscheinliche Genauigkeit. Die Idee besteht darin, dass die Messwerte von Sensoren zwar nie exakt sind, aber normalverteilt um einen korrekten Mittelwert mit einer, für den jeweiligen Sensor typischen, Varianz liegen. Den korrekten Mittelwert gilt es zu ermitteln. Im eindimensionalen Fall lässt sich beispielsweise die Position eines Punktes auf einer Geraden durch das regelmäßige Messen der Geschwindigkeit bestimmen, wenn die initiale Position bekannt angenommen werden kann. In diesem Fall wird die Geschwindigkeit regelmäßig über einen Sensor gemessen und die resultierende Position auf der Geraden aktualisiert. Anstatt lediglich die Messwerte zu addieren, versucht das Kalman-Filter die Messungenauigkeiten zu berücksichtigen und bereits bevor neue Messergebnisse eintreffen, eine Vorhersage anhand der letzten Berechnungen zu berechnen (predict). Anschließend wird diese Vorhersage anhand der eintreffenden Messdaten aktualisiert und ggf. korrigiert und neu bewertet (update). Diese beiden Teilschritte werden während der Laufzeit des Filters kontinuierlich wiederholt:

Das Kalman-Filter gehört zur Familie der Gauß-Filter, die wiederum eine Implementierung der Bayes-Filter im Kontinuierlichen darstellen. Wie in allen Gauß-Filtern werden Annahmen durch mehrdimensionale

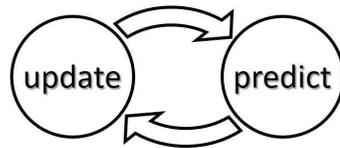


ABBILDUNG 2.13: In einem Kalman-Filter werden die beiden Schritte „Update“ und „Predict“ kontinuierlich im Wechsel aufgerufen.

Normalverteilungen dargestellt. In [164] werden diese mehrdimensionalen Normalverteilungen folgendermaßen definiert:

$$p(x) = \det(2\pi\Sigma)^{-\frac{1}{2}} \exp \left\{ -\frac{1}{2}(x - \mu)^T \Sigma^{-1}(x - \mu) \right\}. \quad (2.18)$$

Die von  $x$  abhängige Wahrscheinlichkeitsdichtefunktion  $p(x)$  wird durch den Mittelwert  $\mu$  und die Kovarianzmatrix  $\Sigma$  parametrisiert. Der Vektor  $\mu$  hat dabei die gleiche Dimension wie  $x$  und die symmetrische Matrix  $\Sigma$  hat die Dimension von  $x$  im Quadrat. Das Kalman-Filter repräsentiert nun die Annahme aus Gleichung 2.18 zum Zeitpunkt  $t$  durch die beiden Parameter  $\mu_t$  und  $\Sigma_t$ .

Die Grundvoraussetzung für das Kalman-Filter ist, dass die a-posteriori-Wahrscheinlichkeitsverteilung der Annahme  $p(x_t)$  zu jedem Zeitpunkt  $t$  einer Gaußverteilung entspricht. Dies ist genau dann gewährleistet, wenn folgende drei Punkte erfüllt sind:

1. Die Übergangswahrscheinlichkeit  $p(x_t|u_t, x_{t-1})$  muss eine lineare Funktion sein, überlagert mit Gauß'schem Rauschen.
2. Die Messgrößenwahrscheinlichkeit  $p(z_t|x_t)$  muss ebenfalls linear in ihren Argumenten sein, mit überlagertem Gauß'schem Rauschen.
3. Die initiale Annahme  $p(x_0)$  muss normalverteilt sein.

[164] beschreibt das Kalman-Filter wie folgt: Als Eingabe erhält das Kalman-Filter die den zum Zeitpunkt  $t - 1$  angenommenen Mittelwert  $\mu_{t-1}$ , sowie die zugehörigen Kovarianzen in der Matrix  $\Sigma_{t-1}$ . Für den *Update*-Schritt benötigt das Kalman-Filter zudem die Steuerparameter  $u_t$  und die aktuellen Messwerte  $z_t$ . Ergebnis ist dann die Annahme zum Zeitpunkt  $t$ , bestehend aus  $\mu_t$  und  $\Sigma_t$ . Die vorhergesagte (*predicted*) Annahme aus  $\bar{\mu}$  und  $\bar{\Sigma}$  wird einen Schritt später berechnet, jedoch noch ohne die

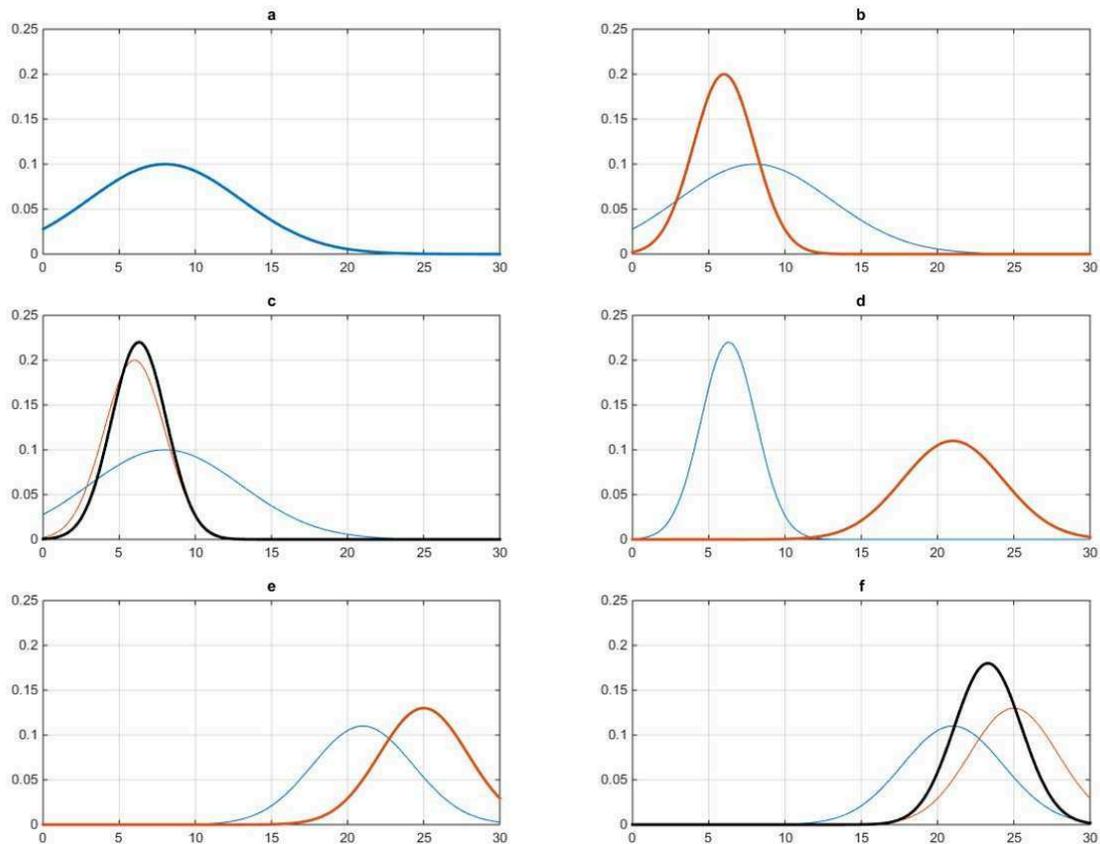


ABBILDUNG 2.14: Illustration des Kalman-Filters: (a) Initiale Annahme, (b) eine Messung mit assoziierter Unsicherheit, (c) Annahme, nach Integration der Messung durch das Kalman-Filter, (d) Annahme nach Bewegung nach rechts (verursacht weitere Unsicherheit), (e) eine neue Messung mit Unsicherheiten und (f) die resultierende Annahme.

Messung  $z_t$  zu berücksichtigen. Stattdessen werden die Steuerungswerte  $u_t$  berücksichtigt. Der Mittelwert wird dann durch die Verwendung der deterministischen Variante der Zustandsübergangsfunktion

$$x_t = A_t x_{t-1} + B_t u_t + \epsilon_t \quad (2.19)$$

ermittelt, wobei der Zustandsvektor  $x_{t-1}$  durch den Mittelwert  $\mu_{t-1}$  ersetzt wird. Darin stellen  $x_t$  und  $x_{t-1}$  die Zustandsvektoren zu den jeweiligen Zeitpunkten  $t$  und  $t-1$  dar. Die Matrix  $A_t$  ist quadratisch in der Dimension  $n$  des Zustandsvektors  $x_t$ , und die Matrix  $B_t$  ist von der Dimension  $n \times m$ , wobei sich  $m$  aus der Dimension des Steuerungsvektors  $u_t$  ergibt. Die Gauß'sche Zufallsvariable  $\epsilon_t$  besitzt dieselbe Dimension

wie der Zustandsvektor und modelliert die Unsicherheiten beim Zustandsübergang. Dass Zustände von ihren vorhergehenden Zuständen abhängen wird hier durch die lineare Matrix  $A_t$  berücksichtigt, die gleich zweimal in die Kovarianzmatrix multipliziert wird, da diese quadratisch ist. Über eine Gewichtung, dem sogenannten *Kalman Gain*  $K_t$ , kann festgelegt werden, in welchem Maße die letzte Messung in den neuen Zustandsvektor einfließen soll. Die Differenz zwischen der vorhergesagten Messung und der tatsächlichen Messung  $z_t$  wird beim Kalman-Filter als *Innovation* bezeichnet. Letztlich werden die neuen Kovarianzen der a-posteriori-Annahme, entsprechend der Messung, berechnet. Eine Darstellung der einzelnen Zwischenschritte befindet sich in Abbildung 2.14.

Die Annahme, dass Beobachtungen lineare Funktionen des Zustands sind, und dass der nächste Zustand eine lineare Funktion des vorhergehenden Zustands ist, sind ausschlaggebend für die Richtigkeit des Kalman-Filters. Wenn aber die Voraussetzung der Linearität nicht erfüllt ist, so existieren Möglichkeiten, diese Linearitätsbedingung zu entschärfen. Eine analytische Herangehensweise nutzt das Extended Kalman-Filter, wohingegen das Unscented Kalman-Filter eine statistische Herangehensweise anstrebt. Eine ausführliche Betrachtung der beiden Linearisierungsverfahren für Kalman-Filter sind ebenfalls in [164] zu finden und sollen daher in dieser Arbeit nicht nähergehend behandelt werden.



## Kapitel 3

# Konzept

Grundlegend für die Interaktion von mobilen System mit ihrer Umgebung ist eine gegenständliche und räumliche Umgebungsbeschreibung. Kern der vorliegenden Dissertation ist ein allgemeines Konzept zur sensorischen Umgebungserfassung und anwendungsübergreifenden Umweltmodellierung. Ziel ist die Erzeugung eines Weltmodells, basierend auf einer semantischen Umgebungsbeschreibung gepaart mit einem landmarkenbasierten Lokalisierungsverfahren. Eine generische Darstellung des Gesamtkonzepts ist in Abbildung 3.1 abgebildet und wird in diesem Kapitel im Detail vorgestellt. Dazu muss auch der Begriff des anwendungsübergreifenden Semantischen Weltmodells definiert werden, da dieser eine zentrale Rolle in der vorliegenden Arbeit spielt. Die Umsetzung des allgemeinen Konzepts in konkretem Anwendungsgebieten ist Gegenstand der folgenden Kapitel. Die zur Umsetzung genutzte Simulationsplattform VEROSIM wird zuvor in Abschnitt 3.3 dieses Kapitels vorgestellt. Ebenfalls für die konkrete Umsetzung in verschiedenen Anwendungsfällen verwendet, wurden eine mobile Roboterplattform, die in Abschnitt 3.4 vorgestellt wird, sowie eine im Rahmen dieser Arbeit entwickelte Umgebungserfassungsplattform. Die detaillierte Beschreibung dieser Einheit ist Gegenstand von Abschnitt 3.5. Eine weiterführende Betrachtung des allgemeinen Konzepts unter der Berücksichtigung von simulationsgestützten Methoden zur Planung und Steuerung wird in Abschnitt 3.6 gegeben und schließt dieses Kapitel ab.

### 3.1 Umgebungserfassung, Lokalisierung und Umweltmodellierung

Das grundlegende Konzept dieser Arbeit ist in Abbildung 3.1 dargestellt und untergliedert sich zunächst grob in zwei Bereiche: Der erste Bereich umfasst die sensorische Erfassung und anwendungsübergreifende Modellierung der lokalen Umgebung eines mobilen Systems. Um eine Unabhängigkeit von der Vielzahl an unterschiedlichsten Sensordatenformaten zu erreichen (vgl. hierzu Kapitel 2.1.4) wird hier zunächst eine Abstraktion der Sensordaten vorgenommen. Diese reduziert die im Weiteren zu betrachtenden Formate auf wenige Abstraktionsklassen und erlaubt somit eine Entkopplung der Erfassungsalgorithmen von konkreten Sensortypen. Die angestrebte, anwendungsübergreifende Umgebungsbeschreibung wird erreicht, indem vorab verschiedene Anwendungen betrachtet und deren Anforderungen an die Umgebungserfassung analysiert werden. Auf dieser Grundlage werden anwendungsübergreifende Modellbeschreibungen erzeugt, die wiederum als Grundlage für die zu entwickelnden Detektionsalgorithmen dienen, mit deren Hilfe aus den abstrahierten Sensordaten Instanzen der jeweiligen Modellbeschreibungen erzeugt werden können. Die Detektion, Erkennung und Beschreibung von Objekten ist Aufgabe der semantischen Umgebungserfassung, dem ersten Teil des allgemeinen Konzepts.

Damit ein mobiles System die erfassten Umgebungsinformationen auch räumlich zuordnen kann, benötigt es die Fähigkeit die eigene Position in der Umgebung zu bestimmen, was im Wesentlichen den zweiten Teil des allgemeinen Konzepts ausmacht. Es erfolgt entweder ein Abgleich der lokal erfassten Umgebungsinformationen mit den Informationen aus einem globalen Umweltmodell, oder die lokalen Detektionen werden über die Zeit verfolgt und erlauben es die Bewegung des mobilen Systems zu rekonstruieren. Eine Kombination mehrerer dieser Verfahren ist möglich und wird im Laufe dieser Arbeit auch verwendet. Durch die Kombination der Umgebungserfassung mit der Lokalisierung wird die Erzeugung eines semantischen Weltmodells ermöglicht, das als Datengrundlage für ein globales Lokalisierungsverfahren sowie für semantische Umweltanalysen dient.

Alle wesentlichen Komponenten dieses Konzepts, also die semantische Umgebungserfassung, die Lokalisierung und die Umweltmodellierung werden in den folgenden Kapiteln im Detail behandelt. Zuvor soll

### 3.1. Umgebungserfassung, Lokalisierung und Umweltmodellierung 57

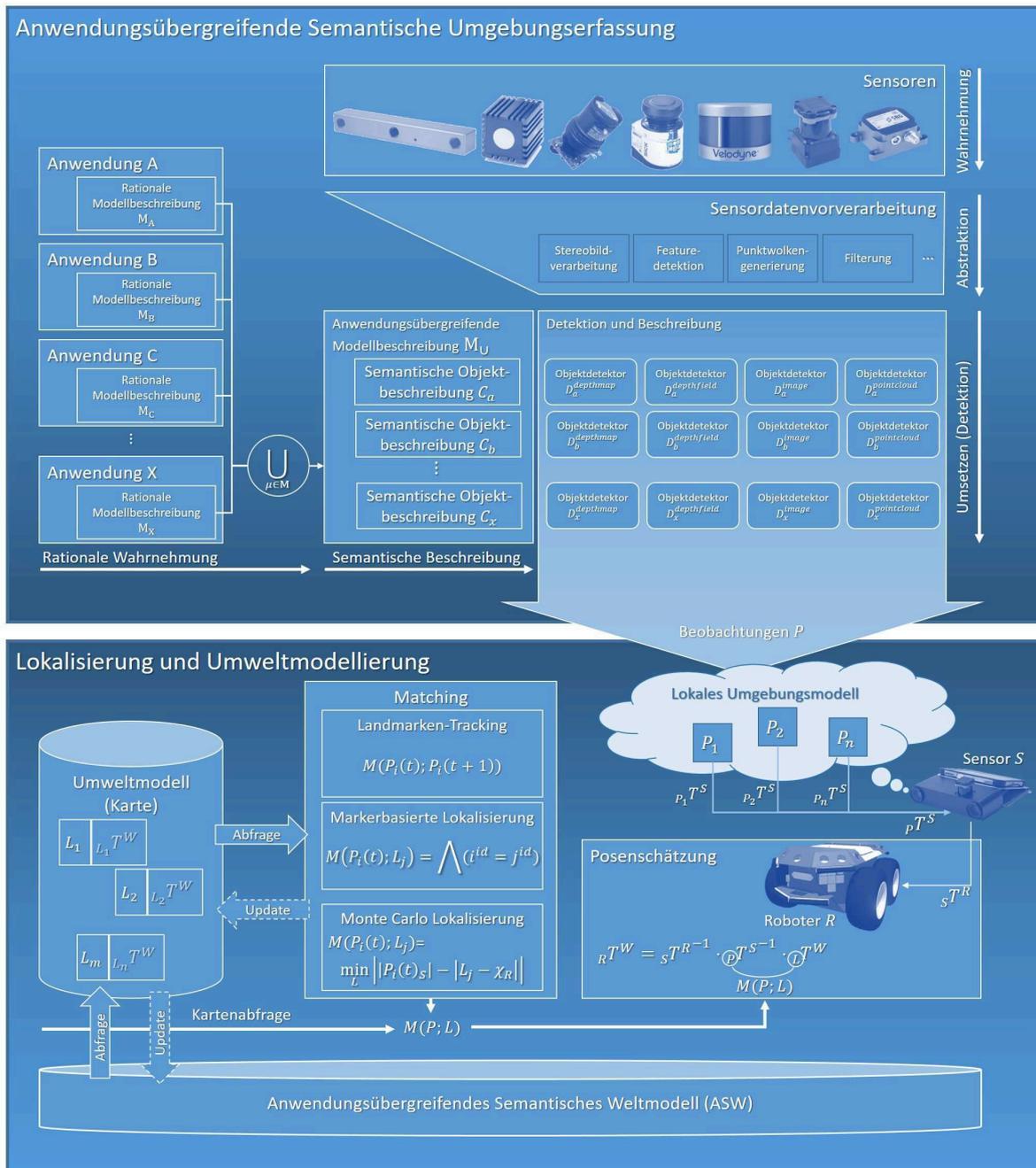


ABBILDUNG 3.1: Grundlegendes Konzept dieser Arbeit. Zentrale Punkte sind die anwendungsübergreifende Umwelterfassung und Beschreibung sowie die auf den lokal erfassten semantischen Landmarken operierende Lokalisierung. Der Abgleich des lokalen Umgebungsmodells mit dem globalen Umweltmodell erlaubt den sukzessiven Aufbau eines anwendungsübergreifenden Semantischen Weltmodells (ASW).

nun aber eine allgemeine Definition für den Begriff des Semantischen Weltmodells gegeben werden und die, im Rahmen dieses Konzepts, wesentlichen Aspekte vorgestellt werden.

## 3.2 Anwendungsübergreifende Semantische Weltmodelle

Diese Arbeit baut wesentlich auf der Idee der Anwendungsübergreifenden Semantischen Weltmodelle (ASW) auf. Aus diesem Grund sollen zunächst die Begriffe „Modell“ und „Semantik“ näher untersucht werden, sodass im Anschluss eine Definition für das „Semantische Weltmodell“ gegeben werden kann.

Der Verein Deutscher Ingenieure (VDI) definiert den Begriff „Modell“ als vereinfachte Nachbildung eines geplanten oder existierenden Systems mit seinen Prozessen in einem anderen begrifflichen oder gegenständlichen System. Es unterscheidet sich hinsichtlich der untersuchungsrelevanten Eigenschaften nur innerhalb eines vom Untersuchungsziel abhängigen Toleranzrahmens vom Vorbild [166]. Abbildung 3.2 veranschaulicht hier den Wahrnehmungsprozess und die resultierende Modellerzeugung.

Diese Beschreibung ist eine Spezialisierung des allgemeinen Modellbegriffs von Herbert Stachowiak der ihn in [167] mit Hilfe dieser drei Hauptmerkmale definiert:

- **Abbildungsmerkmal:** Modelle sind stets Modelle *von etwas*, nämlich Abbildungen, Repräsentationen natürlicher oder künstlicher Originale, die selbst wieder Modelle sein können.
- **Verkürzungsmerkmal:** Modelle erfassen im allgemeinen *nicht alle* Attribute des durch sie repräsentierten Originals, sondern nur solche, die den jeweiligen Modellerschaffern und/oder Modellbenutzern relevant scheinen.
- **Pragmatisches Merkmal:** Modelle sind ihren Originalen nicht per se eindeutig zugeordnet. Sie erfüllen ihre Ersetzungsfunktion
  - a. für *bestimmte* – erkennende und/oder handelnde, modellbenutzende – Subjekte,

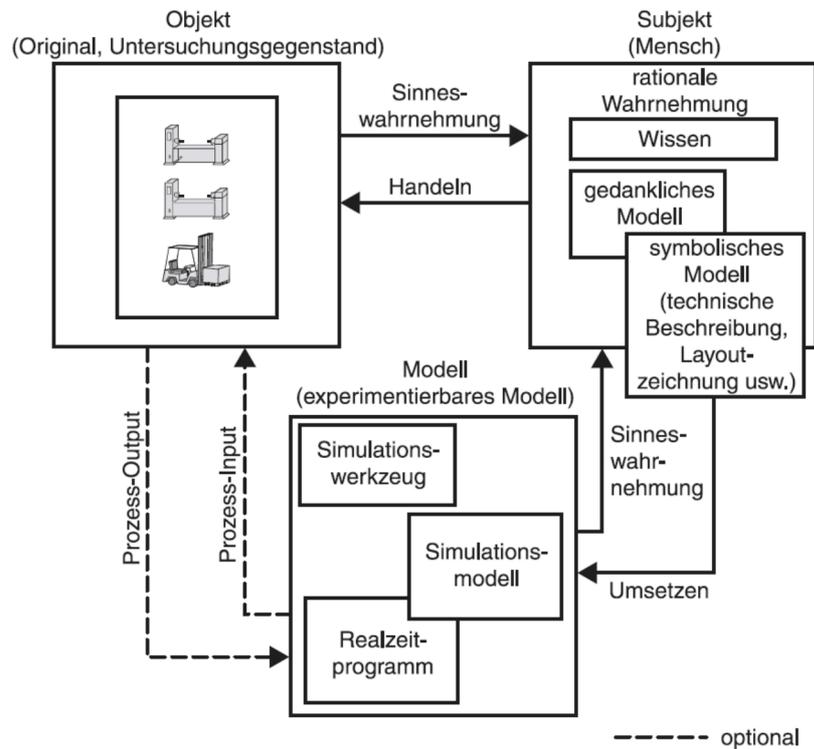


ABBILDUNG 3.2: Modelldefinition nach VDI-Richtlinie 3633.  
Bildquelle: [166]

- b. innerhalb bestimmter Zeitintervalle und
- c. unter Einschränkung auf bestimmte gedankliche oder tatsächliche Operationen.

Aufbauend auf diesen Betrachtungen des allgemeinen Modellbegriffs ergeben sich für ein Weltmodell folgende Anforderungen:

- Ein Weltmodell ist eine vereinfachte Nachbildung der Welt und repräsentiert diese anhand anwendungsrelevanter Attribute.
- Das Weltmodell muss eine Ersetzungsfunktion im Rahmen der modellbenutzenden Anwendung(en) erfüllen.

Daraus ergibt sich, dass letztlich die Anwendungen die jeweilig relevanten Attribute festlegen und das anwendungsübergreifende Modell eine Vereinigung dieser einzelnen Modelle darstellt (siehe Abbildung 3.3).

Der Begriff der Semantik wird in der Fachliteratur in erster Linie im Sinne der Zeichentheorie (Semiotik) als die Theorie oder Wissenschaft

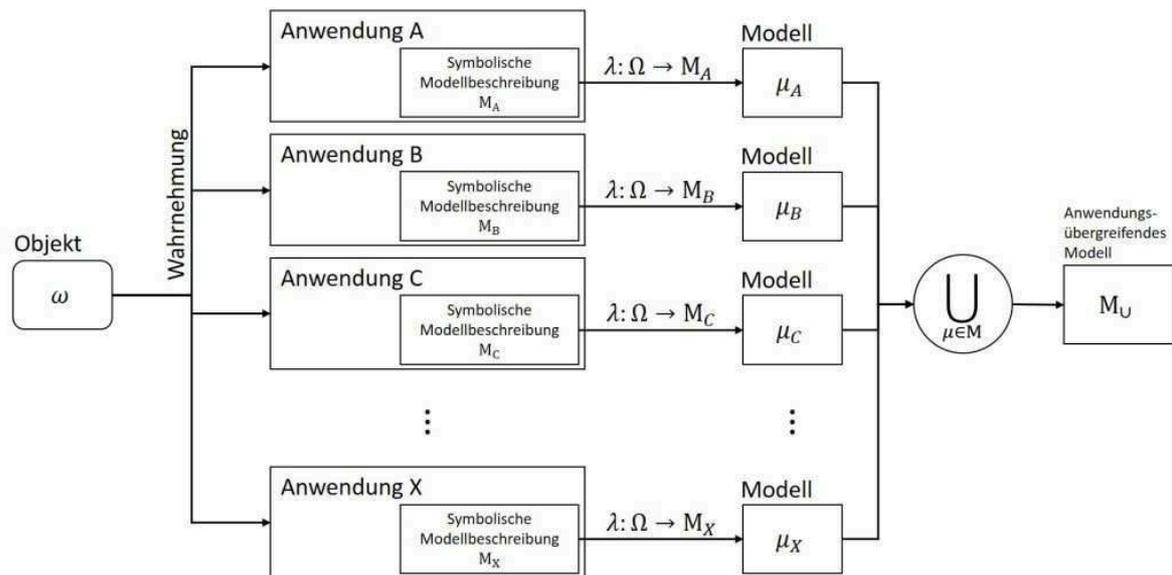


ABBILDUNG 3.3: Erzeugung anwendungsübergreifender Modelle als Vereinigung der Modellbeschreibungen aller Anwendungen.

von der Bedeutung der Zeichen verwendet. Aber auch losgelöst von der Zeichentheorie wird der Begriff Semantik dann verwendet, wenn etwas *bezeichnet* oder *mit einer Bedeutung belegt* wird. Beispielsweise werden in der Informatik formale Sprachen über ihre Syntax und Semantik beschrieben, wobei die Syntax alle zulässigen Wörter der Sprache definiert und die Semantik die Bedeutung dieser Wörter festlegt. In [168] führt Peter Chen das Entity-Relationship Modell (ERM) zur Datenmodellierung ein, und schreibt: *“This model incorporates some of the important semantic information about the real world.”* Aus diesem Grund wird das ERM auch häufig als *“Semantisches Datenmodell”* bezeichnet und als abstrakte, formale Beschreibung und Darstellung eines Ausschnittes der in einem bestimmten Zusammenhang „wahrgenommenen Welt“. Als wesentliche Arbeitsinhalte bei der Erstellung eines semantischen Datenmodells nach dem ERM sind festgelegt:

- Aus den als kontextrelevant ermittelten *“Informationsbegriffen”* werden *Einheiten (Entitäten)* identifiziert und als Entitätstypen benannt, über die Informationen verarbeitet und/oder gespeichert werden müssen.

- Es wird ermittelt, *welche Beziehungen* (ebenfalls kontextrelevant) *zwischen diesen Entitäten* bestehen oder bestehen können

Für ein semantisches (Umwelt-)Modell heißt dies, es muss ein Semantisches Datenmodell zugrunde liegen, das zum Einen die beinhalteten Entitätstypen benennt, also festlegt, was Teil des Umgebungsmodells ist und was nicht, und zum Anderen die kontextrelevanten Beziehungen zwischen diesen Entitätstypen wiedergibt. Damit legt das semantische Datenmodell fest

- welche Attribute der Umwelt abgebildet werden sollen,
- welche Bedeutung ihnen zukommt und
- welche kontextrelevanten Beziehungen sie ggf. untereinander haben.

In [169] gibt Andreas Nüchter eine Definition für Semantische 3D-Karten in der mobilen Robotik: *"A semantic 3D map for mobile robots is a metric map, that contains in addition to geometric information of 3D data points, assignments of these points to known structures or object classes"*. Mit dieser Definition beschränkt Nüchter den Begriff der semantischen 3D-Karte auf klassifizierte 3D-Punktwolken. Diese Punktwolken stammen meist von Laserscannern, die aufgrund ihres großen Sichtfelds und hohen Tiefengenauigkeit (vgl. hierzu den Abschnitt 2.1.2) im Bereich der mobilen Robotik bevorzugt eingesetzt werden.

Im Forschungsbereich autonomer Fahrzeuge werden ebenfalls bevorzugt Laserscanner als primäre Sensoren eingesetzt (z.B. [170, 171, 172]) um das Umfeld des Fahrzeugs zu überwachen und andere Verkehrsteilnehmer zu detektieren und zu erkennen. Anhand der semantischen Unterscheidung der einzelnen Verkehrsteilnehmer lassen sich für das selbstfahrende Auto Prognosen ableiten, wie sich die jeweiligen Zustände der erkannten Objekte wahrscheinlich ändern und welche Aktionen des autonomen Fahrzeugs dadurch als sicher eingestuft werden, um sie letztendlich auszuführen. Autonome Fahrzeuge benötigen also ein lokales Umgebungsmodell basierend auf bekannten Objekten mit einer Bedeutung, um selbstständig Aktionen zu planen und durchzuführen.

Basierend auf diesen Vorüberlegungen leitet sich somit folgende formale Definition für ein semantisches Umweltmodell ab:



ABBILDUNG 3.4: Definition und Abgrenzung der Begriffe Feature, Landmarke und semantische Landmarke.

**Definition 3.1** *Ein semantisches Umweltmodell ist eine gegenständliche und anwendungsbezogene Teilrepräsentation der erfassbaren Umgebung bestehend aus erkennbaren Objekten mit einer Bedeutung und einem zugrundeliegenden semantischen Datenmodell*

Im Grunde kann jede Karte auch als Umweltmodell angesehen werden, da sie die Umgebung in einer bestimmten Weise beschreibt. Andererseits ist die Tauglichkeit eines Umgebungsmodells als Karte nicht zwangsläufig gegeben, da ein Umgebungsmodell auch ausschließlich aus Informationen bestehen kann, die sich nicht für eine Lokalisierung eignen (in Kapitel 4.1 werden die Anforderungen an die Lokalisierung genauer betrachtet), bzw. keine Lokalisierungslösung für die Daten des vorliegenden Umgebungsmodells umgesetzt ist.

Nun müssen die zu erkennenden Objekte näher untersucht werden um das semantische Umweltmodell dieser Arbeit genauer von den semantischen 3D-Karten aus [173] abzugrenzen. Dazu sollen die Begriffe Merkmal (Feature), Landmarke und semantische Landmarke definiert werden. Der Einfachheit halber soll diese Definition anhand eines Beispiels geschehen. Abbildung 3.4 zeigt eine Kiste (Mitte), die in die Bildebene eines Sensors projiziert wird (links).

In der Projektion (im Sensorraum) werden meist Merkmale (Features) gesucht und anhand vieler Kriterien beschrieben (Feature-Deskriptor) sodass diese Merkmale mit den Merkmalen einer späteren Aufnahme verglichen werden können. In der Projektion sind die Merkmale (Eckpunkte der abgebildeten Kiste) durch rote Kreise gekennzeichnet. Features sind demnach Merkmale in Sensordaten als Ergebnis einer sensorspezifischen Featuredetektion. Werden die Merkmale nicht im Sensorraum sondern im Raum der Szene durch 3D-Koordinaten angegeben, so wird

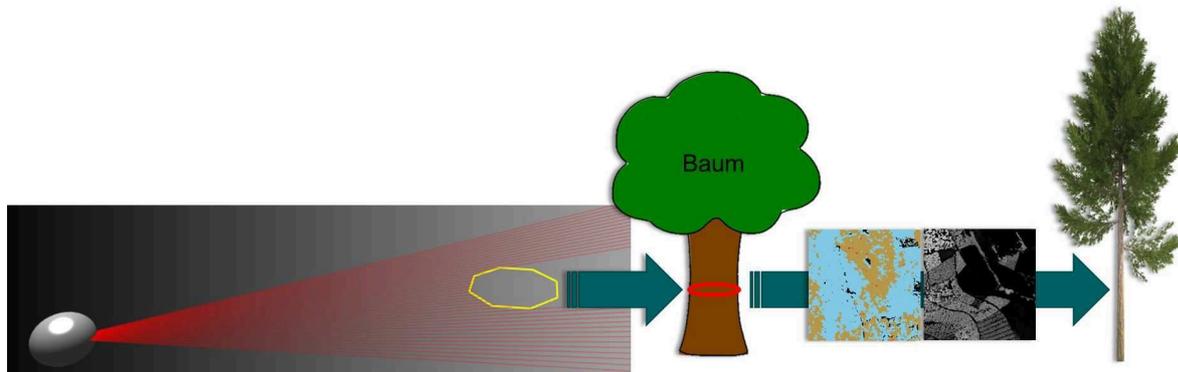


ABBILDUNG 3.5: Erweiterung semantischer Objekte um weitere Attribute aus externen Datenquellen. v.l.n.r.: In den Laserdaten wird ein Kreis detektiert. Die semantische Erfassung erkennt daraus einen Baum mit bestimmtem Stammdurchmesser. Durch die Nutzung weiterer Daten wird ein explizit beschriebenes Baummodell erzeugt.

zur Unterscheidung oft von Landmarken gesprochen. Im Beispiel entsprechen die 3D-Koordinaten der Kistenecken (Mitte, rote Markierungen) diesen Landmarken. Landmarken sind also 3D-Merkmale als Ergebnis einer unspezifischen Objektdetektion. Erst durch eine semantische Umgebungserfassung auf Basis eines semantischen Datenmodells wo Kisten als modellierbare Entitäten enthalten sind, können die detektierten Eckpunkte in einem Erkennungsschritt als Kiste mit einer entsprechenden Kantenlänge erkannt werden (in der Abbildung 3.4 rechts). Durch diesen letzten Schritt wird aus einer unspezifischen Objektdetektion eine semantische Objekterkennung. Dabei gibt die Erkennung der Detektion eine Bedeutung.

Darüber hinaus können die semantischen Objekte, durch die Bestimmung und das Hinzufügen weiterer Attribute, spezifiziert und ggf. eindeutig(er) beschrieben werden. Diese Informationen werden ebenfalls aus den Sensordaten hergeleitet, oder, stehen dem Erfassungssystem weitere Datenquellen zur Verfügung, so lassen sich die dort gespeicherten Informationen mit den erfassten Objekten verschneiden. Ein Beispiel für eine solche Verschneidung ist in Abbildung 3.5 dargestellt. Hier wird der, in Kapitel 4.2.2 beschriebene, Baumdetektor für 2D-Laserscanner genutzt, um aus einem Scanprofil eine halbkreisförmige Struktur detektieren zu können, die dann zum Vollkreis extrapoliert und als Baum erkannt werden kann (gelber Kreis). Aus der Position des Kreismittelpunkts und dem

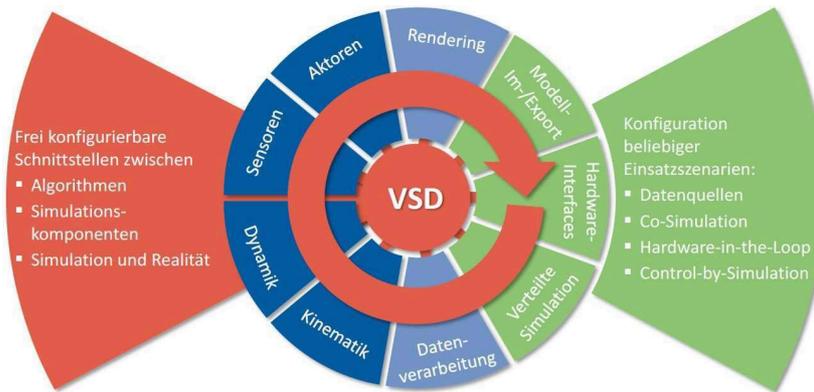


ABBILDUNG 3.6: Das Grundkonzept des Simulationssystems VEROSIM: Die Active Simulation Database (VSD) ist zentrale Datenhaltung und aktiver Kommunikationsknoten für die je nach Anwendung beteiligten Komponenten. Bildquelle: [174]

Kreisumfang wird ein semantisches Objekt "Baum" mit dem Stammdurchmesser als Attribut erzeugt. In Verbindung mit der in Kapitel 5.4 behandelten Lokalisierung lässt sich der Baum in einem Referenzkoordinatensystem angeben und mit, weiteren, vorhanden Daten verschneiden und ergänzen. Liegen also, wie in diesem Beispiel, eine Baumartenklassifizierung und ein Höhendifferenzmodell<sup>1</sup> für das zu betrachtende Gebiet vor, so lässt sich der Baum durch Verschneidung der vorliegenden Daten neben dem Stammdurchmesser auch mit seiner Baumart und Höhe beschreiben. Diese detaillierte Beschreibung ist für die Lokalisierung i.d.R. nicht von Bedeutung, kann aber als Grundlage für eine anschließende forstwirtschaftliche Untersuchung des Baumbestands genutzt werden. Eine nähere Betrachtung dieses Anwendungsfalls befindet sich in Kapitel 7.1.1.

### 3.3 Die Simulationsplattform VEROSIM™

Die in der vorliegenden Dissertation erarbeiteten Methoden wurden mit Hilfe des Simulationssystems VEROSIM (Virtual Environments and Robotics Simulation System) [174] realisiert. VEROSIM dient hier als

<sup>1</sup>Ein Höhendifferenzmodell berechnet sich aus einem Digitalen Oberflächenmodell (DOM) und einem Digitalen Geländemodell (DGM) durch Bildung der Differenz beider Modelle. Das Ergebnis ist ein bezüglich des Geländes normalisiertes Höhenmodell der Vegetation.

Plattform zur simulationsgestützten Entwicklung, Ausführung und Evaluation der umgesetzten Algorithmen. VEROSIM wird am Institut für Mensch-Maschine-Interaktion (MMI) der RWTH Aachen University als universelle Entwicklungsplattform und als Grundlage für die Erzeugung Virtueller Testbeds eingesetzt.

Angelegt an sogenannten Mikrokernarchitekturen besteht das Simulationssystem aus einem minimalen Kern, der lediglich die grundlegende, aktive, objektorientierte Echtzeitdatenbank, die VEROSIM Active Simulation Database (VSD), bereitstellt. Alle weiteren Funktionalitäten des Simulationssystems werden in Form von Softwaremodulen, sogenannten Plugins, hinzugefügt und erweitern dadurch die VSD um allgemeine Datengrundlagen, Algorithmen für Simulation, Datenverarbeitung, Steuerung und Regelung von Mechanismen zur zustands- und ereignisorientierten Kommunikation bis hin zur grafischen Darstellung [174].

Abbildung 3.6 veranschaulicht den modularen Aufbau des Simulationssystems durch einen kleinen VSD-Kern und in beliebiger Kombination ansteckbarer Plugins. Zu Kerngebieten existieren bereits mehrere, aufeinander abgestimmte Plugins, die sich zu Frameworks ergänzen, wie die Starrkörpersimulation, die Sensorsimulation oder das Rendering. Durch die Funktionalitäten dieser Frameworks und deren Integration in einer gemeinsamen Datenbasis, der VSD, können zu jeder Zeit fast beliebig komplexe Szenarien erstellt und simuliert werden.

Alle Zustandsinformationen liegen zu jedem Zeitschritt der Simulation, vollständig in der VSD und können zur Validierung der implementierten Algorithmen verwendet werden.

### 3.3.1 Virtuelle Testbeds

Das Konzept der vorliegenden Dissertation gliedert sich uneingeschränkt in die Definition eines datenverarbeitenden Systems (Abbildung 3.7) aus [175] ein. Dabei wird das datenverarbeitende System (*data processing system* - DPS) als Teil eines sogenannten Virtuellen Testbeds (VTB) angesehen. Das VTB besteht zum einen aus der Simulation von Sensoren, Aktuatoren und einem zu entwickelndem bzw. zu testenden System in einer simulierten Umgebung. Die von den Sensoren erzeugten Daten bilden die Eingangsgröße des DPS. Dort werden die Daten verarbeitet, es entsteht ein Umgebungsmodell des Systems und ggf. Ausgangsgrößen zur Ansteuerung von Aktuatoren. Finden alle diese Prozesse innerhalb

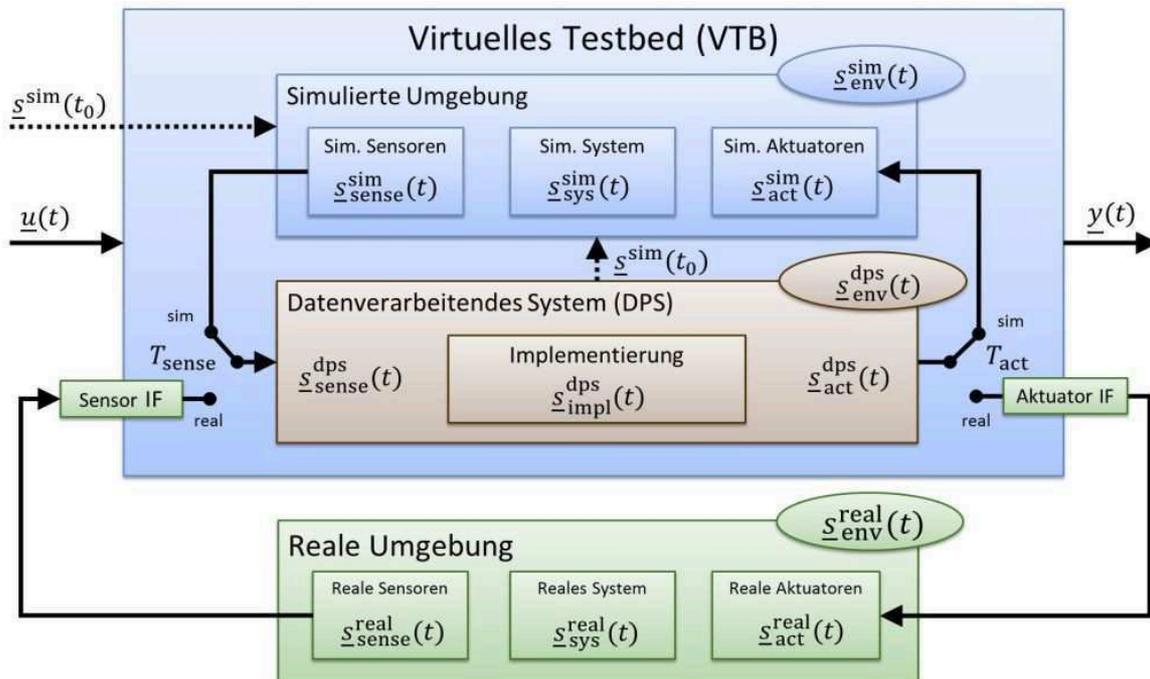


ABBILDUNG 3.7: Abhängigkeiten zwischen simulierten und realen Umgebungen und Komponenten, einem datenverarbeitendem System und einem Virtuellen Testbed. Bildquelle: [175]

eines Simulationssystems als Teil einer Simulation statt, so sprechen wir von einem Virtuellen Testbed. Darüber hinaus berücksichtigt das Konzept parallel auch reale Sensoren, reale Systeme und deren Aktuatoren in der realen Umgebung. Durch eine allgemeine und übergeordnete Schnittstelle für die Sensoren und Aktuatoren ermöglicht das Konzept jederzeit eine Umschaltung zwischen dem simulierten und dem realen System und bietet darüber hinaus sogar die Möglichkeit zum parallelen Betrieb beider Systeme.

Im Rahmen der Einführung des Industrie 4.0 Konzepts wurde im Bezug auf die modellbasierte Abbildung realer Objekte im Computer der Begriff *Digitaler Zwilling* oder *Digital Twin*<sup>2</sup> im Bereich der Industrie etabliert und in den Fokus der Öffentlichkeit gerückt. Ein Experimentierbarer Digitaler Zwilling (EDZ) ist nach VDI-Richtlinie 3633 die Erweiterung des Digitalen Zwillings um eine Simulation seiner dynamischen

<sup>2</sup>Nach der in 2012 publizierte Veröffentlichung [176] wurde der Begriff durch das „Defense Sciences Office“ der DARPA eingeführt.

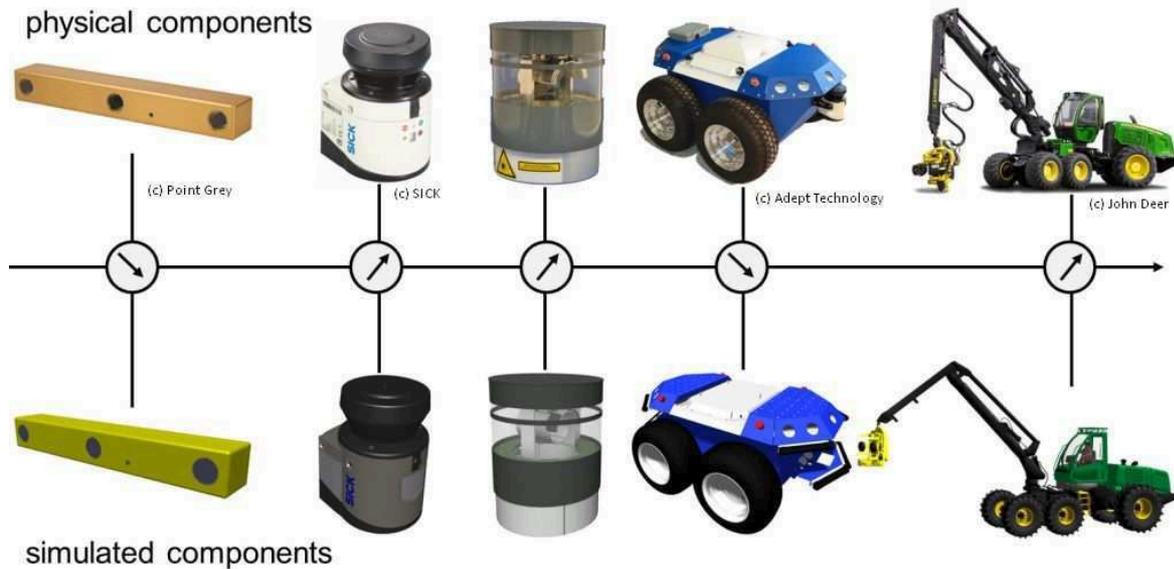


ABBILDUNG 3.8: Experimentierbare Digitale Zwillinge und ihre realen Gegenstücke.

Prozesse zu einem experimentierbaren Modell. Mit dem EDZ ist es möglich durch Experimente in der Simulation zu Erkenntnissen zu gelangen, die auf die Wirklichkeit übertragbar sind [166]. Bezogen auf Abbildung 3.7 ist der EDZ die Simulation des DPS und das Virtuelle Testbed ist Laufzeitumgebung des EDZ.

### 3.3.2 Simulationsgestützte Entwicklung

Die Entwicklung neuer Methoden und Algorithmen in der mobilen Robotik ist durch die Komplexität der Aufgaben und die Vielzahl an beteiligten Komponenten äußerst anspruchsvoll.

Hier setzt das Konzept des Virtuellen Testbeds an. Es erlaubt eine realitätsnahe und auf Umwelteinflüsse reagierende übergreifende Simulation eines kompletten Systems in komplexen Szenarien. Durch vielfältige Schnittstellen können komplexe Detailsimulationen einzelner Komponenten in die Gesamtsimulation integriert werden, oder in sogenannten Hardware-in-the-Loop (HiL) Szenarien, reale Teilkomponenten in Kombination mit Digitalen Zwillingen betrieben werden. Abbildung 3.8 zeigt eine Gegenüberstellung von simulierten und realen Komponenten, die in Hybriden Testbeds ausgetauscht werden können.

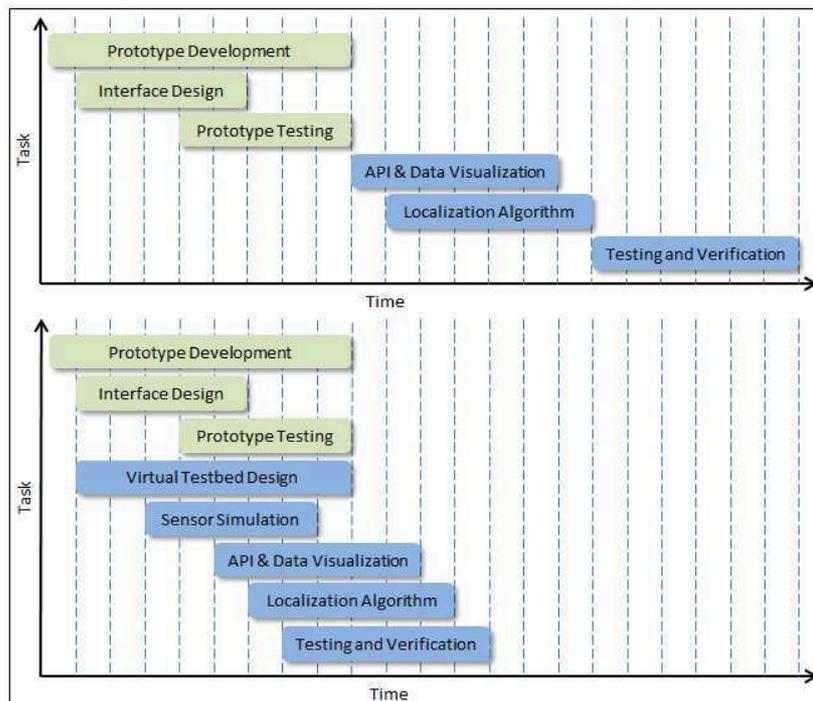


ABBILDUNG 3.9: Simulationsgestützte Entwicklung reduziert die Entwicklungszeit im Vergleich zu Entwicklungen nach dem Wasserfallmodell. Bildquelle: [9]

Durch die frühzeitigen Tests und die virtuelle Inbetriebnahme einzelner Komponenten in der Simulation reduziert sich die Entwicklungszeit multidisziplinärer Systeme im Vergleich zu Entwicklungen nach dem Wasserfallmodell erheblich. Abbildung 3.9 zeigt die Beschleunigung des Entwicklungsprozesses anhand eines Beispiels. Das Virtuelle Testbed ermöglicht die Entwicklung und das Testen der Softwarekomponenten unabhängig von der Hardwareentwicklung. Dazu werden die Algorithmen als Plugins des Simulationssystems entwickelt und vom Simulationssystem ausgeführt. An Stelle der realen Hardware kann dann mit den simulierten Komponenten im Virtuellen Testbed gearbeitet werden. Auf diese Weise können sequentielle Entwicklungsprozesse parallelisiert und dadurch erheblich beschleunigt werden.

### 3.3.3 Simulationsgestützte Steuerung

Bisher wurde das Simulationssystem lediglich als Betriebssystem für die Virtuellen Testbeds und die Digitalen Zwillinge betrachtet. Das oben



ABBILDUNG 3.10: Die mobile Roboterplattform Seekur Jr™ in verschiedenen Testumgebungen

beschriebene Konzept der Hybriden Testbeds lässt sich jedoch nur dann umsetzen, wenn neben den virtuellen auch die realen Komponenten angesprochen und gesteuert werden können. Dies ist nur dann möglich, wenn für beides dieselben Schnittstellen bereitgestellt werden. In VEROSIM wird dies durch ein generisches Input-Output-Framework (IO) realisiert, das für den Datenaustausch zwischen simulierten und verbundenen realen Komponenten genutzt wird. Es basiert auf der VEROSIM Active Simulation Database und nutzt insbesondere das Metainformationssystem zur korrekten Behandlung von unterschiedlichen Datentypen. Details zum IO-Framework sind nachzulesen in [9]. Sind am Ende des Entwicklungsprozesses alle Hardware-Komponenten fertiggestellt, erlaubt die Nutzung des generischen Schnittstellenkonzepts eine vollständige Verschiebung der Anwendung aus der Simulation hin zum realen System. Das Simulationssystem dient weiterhin als Laufzeitsystem für die als Plugins realisierten Algorithmen. Durch die ebenfalls im Simulationssystem realisierte Hardwareansteuerung werden nun aber nicht mehr die Digitalen Zwillinge angesteuert, sondern die reale Hardware. Da die Steuerung aus dem Simulationssystem heraus erfolgt, spricht man hier von der simulationsgestützten Steuerung.

### 3.4 Die mobile Roboterplattform SeekurJr™

Die Lokalisierung mobiler Systeme ist eine Kernkomponente des in dieser Arbeit vorgestellten Konzepts. Daher ist eine mobile Plattform ideal um die Tauglichkeit der jeweiligen Umsetzung in den einzelnen Anwendungen demonstrieren zu können. Für die meisten der in Kapitel 7 vorgestellten Anwendungen wird der mobile Roboter „Seekur Jr“ der



ABBILDUNG 3.11: Anwendung der simulationsbasierten Steuerung und Evaluierung mit dem mobilen Roboter Seekur Jr.

Firma *Omron Adept MobileRobots, LLC*. verwendet (Abbildung 3.10). Er verfügt über eine ausreichende Belastbarkeit um auch schweres Equipment wie Aufbauten und Sensoren zu transportieren, und erlaubt über ein „Power Distribution Board (PDB)“ die interne Spannungsversorgung des Roboters für externe Verbraucher (z.B. Sensoren) zu nutzen. Der Roboter verfügte bereits über einen internen Rechner, der für die Verarbeitung anspruchsvoller sensordatenverarbeitender Algorithmen durch ein anderes Mainboard (Spectra-Board LS-574TXD) mit Intel Core i7 CPU (i7-3610QE, 2,3GHz) und 8GB Arbeitsspeicher ersetzt wurde. Zum Antrieb verfügt der Roboter über vier große, über einen Differentialantrieb (differential drive) angetriebene, Räder. Die Low-Level-Ansteuerung der einzelnen Räder erfolgt intern über einen vom Hersteller programmierten Mikrocontoller, der zwei einfache Steuerungsbefehle (Rotation, Translation) entgegennimmt. Durch diese Ansteuerung ist es dem Roboter möglich auf der Stelle zu drehen (Rotation), vorwärts und rückwärts zu fahren (Translation) oder beides zu kombinieren um Kurven mit beliebig großem Radius fahren zu können.

Parallel zur Hardwareplattform existiert ein Digitaler Zwilling des mobilen Roboters im Simulationssystem VEROSIM, der über dieselben Schnittstellen angesprochen und gesteuert werden kann, wie die reale

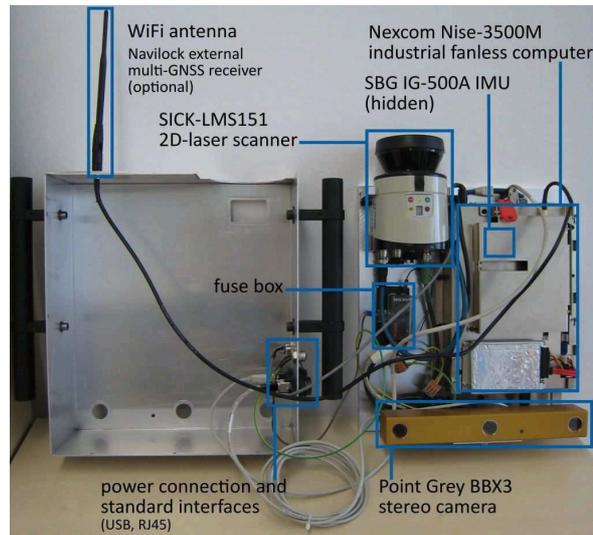


ABBILDUNG 3.12: Komponenten des ersten SELMA-Prototypen

Hardware. Damit können alle Tests zunächst in einer vollständig simulierten Umgebung durchgeführt werden und anschließend der Roboter aus dem Simulationssystem gesteuert werden, das auf dem Bordcomputer des Roboters ausgeführt wird. Darüber hinaus lassen sich alle Systemzustände des realen Systems auf die Digitalen Zwillinge übertragen, sodass die Ergebnisse real durchgeführter Experimente in der Simulationsumgebung parallel evaluiert werden können (Abbildung 3.11)

### 3.5 Die Umgebungserfassungsplattform „SELMA“

Die Semantic Environment perception, Localization and Mapping unit oder die SELMA-Einheit ist eine Plattform, die aus einem Verbund mehrerer Sensoren zur Umgebungserfassung und Lokalisierung, sowie einem Rechner besteht. Ähnlich zur mobilen Plattform SeekurJr wird auf SELMA eine Instanz von VEROSIM ausgeführt, das gleichermaßen dafür benutzt wird, die Sensoren anzusteuern und die Algorithmen zur semantischen Umgebungserfassung, Lokalisierung und Umweltmodellierung auszuführen.

In der ersten Umsetzung der SELMA-Einheit (Abbildung 3.13) wurde der Rechner fest in das Gehäuse verbaut, um eine Einheit zu erhalten, die



ABBILDUNG 3.13: SELMA-Einheit in der ersten prototypischen Version.

lediglich eine externe Stromversorgung benötigt. Als Sensoren wurden ein 2D-Laserscanner, eine Stereokamera, eine IMU und ein GPS-Receiver verwendet. Eine genaue Übersicht der verbauten Komponenten und ihre Anordnung im Gehäuse gibt Abbildung 3.12. Der verbaute Rechner ist ein lüfterloser Industrie-PC der über alle notwendigen Hardware-Schnittstellen zum Anschließen und Steuern der Sensoren verfügt. Mit dieser Zusammenstellung ist die Einheit entsprechend schwer (12,3kg inklusive Akkus) und kann nur für kurze Zeit von Hand getragen werden. Alternativ kann die Einheit von der Roboterplattform SeekurJr getragen und mit Strom versorgt werden (Abbildung 3.13 rechts). Nachteilig hat sich bei dieser Variante das nicht vorhandene Display herausgestellt, sodass meist ein weiterer Laptop zur Anzeige der Ergebnisse verwendet werden muss.

In der zweiten Umsetzung der Sensorplattform SELMA wurde, entsprechend des in VEROSIM verwurzelten Paradigmas der Modularisierung, die Sensoreinheit von der Verarbeitungseinheit entkoppelt. Die Sensoreinheit wurde stark miniaturisiert und als Sensoren sind nun keine Industriekomponenten sondern mobile Alternativen integriert worden. Als Rechner wird ein Tablet-PC<sup>3</sup> verwendet, das auf einer optionalen Stütze an der Sensoreinheit aufgelegt werden kann. Durch das stark reduzierte Gewicht (nur 2,3kg inklusive Akkus, d.h. eine Gewichtsreduktion um über 80% im Vergleich zur ersten Version) und die Möglichkeit zur Verwendung eines Tragegeschirrs, kann die Sensorplattform inklusive des Rechners leicht über einige Zeit getragen werden, ohne den Benutzer

<sup>3</sup>Für den Außeneinsatz im Wald wird ein stoß- und spritzwassergeschütztes Toughpad von Panasonic verwendet. Für Demonstrationszwecke mit visueller Ausgabe wird alternativ ein Microsoft Surface Pro verwendet, das über leistungstärkere CPU und GPU verfügt.



ABBILDUNG 3.14: SELMA-Einheit in der zweiten, miniaturisier-  
ten Version mit externem Tablet-PC.

stark in seiner Bewegungsfreiheit einzuschränken. Damit ist sie ideal auch zur Verwendung in unwegsamem Gelände, wie z.B. im Wald, geeignet. Die zweite Version ist in Abbildung 3.14 zu sehen. Insbesondere sind hier die deutlich kleineren Dimensionen der Einheit zu sehen. Der externe Tablet-PC kann beliebig ausgetauscht werden und muss lediglich über eine USB-3.0-Schnittstelle mit der SELMA-Einheit verbunden werden.

### 3.6 Das VEROSIM<sup>TM</sup>-Visual4-Konzept

Aufbauend auf dem grundlegenden Konzept der vorliegenden Arbeit und den Möglichkeiten, die das Virtuelle Testbed in Verbindung mit den Paradigmen der simulationsgestützten Entwicklung und Steuerung bietet, wurde im Rahmen des European Satellite Navigation Contest (ES-NC 2013) ein weiterführendes Konzept entwickelt, das die Fähigkeiten eines mobilen Systems über die Umgebungserfassung, Lokalisierung und Umweltmodellierung hinaus um Aspekte zur simulationsgestützten Planung und autonomen Steuerung erweitert. Diese Idee wurde in der Regionalausscheidung des ESNC-Wettbewerbs 2013 mit dem zweiten Platz ausgezeichnet<sup>4</sup>

Dieses Konzept wurde VEROSIM-Visual-4 genannt, da es grundlegend auf dem Simulationssystem VEROSIM aufbaut und zur Umgebungserfassung in erster Linie optische Sensoren verwendet, also die Umgebung visuell - in weitestem Sinne - wahrnimmt. Eine Übersicht

<sup>4</sup><http://www.agit.de/technologieregion-aachen/beitrag/artikel/pm-0114-top-technology-cluster-ttc-laedt-am-17-februar-ein-nrw-gewinner-des-european-satellite.html> (Seitenzugriff am 10.05.2017)

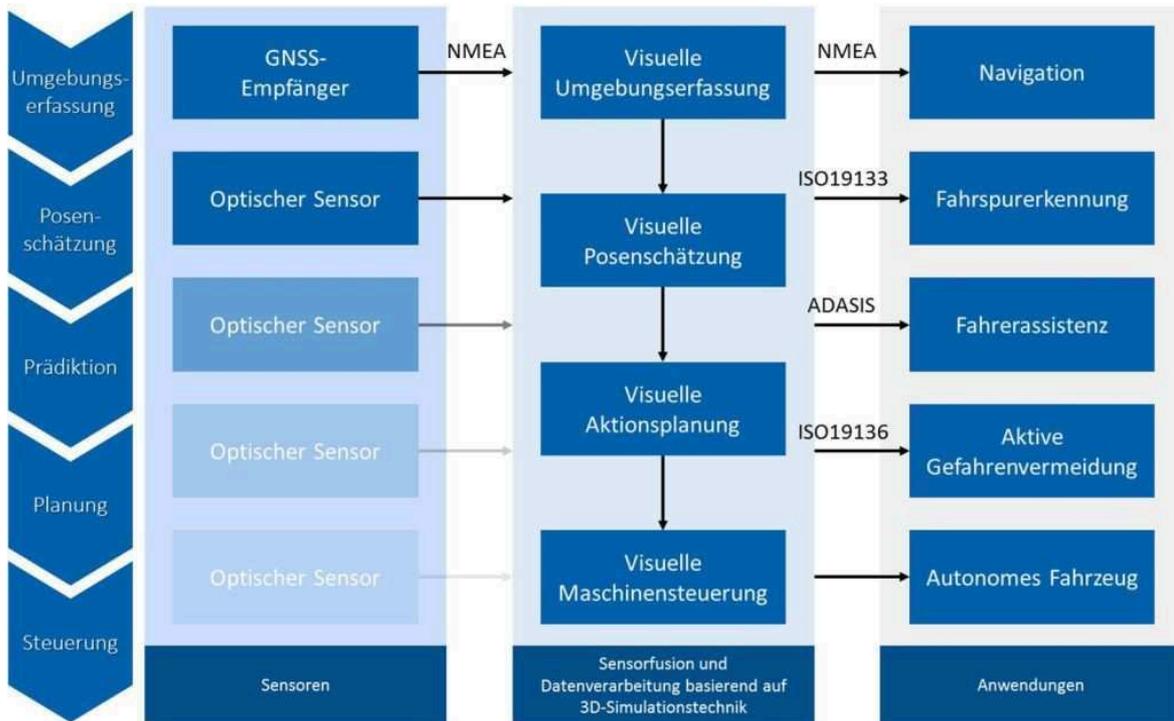


ABBILDUNG 3.15: Das VEROSIM-Visual-4 Konzept beschreibt die simulationsgestützte Fusion von Sensordaten und sensor-datenverarbeitenden Komponenten zu einem Gesamtsystem, das sukzessive auf seinen Teilergebnissen aufbaut, sodass die Anwendungen, von der Umgebungserfassung bis hin zum autonomen Fahrzeug, abgedeckt sind.

zu diesem Konzept gibt das Diagramm in Abbildung 3.15. Die Idee basiert auf dem Konzept der simulationsgestützten Entwicklung, Planung, Optimierung, und Steuerung, die im Detail in [175] beschrieben wird.

Der Grundgedanke des Konzepts umfasst zunächst die Semantische Umgebungserfassung und eine darauf aufbauende Posenschätzung bzw. Lokalisierung und gibt die Ergebnisse in etablierten Standardformaten aus, sodass bereits bestehende Endanwendungen diese Daten direkt und ohne weitere Anpassungen nutzen können. Als Beispiel kann etwa die Sensorplattform SELMA aus Abschnitt 3.5 als Lokalisierungseinheit im Wald verwendet werden und gibt die Lokalisierungsinformationen als Signal im NMEA-Format eines GNSS-Empfängers aus (siehe Abschnitt 2.1.1). Dieses Signal kann anschließend von Navigationsanwendungen verwendet werden, die gewöhnlich ihre Daten von GNSS-Empfängern

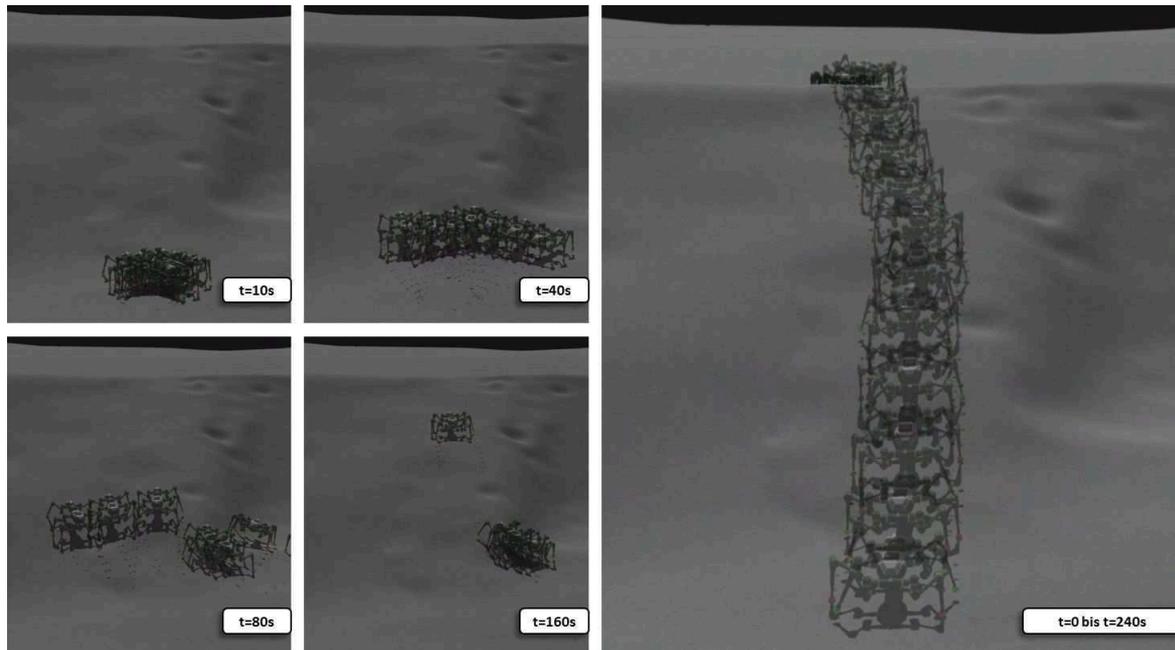


ABBILDUNG 3.16: Simulationsgestützte Prädiktion auf Basis eines Mentalen Modells. In der Simulation werden verschiedene Entscheidungen ausgeführt (vier Bilder links) und bewertet. Die Aktion, die zum Ziel führt wird ausgewählt (rechts). Bildquelle: [175].

erhalten, ohne eine eigene Umwandlung der Daten vornehmen zu müssen. Damit muss die Endanwendung nicht angepasst werden, profitiert aber direkt von der präzisen und zuverlässigen Lokalisierung im Wald, wo das GNSS-Signal aufgrund von Störungen oft nur sehr ungenau ist. Die Ergebnisse der Umgebungserfassung und Lokalisierung werden gemäß des Konzepts der vorliegenden Arbeit in ein gemeinsames Umweltmodell gespeichert.

Dieses Umweltmodell dient dem mobilen System im VEROSIM-Visual4 Ansatz als sogenanntes Mentales Modell, also als Grundlage für eine simulationsgestützte Prädiktion. Dadurch können in der Simulation verschiedene alternative Aktionen durchgeführt und anschließend der jeweilige Ausgang bewertet werden.

Abbildung 3.16 verdeutlicht diese Entscheidungsfindung in einem einfachen Beispiel: Ein Roboter soll sich auf eine Position bewegen und muss dafür einen Hang mit starker Neigung überwinden. In der Simulation werden dazu verschiedene Startparameter des Roboters getestet. Die vier Teilbilder auf der linken Seite von Abbildung 3.16 zeigen die

Ergebnisse der alternativen Bewegungen zu verschiedenen Zeitpunkten. Die Alternative, die in der Simulation zum Erfolg führt, wird auf der rechten Seite der Abbildung dargestellt.

Auf Basis der in der Simulation am erfolgreichsten Parametrierung kann nun eine Planung durchgeführt werden, die anschließend automatisiert, und begleitet von erneuter bzw. weitergeführter Umgebungserfassung und Lokalisierung, ausgeführt wird, indem nun nicht der Digitale Zwilling im Virtuellen Testbed angesteuert wird, sondern der reale Roboter in der realen Umgebung..

Parallel zur Ausführung auf der realen Hardware ist es möglich die Simulation mitlaufen zu lassen und bei zu großen Abweichungen zwischen der Lokalisierung des realen Systems und dem geplanten Pfad den Prädiktionsschritt zu wiederholen und anschließend neu zu planen.

Das VEROSIM-Visual4-Konzept basiert grundlegend auf den im Folgenden ausführlich behandelten Themen: Semantische Umgebungserfassung, Lokalisierung und Umweltmodellierung.

## Kapitel 4

# Semantische Umgebungserfassung

Die Semantische Umgebungserfassung umfasst die *Detektion von Strukturen* in den Daten exterozeptiver Sensoren, die darauf aufbauende *Erkennung von Objekten* und die anschließende *semantische Beschreibung* dieser Objekte. Im Kontext der Lokalisierung werden diese Objekte als semantische Landmarken bezeichnet, um zu verdeutlichen, dass diese Objekte die *speziellen Anforderungen der Lokalisierung* erfüllen. In diesem Kapitel werden sowohl die Fragen der Anforderungen beantwortet als auch Methoden vorgestellt die Lösungen der Aufgaben in der semantischen Umgebungserfassung beschreiben.

Die Umgebungserfassung dient dazu, die Daten der angebundenen Sensoren zu verarbeiten und relevante Informationen daraus zu extrahieren. Welche Informationen relevant sind, ist in erster Linie von der nachgelagerten Anwendungen abhängig, jedoch welche Informationen überhaupt aus den Sensordaten extrahiert werden können ist abhängig vom jeweilig eingesetzten Sensortyp. Daraus ergibt sich eine indirekte Abhängigkeit zwischen Sensoren und Anwendungsfällen, der im Wesentlichen bereits bei der Umgebungserfassung begegnet werden muss. Zunächst stellt sich also die Frage, welche Informationen für nachgelagerte Anwendungen von Interesse sein, und durch welche Sensoren diese auf welche Weise gesammelt werden können. Angelehnt an die Anwendungen im Wald, bei planetaren Explorationsmissionen und bei der Indoor-Lokalisierung stehen in der vorliegenden Arbeit entsprechend Bäume, Steine und Felsen sowie künstliche Marker als Landmarken im Vordergrund.

## 4.1 Anforderungen der Lokalisierung

Der Mensch orientiert sich und interagiert mit seiner Umgebung gewöhnlich anhand von Objekten und Merkmalen, die aufgrund besonderer Eigenschaften in ihrer Umgebung hervorstechen und sich daher gut beschreiben und merken lassen [177]. Dabei sind mit diesen Merkmalen nicht, wie etwa in der Bildverarbeitung üblich, abstrakte Merkmale im Bildraum gemeint, sondern *semantisch beschreibbare Entitäten der Umwelt*, die aufgrund von Besonderheit gut von anderen Entitäten der Umwelt unterschieden werden können. Mit anderen Worten, der Mensch beschreibt seine Umwelt selbst wohingegen in der Bildverarbeitung die projektive Abbildung der Umwelt, also etwa das Kamerabild, beschrieben wird. Jedoch nur durch die Erfassung und Beschreibung von *Objekten in der Umgebung auf der Basis von Sensordaten* anstatt die Sensordaten selbst zu beschreiben, kann die Unabhängigkeit vom eingesetzten Sensor gewährleistet werden, denn es wird ein einheitlicher Bezugsraum geschaffen, der einer späteren Sensorfusion zuvorkommt.

Im Grunde können alle Objekte mit Tiefeninformationen, die aus den Sensordaten erfasst wurden als Landmarken dienen, sofern sie über bestimmte Eigenschaften verfügen. Zu diesen bestimmten Eigenschaften gehören in erster Linie eine **eindeutig bestimmbare Position** sowie eine **zuverlässige Detektierbarkeit** der Objekte auch aus verschiedenen Perspektiven. Je exakter sich der Standort der Landmarken aus den Sensordaten bestimmen lässt, desto genauer kann sich das mobile System wiederum an diesen Landmarken lokalisieren. Daher ist bei der Wahl geeigneter Objekte deren Kompaktheit, also ein Maß über die Ausdehnung des Objekts im Verhältnis zu seiner Größe, von Bedeutung. Weit ausgedehnte und unförmige Objekte, wie etwa Büsche und Sträucher, eignen sich demnach weniger für diesen Anwendungszweck als Objekte mit einer klar abgegrenzten und ggf. auch regelmäßigen Form, wie z.B. Baumstämme, Verkehrsschilder oder künstliche Marker. Dabei muss das Objekt nicht zwingend aus allen Richtungen gleich gut erkennbar sein. Auch ein Objekt, das aus einigen wenigen Perspektiven nicht detektierbar ist, wie z.B. ein Verkehrsschild, das von der Seite nicht zweifelsfrei als solches erkannt werden kann, lässt sich dennoch als Landmarke verwenden. Wichtiger als die Frage, ob ein Objekt aus allen Richtungen detektiert werden kann, ist an dieser Stelle die Frage, ob ein Objekt eindeutig detektiert wird, also ob es aus der einen Richtung als

ein bestimmtes Objekt erkannt wird und aus einer anderen Richtung nicht als ein anderes Objekt. Solche Fehldetektionen haben einen stärkeren, negativen Einfluss auf die Lokalisierung, als fehlende Detektionen, da fehlende Detektionen nicht weiter berücksichtigt werden, fehlerhafte Detektionen aber zu falschen Paarungen und damit zu fehlerhaften Bewegungsschätzung führen können.

## 4.2 Detektion und Erkennung von Objekten

Bei der Umgebungserfassung und Modellierung wird durch die Detektion von bestimmten, im Vorfeld festgelegten, Objekten frühzeitig im Verlauf des Verarbeitungsprozesses eine Abstraktion von den verschiedenen zum Einsatz kommenden Sensortypen bewirkt. Das heißt, es wird mit der semantischen Beschreibung der Umgebung bereits die Abhängigkeit zur eingesetzten Vermessungshardware aufgelöst, sodass alle folgenden Verarbeitungskomponenten auf einer gemeinsamen und sensorunabhängigen Umweltbeschreibung aufbauen. Für diesen Zweck bedarf es geeigneter Algorithmen, die aus den jeweiligen Sensordaten räumlich beschreibbare Objekte erfassen, welche die oben aufgeführten Anforderungen der Lokalisierung erfüllen. Als Eingabe erhalten diese Algorithmen die jeweiligen Messdaten der angewendeten Sensoren. Die semantische Umgebungserfassung stellt damit die Schnittstelle zwischen den verschiedenen Sensortypen zur sensorunabhängigen Umweltmodellierung dar, d.h. der jeweilige Sensorraum wird verlassen und für alle folgenden Arbeitsschritte wird der dreidimensionale Bezugsraum der Umgebung verwendet, in dem die Objekte beschrieben werden. Dafür müssen für jede verwendete Sensorklasse eigene Algorithmen zur Objektdetektion und Beschreibung entwickelt und implementiert werden.

Im Automotive-Bereich sind derartige Detektions- und Erkennungsalgorithmen, insbesondere auf dem Forschungsgebiet zum autonomen Fahren Gegenstand aktueller Forschungsarbeiten (z.B. [178]). Zur Lokalisierung ist in den meisten Fällen eine Menge von 1-5 detektier- und beschreibbaren Objekten ausreichend. Dabei sollten Objekte, die von verschiedenen Typen von Sensoren zu erfassen sind, bevorzugt ausgewählt werden, da sie eine Fusion verschiedener Detektionsarten erlauben, was in der Regel zu zuverlässigeren Detektionsraten führt. Dabei können die einzelnen Detektionsalgorithmen auch sehr simpel gehalten werden,

wie z.B. eine einfache Clusterdetektion in 2d-Laserscanner-Profilen zur Erfassung von Baumpositionen im Anwendungsszenario „Wald“.

Demnach muss für jeden verwendeten Sensortyp ein Verfahren zur Detektion der entsprechenden Objekte entwickelt werden. Auch die zu erfassenden Objekte selbst müssen vorher festgelegt und im Rahmen der Fähigkeiten des jeweiligen Sensors, das heißt anhand der vom Sensor erfassbaren Eigenschaften, beschrieben werden.

Grundlage für Lokalisierung, Kartenerzeugung, Umgebungsanalyse und alle darauf aufbauenden Anwendungen ist die geeignete Abbildung der realen Umgebung in ein entsprechendes Modell. Ein solches Modell setzt sich idealerweise aus einzelnen Komponenten oder Objekten zusammen, die zum einen über eine eigene Pose verfügen und zum anderen Attribute besitzen, die sie beschreiben. Die Pose der Objekte erlaubt eine Lokalisierung und ihre Beschreibung ermöglicht eine Wiedererkennung und ggf. auch eine Analyse der Umgebung. Um ein solches Umgebungsmodell aufbauen zu können, bedarf es geeigneter Algorithmen, die die beschriebenen Komponenten detektieren und attribuieren können. In diesem Abschnitt werden beispielhaft verschiedene dieser Algorithmen vorgestellt.

### 4.2.1 Allgemeine Strukturmerkmale

Bevor komplexe Objekte aus aufgenommenen Sensordaten detektiert werden können, werden in der Regel allgemeine Strukturinformationen extrahiert. Das sind einfache geometrische Merkmale der Umgebung wie Ecken, Kanten und Flächensegmente. In vielen wissenschaftlichen Arbeiten werden diese Informationen bereits direkt für die Lokalisierung verwendet. Im Gegensatz dazu, sollen in der vorliegenden Arbeit diese Strukturmerkmale dazu genutzt werden, Objekte in den Sensordaten zu lokalisieren und extrahieren zu können, bevor sie später als semantische Objekte zusammengefasst und beschrieben werden.

In 2D-Laserscannerdaten, also in meist horizontalen Scanprofilen wie in Abbildung 4.1 zu sehen, sind zusammenhängende Messpunkte für weitere Untersuchungen von Interesse. Aus diesem Grund soll an dieser Stelle die Detektion von Clustern aus zusammenhängenden Scanpunkten im Scanprofil (grüne Linien im Overlay von Abbildung 4.1) näher betrachtet werden. Ein 2D-Scanprofil besteht aus Metainformationen, wie Winkelauflösung, Öffnungswinkel und Offset des Laserscanners und

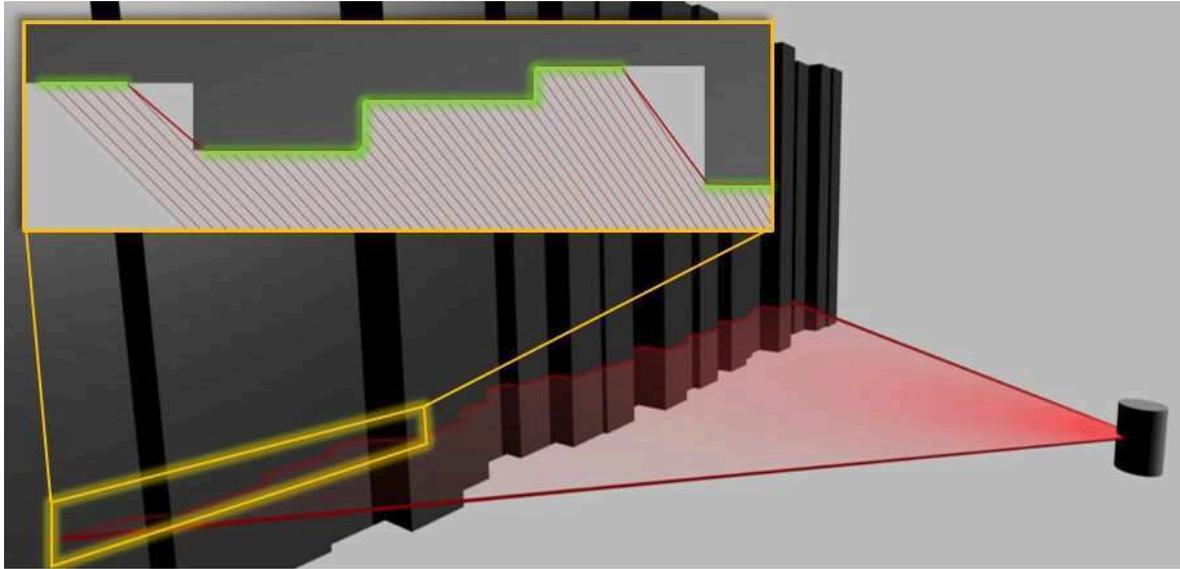


ABBILDUNG 4.1: Umgebungserfassung mit einem 2D-Laserscanner. Das Overlay zeigt einen Ausschnitt des Scanprofils.

den eigentlichen Messwerten. Die Messwerte sind ein Vektor aufeinander folgender Tiefenwerte, die den einzelnen Messstrahlen im Abstand der Winkelauflösung  $\phi$  des Laserscanners entsprechen. Eine naive Herangehensweise zur Detektion von zusammenhängenden Clustern wäre die Suche nach Tiefenwerten, die sich nur um einen erlaubten Schwellwert unterscheiden dürfen. Dabei ist zu beachten, dass sich bei der Betrachtung einer ebenen Fläche mit steigendem Betrachtungswinkel die Abstände der Messpunkte auf dieser Fläche nicht linear verändern, d.h. wie in Abbildung 4.2 zu sehen, steigen die Messpunktabstände  $r_\phi(i)$  zum Rand exponentiell an:

$$r_\phi(i) = \tan(i\phi) - \tan((i-1)\phi). \quad (4.1)$$

Dabei indiziert  $i$  die Strahlen ausgehend von der Sichtrichtung des Sensors, wie in Abbildung 4.2 dargestellt. Das Diagramm auf der rechten Seite von Abbildung 4.2 zeigt den Verlauf der Abstände bezüglich des Betrachtungswinkels und einer Winkelauflösung von  $\phi = 1 \text{ deg}$ . Der

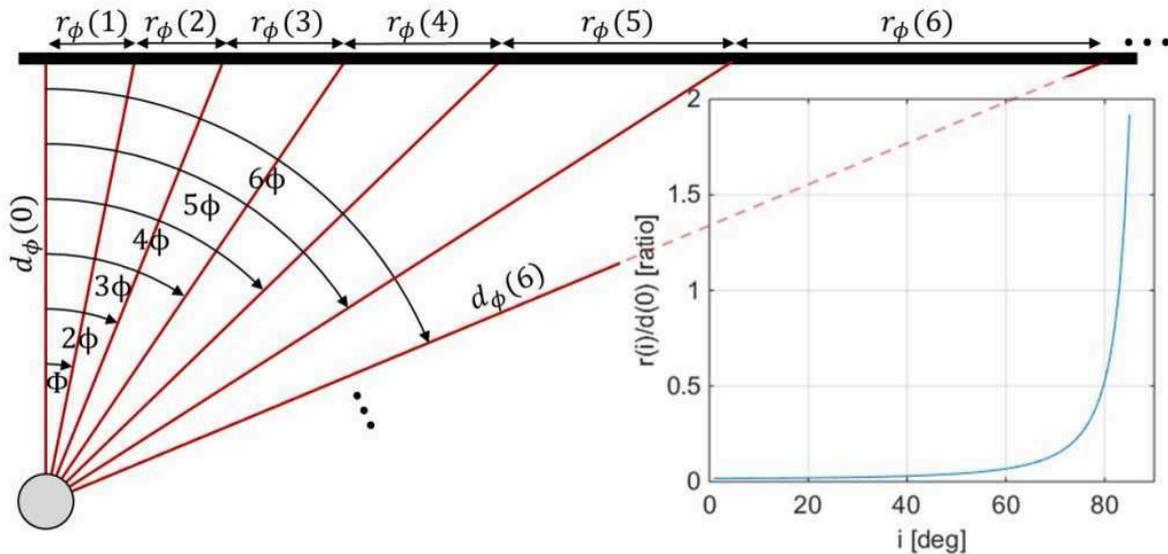


ABBILDUNG 4.2: Zusammenhang zwischen der Differenz zweier benachbarter Tiefenwerte und dem Betrachtungswinkel

Verlauf der Tiefenwerte  $d_\phi$  entspricht der Kosinusfunktion:

$$d_\phi(i) = \frac{d_\phi(0)}{\cos(i\phi)}. \quad (4.2)$$

Zur allgemeinen Betrachtung sei  $d_\phi(0) = 1$  angenommen, damit gibt  $d_\phi$  den Faktor an, um den der jeweilige Tiefenwert größer ist, als der Abstand zwischen Sensor und Ebene:

Damit Strukturen auch in Randbereichen korrekt erkannt werden können, benötigt es eine geometrische Herangehensweise, die geometrische Hypothesen aufstellt und diese anhand der Messwerte beurteilt. Als Beispiel soll an dieser Stelle die Gegenüberstellung des naiven Ansatzes und der geometrischen Herangehensweise in der Skizze in Abbildung 4.3 ausreichen. Links werden, basierend auf den Abständen der Messpunkte, drei mögliche zusammenhängende Cluster gefunden, wohingegen durch die geometrische Betrachtung des Problems, d.h. hier durch die Aufstellung und Bewertung von Geradengleichungen, lassen sich Strukturen korrekt erfassen. Um die grundlegende Idee zu beschreiben, soll dieses Beispiel hier ausreichen. Eine tiefergehende und konkrete Betrachtung wird in den folgenden Abschnitten gegeben.

Sobald man in der Lage ist Struktursegmente aus einem Scanprofil

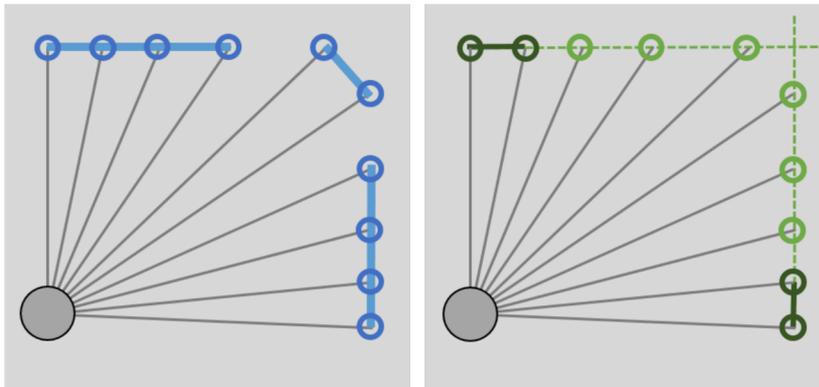


ABBILDUNG 4.3: Gegenüberstellung von Abstands- und Geometrieansatz bei der Strukturdetektion. Links: Es werden drei Cluster gefunden, rechts: es werden zwei Geraden gefunden.

zu extrahieren, können darauf aufbauend auch Ecken detektiert werden, indem man den Winkel zwischen zwei sich schneidenden Flächensegmenten bestimmt. Insbesondere bei der Indoor-Lokalisierung sind rechte Winkel und große Flächen von besonderer Bedeutung, da sie die Charakteristika von Räumen widerspiegeln. Zimmerdecken, Wände und Raumecken geben hier Information über die Geometrie eines Raumes und ermöglichen eine Lokalisierung in Gebäuden oder relativ zu Zimmerwänden.

In nicht strukturierten Umgebungen sind diese Merkmale eher selten zu finden, da beispielsweise in der Natur gewöhnlich keine geradlinigen und rechtwinkligen Strukturen zu finden sind. Nimmt man allerdings diese einfachen Strukturelemente nicht direkt als Landmarken für die Lokalisierung sondern nutzt sie in abgewandelter Form stattdessen als Grundlage für zusammengesetzte Objekte, so lassen sich damit Detektionsalgorithmen entwickeln, die auch im Outdoor-Bereich eingesetzt werden können, wie später in diesem Kapitel gezeigt wird.

### Rechnerische Genauigkeiten

Da die Qualität der Landmarkendetektion unmittelbar von der Genauigkeit der Sensordaten abhängt, soll diese zunächst näher untersucht werden. Die primären Sensoren der SELMA-Einheit (siehe Kapitel 3.5) sind ein 2D-Laserscanner und eine Stereokamera. Auch wenn sich die Aufnahmemethodik beider Sensoren deutlich unterscheiden, lassen sich

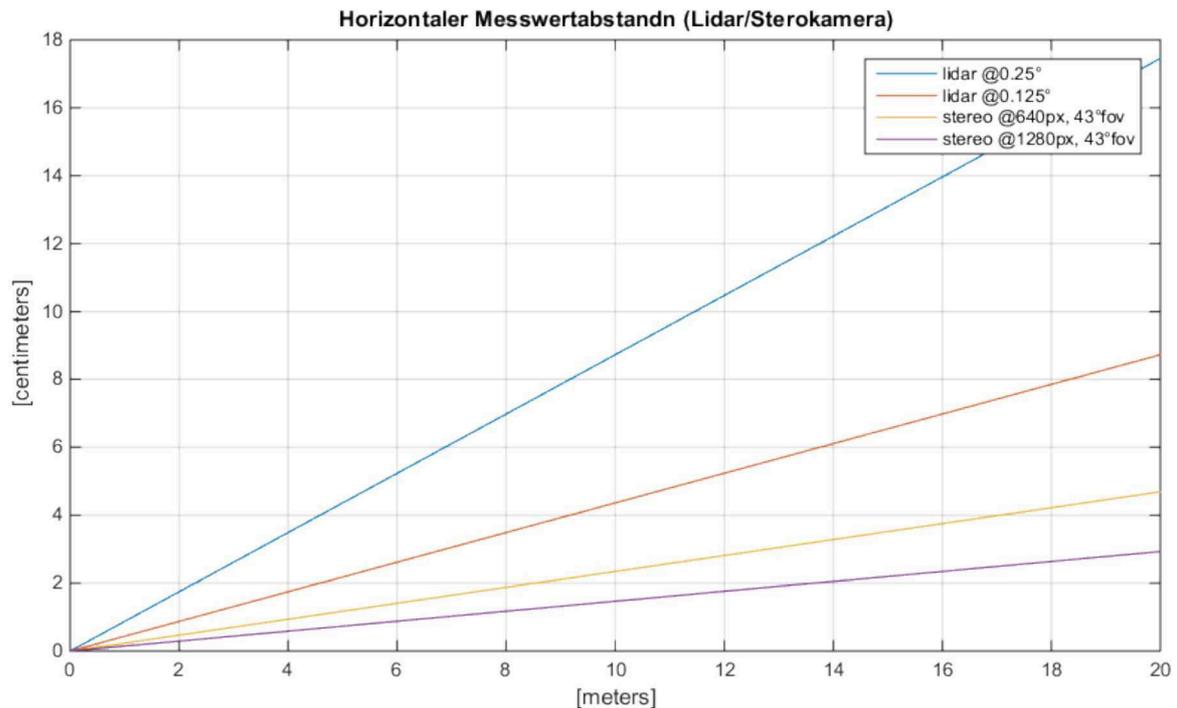


ABBILDUNG 4.4: Gegenüberstellung von Strahl- bzw. Pixelabstand zwischen 2D-Laserscanner und Stereokamera in Abhängigkeit von der Messentfernung.

zwei grundlegende Faktoren zur Bewertung der Genauigkeit identifizieren. Der erste der beiden Faktoren ist die horizontale (im Fall der Stereokamera auch die vertikale) Sensorauflösung. Sie gibt an, mit welcher Schrittweite zwei Messungen in der Horizontalen (bzw. Vertikalen) auseinander liegen und sind in Abbildung 4.4 dargestellt. Für einen 2D-Laserscanner wird, wie bereits verwendet, in der Regel die Winkelauflösung, also der Winkel zwischen zwei Messstrahlen, angegeben und bei der Stereokamera gibt die Bildauflösung die Anzahl der Pixel in Breite und Höhe des Bildes an, was zusammen mit dem Öffnungswinkel des Objektivs eine vergleichbare Aussage darstellt. Aus diesen Informationen lässt sich, in Abhängigkeit zur Entfernung, der jeweilige Messwertabstand auf einem betrachteten Objekt angeben.

Aus Abbildung 4.4 wird deutlich, dass eine Stereokamera mit verhältnismäßig geringer VGA-Auflösung von 640x480 Pixeln und einem horizontalem Öffnungswinkel von üblichen 43 deg bereits eine erheblich größere horizontale Auflösung besitzt, als ein 2D-Laserscanner mit einer ebenfalls häufigen Winkelauflösung von 0,25 deg. Anders sieht es bei der

Tiefenauflösung, also der Genauigkeit in Blick- bzw. Messstrahlrichtung, aus. Wo ein Laserscanner eine nahezu konstante Tiefenauflösung von, je nach verwendeter Wellenlänge und Lichtintensität, wenigen Millimetern oder sogar nur Bruchteilen von Millimetern aufweist, nimmt die Tiefenauflösung einer Stereokamera aufgrund der eingesetzten Stereoskopie zur Tiefenberechnung, mit der Entfernung ab. Abbildung 4.5 zeigt die Abhängigkeit der Tiefenauflösung einer Stereokamera zur Entfernung des betrachteten Objekts auf. Deutlich wird hier, dass es sich nicht um eine lineare Abnahme handelt sondern um eine quadratische nach der Formel

$$z_{acc} = \frac{r_h f b}{d^2}, \quad (4.3)$$

in Abhängigkeit von der horizontalen Bildauflösung  $r_h$ , der Brennweite  $f$  des Objektivs (und damit dem horizontalen Öffnungswinkel), der Baseline  $b$  und der Disparität  $d$  in der Stereoaufnahme.

### 4.2.2 Erkennung von Bäumen

Für die Anwendung im Wald, sind Bäume hervorragende Landmarken für eine zuverlässige Lokalisierung. In der Regel besitzen sie eine dominante Ausrichtung, sind diskret zu beschreiben, d.h. man kann bis auf seltene Ausnahmen sagen, wo ein Baum anfängt und aufhört, ohne dass er sich mit einem weiteren Baum überschneidet. Meistens existieren ausreichend viele Bäume um permanent eine gewisse Anzahl sehen zu können und sie bewegen sich nicht, was die Annahme einer statischen Szene erlaubt.

Basierend auf den Betrachtungen allgemeiner Strukturlandmarken, spiegeln sich Bäume in einem horizontalen 2D-Laserscan als Cluster mehrerer zusammenhängender Messpunkten wider. Durch den kreisförmigen Querschnitt muss, anders als im Beispiel oben, die Orientierung der Bäume relativ zum Scanner für die Clustersuche nicht berücksichtigt werden. Allerdings reicht auch hier die Suche zusammenhängender Messpunkte nicht aus, denn gerade im Outdoor-Bereich sind diese zusammenhängenden Strukturen sehr häufig, da auch Unterholz, größere Steine, Bodenerhebungen und vielerlei weiteres im Scanbild diese Eigenschaft aufweisen.

Es werden weitere Überlegungen benötigt, um aus den erfassten Clustern, Bäume vom Hintergrund und von anderen Objekten unterscheiden

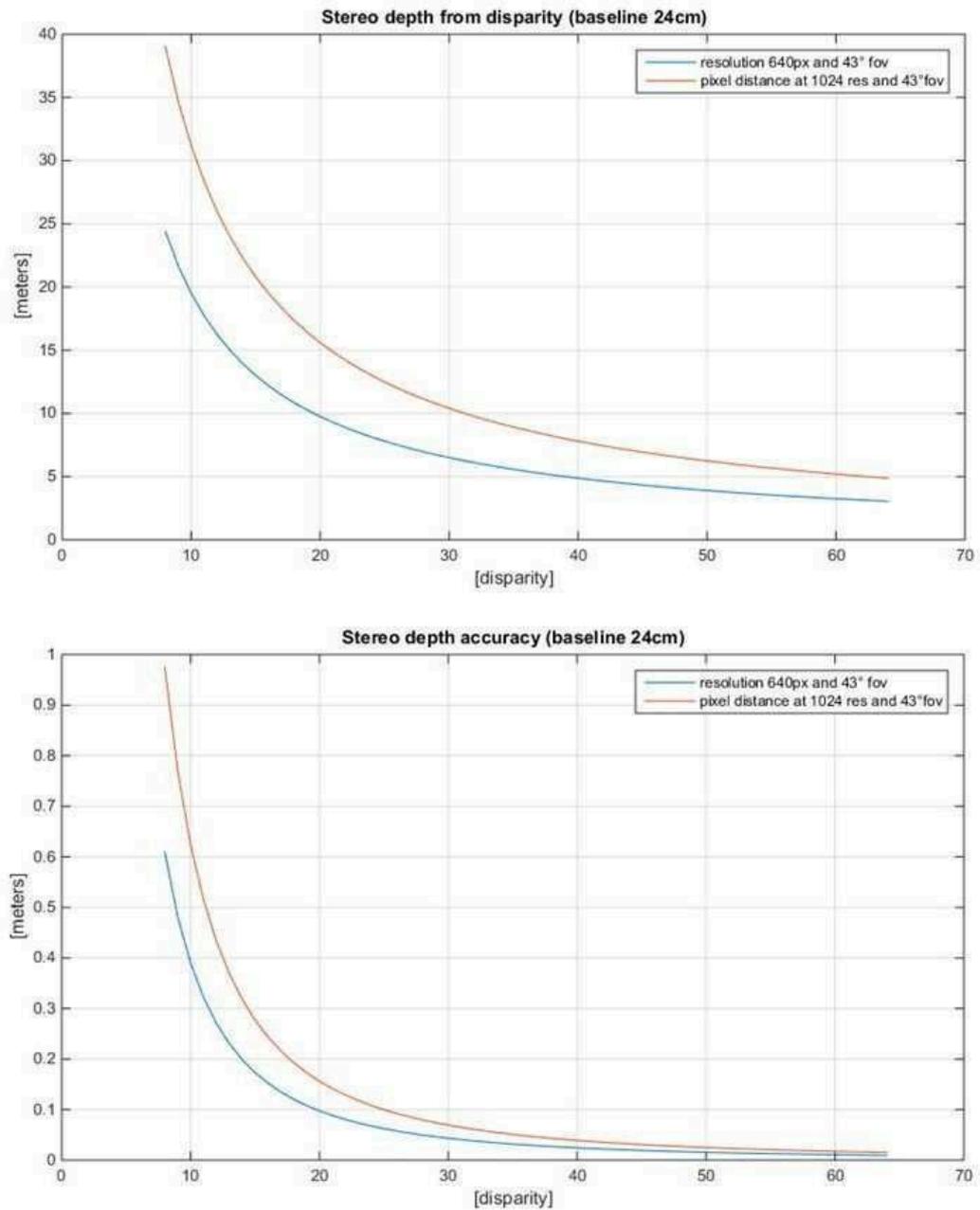


ABBILDUNG 4.5: Analyse der Tiefenberechnung und Tiefengenauigkeit bei Stereokameras

zu können. Daher werden die erkannten und extrahierten Cluster in einem ersten Schritt danach untersucht, ob sich die einzelnen Messpunkte ansatzweise auf einem kreisförmigen Bogen befinden. Diese Analyse lässt sich auf verschiedene Weisen durchführen und liefert damit auch

unterschiedliche Ergebnisse in Bezug auf Genauigkeit und Zuverlässigkeit.

In [179] wird für diesen Zweck ein Template-Matching-Verfahren vorgestellt, das ohne eine vorausgehende Clustersuche auskommt. Zur Detektion von Bäumen mit unterschiedlichen Durchmessern wird hier ein Satz von Templates mit verschiedenen Durchmessern erzeugt. Auf einer gerasterten Karte der Scanpunkte wird anschließend die Kreuzkorrelation der Karte mit diesen Templates berechnet. Bücken schlägt dazu eine Faltung der gerasterten Karte mit den Templates vor. Dazu wird der Bereich der zu detektierbaren Durchmesser in Durchmesserklassen eingeteilt. Mit diesem Ansatz konnten bereits zuverlässige Ergebnisse erzielt werden, allerdings ist die Erzeugung der Templates und die Transformation jedes Scans in eine gerasterte Karte aufwändig.

Eine Alternative stellt die Hough-Transformation für Kreise dar. Für eine zuverlässige Detektion müssen, wie beim Ansatz aus [179], auch hier die Durchmesser der zu suchenden Kreise im Vorfeld möglichst genau vorgegeben werden. Bei homogenen Waldbeständen, wo nahezu alle Bäume das gleiche Alter und somit einen ähnlichen Durchmesser besitzen ist eine solche Vorgabe einfach zu setzen. Sind jedoch viele verschiedene Durchmesser vorhanden, muss die Detektion mehrfach hintereinander mit verschiedenen Vorgaben durchgeführt werden, oder der Radius des Kreises muss als weiterer Parameter aufgenommen werden. Dies führte im direkten Vergleich mit den anderen, hier vorgestellten Methoden zur Detektion und Durchmesserbestimmung, zu verhältnismäßig großen Fehlern (siehe auch Abschnitt 4.3.1).

Eine schnelle und zuverlässige Kreisdetektion erhält man unterdessen durch die Verwendung der allgemeinen Kreisgleichung

$$r^2 = (x - x_M)^2 + (y - y_M)^2 \quad (4.4)$$

mit Radius  $r$  und Kreismittelpunkt  $M = (x_M, y_M)$ . Diese Gleichung besitzt die drei Unbekannten  $r$ ,  $x_M$  und  $y_M$  und ist daher durch drei Messpunkte eindeutig bestimmbar. Da der Querschnitt eines Baumes keinen idealen Kreis bildet, und pro Cluster in der Regel mehr als drei Messpunkte zur Verfügung stehen, bietet es sich an, mit allen Messwerten eines Clusters ein überbestimmtes Gleichungssystem aufzustellen und die Lösung der Kreisgleichung zu finden, die den Fehler minimiert (Mean Squared Error-Ansatz MSE). An der Größe des Fehlers lässt sich

im Nachhinein ablesen, wie gut das Cluster einem Kreisabschnitt entspricht, und mittels Schwellwert entscheiden, ob das Cluster als Kandidat für eine Baumdetektion in Frage kommt oder ob es verworfen wird. Der Vorteil bei der Verwendung der allgemeinen Kreisgleichung ist, dass man im Vorfeld keine Angaben bezüglich Kreisdurchmesser und Mittelpunkt machen muss, sondern sich diese Werte direkt mit der Lösung des Gleichungssystems ergeben. Außerdem können die Messwerte direkt als Liste in kartesischer Form verarbeitet werden und müssen nicht zunächst gerastert werden. Neben der für die Lokalisierung wichtigen Position des Baumes (Mittelpunkt des Kreises) wird auf diese Weise gleichzeitig der Durchmesser des Baumes auf Messhöhe bestimmt, ein Wert der für die Forstwirtschaft von erheblicher Bedeutung ist und daher auch ein wichtiger Bestandteil des semantischen Umweltmodells darstellt. In einem nachgelagerten Kontrollschritt lassen sich Kreise mit zu geringem oder zu großem Durchmesser entfernen. Die Durchmesserbestimmung wird als Teil der Attribuierung von Landmarken in Abschnitt 4.3 noch einmal im Detail behandelt.

Neben Laserscannern lassen sich auch andere Sensoren für die Baumdetektion, sowohl alternativ als auch ergänzend, verwenden. In Kapitel 2.1 wurden dazu einige Sensoren auf ihre Anwendbarkeit für die Umgebungserfassung zum Zweck einer Lokalisierung untersucht. An dieser Stelle soll auf die Stereokamera als Sensor zur Baumdetektion näher eingegangen werden.

Für die Baumerkennung aus Stereobildern wird in der Literatur oft ein kantenbasierter Ansatz vorgeschlagen [180, 77, 181], d.h. in den Farb- oder Graustufenbildern wird nach senkrechten Kanten gesucht, die Baumgrenzen widerspiegeln. Anschließend müssen die linke und rechte Kante zusammengefügt werden, um den Baum zu segmentieren. Dieses Verfahren funktioniert jedoch nicht, wenn starke Verastungen die Außenkanten der Bäume unterbrechen und wenn Bäume sich teilweise überdecken, was die Zuordnung von linker und rechter Kante zusätzlich erschwert. Aus diesem Grund wird an dieser Stelle ein tiefenbasierter Ansatz verfolgt. Statt direkt auf den Bilddaten zu arbeiten, wird in einem ersten Schritt eine dichte Tiefenkarte mittels „Semiglobalem Blockmatching“ [182] berechnet. Zu beachten ist hier, dass das Ergebnis der stereoskopischen Analyse nach [182] keine metrische Tiefenkarte darstellt, sondern lediglich um die Disparitätskarte der stereoskopisch aufgenommenen Bildpaare, worin jeder Pixel den Wert der Verschiebung

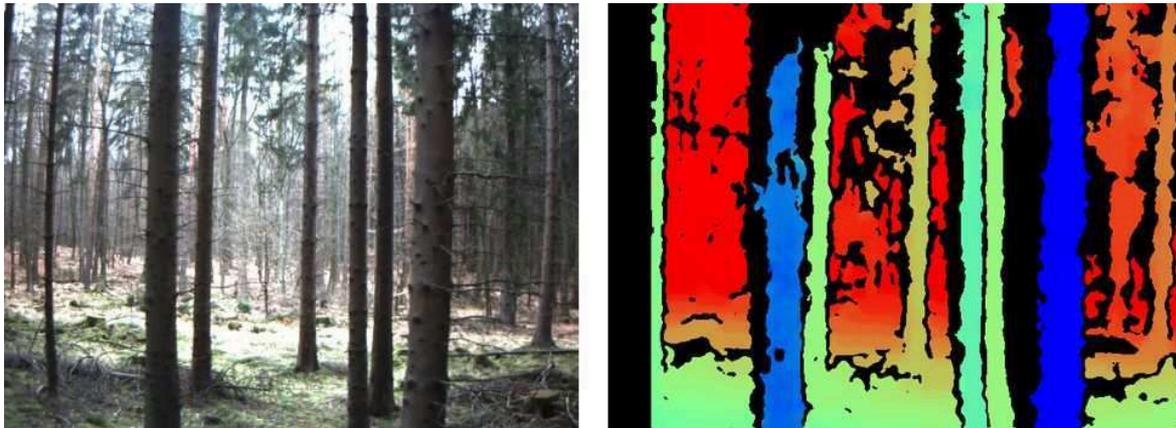


ABBILDUNG 4.6: RGB- und Tiefenbild einer Stereoaufnahme im Wald. Die Baumsegmente lassen sich im Tiefenbild leichter identifizieren als im RGB-Bild.

des entsprechenden Pixels im linken Stereoeinzelbild zum zugehörigen Pixel im rechten Stereoeinzelbild darstellt.

Erst mit Hilfe der Brennweite bzw. horizontalem „Field of View“ der Kameraobjektive und dem Abstand beider Objektive - der sogenannten „Baseline“ - lässt sich aus der Disparität die Tiefe berechnen mittels  $fb/d$ , wobei  $d$  die Disparität in Pixeln und  $b$  die Baseline in Metern darstellt. Die Brennweite  $f$  ist entweder gegeben oder berechnet sich aus dem horizontalen „field of view“  $\alpha_h$  mittels

$$f = w / (2 \tan(\alpha_h / 2)), \quad (4.5)$$

mit der Bildbreite  $w$ .

Die metrische Tiefenkarte eignet sich nun, zusammenhängende Cluster zu extrahieren. Im Unterschied zum eindimensionalen Tiefenfeld des Laserscanners, müssen hier Strukturen in der zweidimensionalen Tiefenkarte extrahiert werden. Bäume sind in einer solchen Tiefenkarten als zusammenhängende Regionen mit vertikal dominanter Ausrichtung abgebildet, wie in Abbildung 4.6 deutlich wird.

Über einen „Floodfill“-Algorithmus lassen sich nun Regionen konstanter Tiefe in der Tiefenkarte selektieren. Dazu wird ausgehend von einer oder mehreren Scanlinien, das sind in diesem Fall Bildzeilen der Tiefenkarte, nacheinander auf allen Pixeln der Scanlinie ein Floodfillalgorithmus ausgeführt, der alle benachbarten Pixel der Tiefenkarte selektiert,

die zum jeweiligen direkten Nachbarn maximal ein im Vorfeld festgelegtes  $\delta_d$  besitzen. Alle bereits selektierten Pixel der Scanlinie müssen nun nicht weiter betrachtet werden und können entsprechend übersprungen werden. Ebenso werden zu kleine Floodfill-Regionen verworfen, da sie auf inhomogene Bereiche der Tiefenkarte hindeuten, was der ursprünglichen Überlegung widerspricht, dass sich Bäume als homogene, zusammenhängende Flächen repräsentieren. Für jede selektierte Region wird anschließend die Kovarianzmatrix

$$K = 1/(n - 1) \sum_{i=1}^n (x_i - c)(x_i - c)^T \quad (4.6)$$

der enthaltenen Bildpunkte erstellt. Mittels Singulärwertzerlegung  $K = U\Sigma V^T$  erhält man die Hauptachsen der Region aus den Eigenvektoren  $U = (\vec{u}_1, \vec{u}_2, \vec{u}_3)$ . Damit lassen sich nun alle Regionen herausfiltern, deren Hauptachse nicht vertikal verlaufen (mit einer Abweichung innerhalb eines vom Benutzer angegebenen, erlaubten Bereichs, z.B.  $5^\circ$  Neigung). Aus den Singulärwerten  $\text{spur}(\Sigma) = (\sigma_1, \sigma_2, \sigma_3)$  lässt sich die Varianz der Region in Richtung der Hauptachsen ablesen. Ist die zur vertikalen Hauptachse orthogonale Varianz zu groß, d.h. ist die Region zu breit (dieser Parameter kann ebenfalls vom Benutzer festgelegt werden), kann die Region ebenfalls verworfen werden. Mit anderen Worten, werden nur vertikaldominante und schmale Regionen weiter betrachtet, da nur sie der Charakteristik eines Baumes in einer Tiefenkarte entsprechen.

Durch eine weitere Analyse der übriggebliebenen Regionen, also der potentiellen Bäume, lassen sich Position und Durchmesser aus der Tiefenkarte extrahieren. Zur Bestimmung des Durchmessers kann hier eine beliebige Höhe in der Tiefenkarte gewählt werden.

Durch die Verwendung einer sensorunabhängigen, gerasterten Tiefenkarte als Eingangsdaten für die Baumdetektion ist es möglich auch für andere depth-aware Kamerasysteme den soeben vorgestellten Detektionsalgorithmus ohne zusätzliche Anpassungen verwenden zu können. Somit erübrigt es sich speziell für Sensoren wie Time-of-Flight-Kameras oder Strukturlichtsensoren (z.B. Kinect-Sensoren) eigene Detektionsalgorithmen zu entwickeln. Dabei ist es unerheblich, ob der Sensor die Tiefenkarte unmittelbar zur Verfügung stellt, oder ob diese zunächst, wie im Fall einer Stereokamera, selbst berechnet werden muss.

Lediglich bei Verfahren, die üblicherweise keine dichten Tiefenkarten

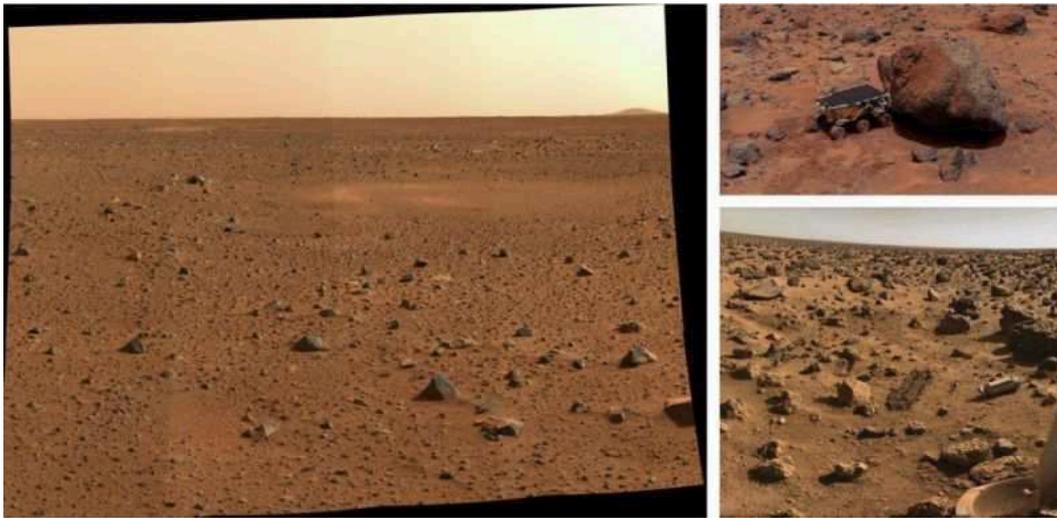


ABBILDUNG 4.7: Beispiele der Marsoberfläche aufgenommen von verschiedenen Mars-Explorationsmissionen (links: Panorama von Spirit-Landestelle; oben: Sojourner-Rover bei einer Spektrometer-Messung; unten: Aufnahme des Viking 2 Landers in der „Utopia Planitia“-Region). Bildquelle: ©NASA

als Ergebnis liefern, wie etwa allgemeine, featurebasierte Structure-from-Motion-Ansätze, ist die Anwendung dieses Detektionsverfahrens eher ungeeignet, da die Floodfill-Methode eine möglichst homogene Abdeckung an Tiefenwerten in der Karte voraussetzt.

### 4.2.3 Erkennung von Felsen und Steinen

In diesem Abschnitt wird die Detektion von Steinen als Landmarken für eine anschließende Lokalisierung behandelt. Abbildung 4.7 zeigt ein paar Beispiele für planetare Oberflächen, die keine anderen Landmarken außer Steine im unmittelbaren Umfeld aufweisen.

Die hier vorgestellten Detektionsalgorithmen wurden zwar ursprünglich für den Anwendungsfall einer extraterrestrischen Explorationsmission konzipiert und implementiert, allerdings bedeutet das nicht, dass sie ausschließlich hierin Anwendung finden können. Das Hauptproblem der Steindetektion gegenüber der im letzten Abschnitt behandelten Baumdetektion ist die Frage der Zuordnung, d.h. meist existiert keine klare Abgrenzung, was ein Stein ist, und was nicht. In der Literatur werden Steine in der mobilen Robotik oder bei autonomen Fahrzeugen meist nicht explizit sondern nur allgemein als Hindernisse behandelt. Darüber

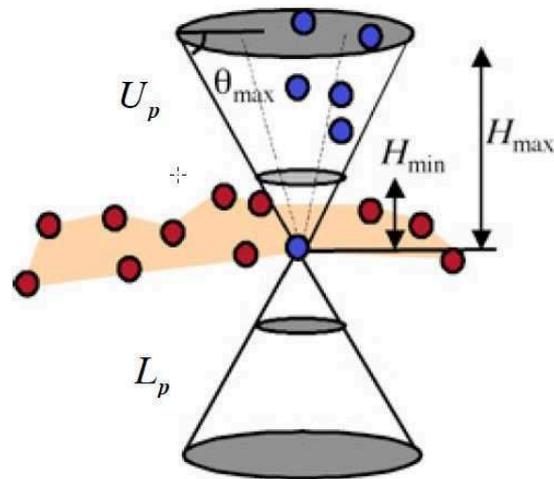


ABBILDUNG 4.8: Darstellung des Suchraums (grauer Doppelkegel) in einer Punktwolke zur Bestimmung von Hindernissen (blaue Punkte) bezüglich vertikalem Punktabstand und Steigung zwischen zwei Punkten. Bildquelle: [81].

hinaus beinhalten die Detektionsalgorithmen keine Methodik, um die Hindernisse semantisch zu klassifizieren.

Die Hinderniserkennung aus [81] lässt sich anhand von parametrierbarer Steigung und Höhenunterschieden zumindest zur Abgrenzung zwischen Hindernissen und befahrbaren Bereichen nutzen. Wenn man nun Hindernisse im Explorationsszenario als Steine, oder allgemein als Landmarken, betrachten möchte, so ließen sie sich über eben diese Attribute Steigung und Höhenunterschied, semantisch beschreiben.

Das Verfahren nutzt die bereits zur Punktwolke umgerechneten Sensordaten einer Stereokamera als Eingangsdaten. In der 3D-Punktwolke werden alle Punkte als Hindernis markiert, die zu einem weiteren Punkt der Punktwolke einen minimalen Höhenunterschied und eine Steigung überschreiten. Geometrisch lässt sich dies wie in Abbildung 4.8 dargestellt veranschaulichen: Für jeden Punkt der Punktwolke wird der über Mindesthöhe  $H_{min}$  und Steigung  $\theta_{max}$  definierte Doppelkegel gebildet und alle Punkte innerhalb des Kegels als Hindernis markiert. Durch die Angabe einer Maximalhöhe  $H_{max}$  wird der Suchraum eingegrenzt. Erhebungen im Gelände lassen sich in einem ersten Schritt durch geeignet gewähltes  $\theta_{max}$  und Eingrenzung des Verhältnisses zwischen Höhe und Breite des Hindernisses grob herausfiltern. Dieser Ansatz lässt sich

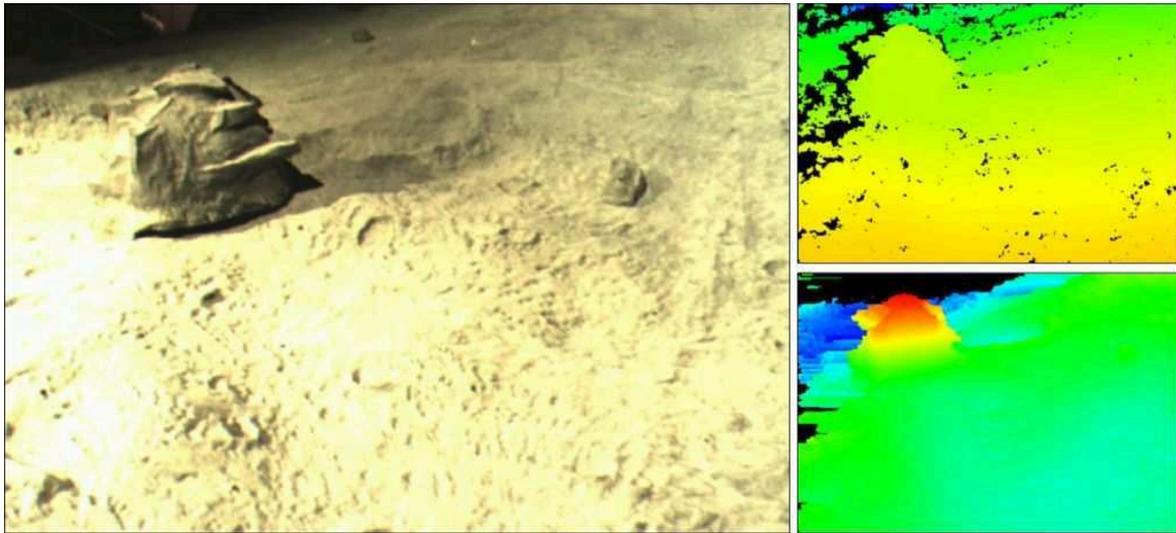


ABBILDUNG 4.9: Vergleich aus Stereobild (links) berechneter Tiefenkarte (r.o. farbkodiert von warm (nah) nach kalt (fern)) mit einer Höhenkarte, die den Abstand der Punkte zu einer berechneten Bodenebene angibt (r.u. farbkodierte Höhe von kalt nach warm).

zwar gut für die Aufgabe der Steindetektion parametrieren, d.h. dass sich mit ihrer Hilfe eine klare Abgrenzung zwischen Landmarken und restlichen Objekten schaffen lässt, jedoch ist dieses Verfahren sehr rechenaufwändig, da die Nachbarschaftsbeziehungen aller Punkte paarweise zueinander betrachtet werden müssen. Eine Eingrenzung des Suchraums durch die Angabe einer Maximalhöhe  $H_{max}$  ist zwar möglich, jedoch ist die Anzahl der zu betrachtenden Nachbarschaften bei kurzen Entfernungen zum Sensor dennoch sehr hoch, da die Auflösung hier maximal ist.

Besser geeignet ist hier ein Verfahren, das auf der Idee von [183] basiert. Anstatt Hindernisse, in diesem Fall Steine und Felsen, direkt in der vorliegenden Punktwolke zu suchen, wird zunächst aus der Punktwolke eine Höhenkarte erzeugt. Eine Tiefenkarte ist ein Bild, das an jedem Pixel an Stelle des Farbwerts des abgebildeten Punkts dessen Distanz zum Sensor enthält. Im Gegensatz dazu enthält eine Höhenkarte pro Pixel die Entfernung des jeweils zugehörigen Punkts zu einer Bodenebene. Durch die Verwendung einer höhenkodierten anstelle einer tiefenkodierten Punktwolke lassen sich Felsen auch durch das menschliche Auge deutlich einfacher erkennen, wie z.B. in Abbildung 4.9 zu sehen ist.

Durch die im Vergleich zur Punktwolke vereinfachten Darstellung als Höhenkarte lassen sich Standard-Bildverarbeitungsalgorithmen auf die Karte anwenden, die durch eine effiziente Nutzung von Bildeigenschaften optimiert sind und somit eine bessere Performanz bieten als Analyseverfahren für allgemeine Punktwolken. Es bedarf jedoch zunächst einiger Vorverarbeitungen, um die Punkte von einer Distanzkodierung zum Sensor in eine Höhenkodierung bezüglich einer Bodenebene zu transformieren.

Das vollständige Verfahren des Steindetektors lässt sich in folgende drei Schritte einteilen:

- 1 Transformation der Punktwolke in eine Höhenkarte
  - a Annäherung des Bodens über eine Referenzebene (Least-Squares-Ansatz)
  - b Affine Koordinatentransformation der 3D-Punkte (relative Höhen)
- 2 Segmentierung der Höhenkarte in Höhenklassen (Standard-Bildverarbeitung)
  - a K-Means Segmentierung
  - b Klassen gleicher relativer Höhe zusammenfassen
- 3 Steinregionen aus Höhenkarte extrahieren und attribuieren
  - a Floodfillanalyse bezüglich Klassengrenzen
  - b Beschreibung der möglichen Steinregionen (Höhe, Durchmesser, Kompaktheit)
  - c Erzeugung von Landmarkenbeobachtungen nach Auswertung der Regionsattribute

**Bodenschätzung und Höhenkartengenerierung** Im trivialen Fall einer nahezu planaren Umgebung ist diese Schätzung einfach anhand von Vorgaben bezüglich der Sensorlage<sup>1</sup> zu ersetzen wie in der Abbildung 4.10 oben links zu sehen ist. Die höhenkodierte Darstellung der

---

<sup>1</sup>Mobile Roboter sind meistens mit IMUs ausgestattet, um ihre eigene Lage messen zu können. Ist die Sensorpose bezüglich des Roboterreferenzkoordinatensystems bekannt, so lässt sich die Pose des Sensors im Raum ebenfalls bestimmen

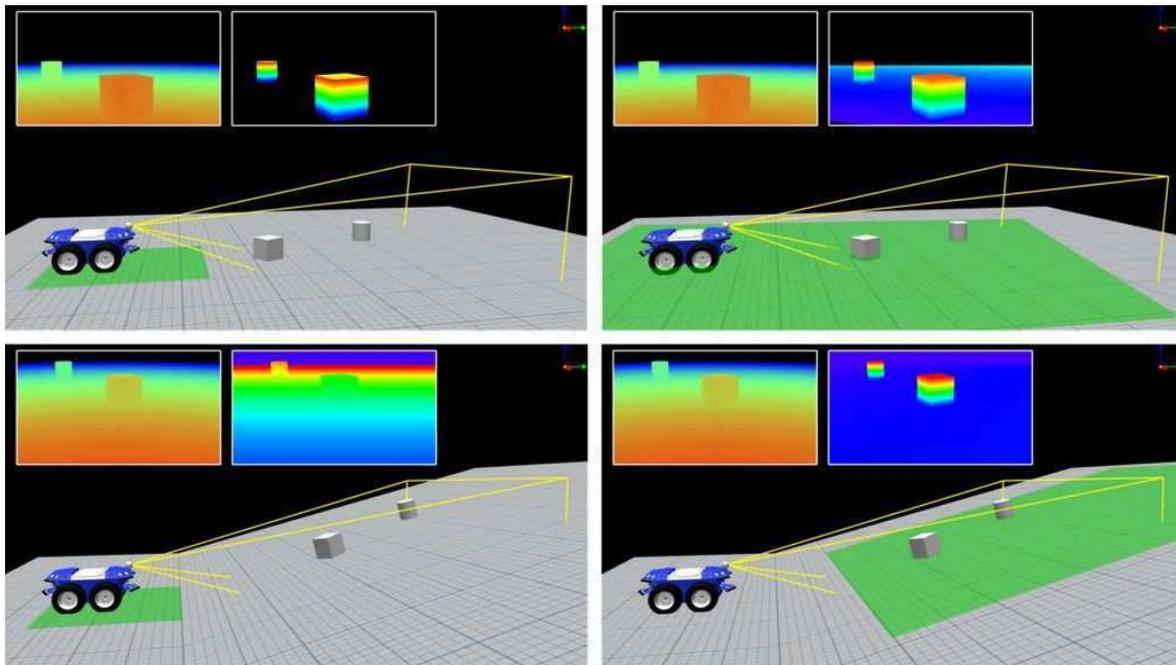


ABBILDUNG 4.10: Gegenüberstellung der Tiefenkarte einer Stereokamera (Sichtbereich als gelbes Sicht-Frustum dargestellt) distanzkodiert (jeweils linkes Overlay) und höhenkodiert (jeweils rechtes Overlay). Linke Spalte: Annahme einer konstant planaren Bodenebene zur Höhenberechnung. Rechte Spalte: Dynamische Berechnung der Bodenebene aus Stereodaten.

Sensordaten (rechtes Overlay) zeigt die deutliche Abgrenzung der Objekte zum Boden (der Boden ist hier bereits vollständig heraus-gefiltert, da er ideal mit der vorgegebenen Bodenebene (grüne Hervorhebung) übereinstimmt). Sobald der triviale Fall verlassen wird, und die Bodenebene nicht mehr zur Vorgabe passt, so sind auch die Objekte auf dem Boden nicht mehr einfach zu detektieren (Abbildung 4.10 links unten). Die Exploration auf planetaren Oberflächen setzt demnach eine Bodenebenenschätzung voraus, die auch im anspruchsvollem Gelände eine zuverlässige Extraktion von Objekten (z.B. Steine und Krater) gewährleistet, damit diese anschließend zur Lokalisierung und Kartenerzeugung genutzt werden können. Die rechte Spalte von Abbildung 4.10 zeigt die Ergebnisse der Bodenschätzung im Vergleich zur oben behandelten Bodenvorgabe. Im Fall einer ansteigenden Ebene (Abbildung 4.10 rechts unten) ist deutlich der Vorteil dieses Vorgehens im Vergleich zur Vorgabe zu erkennen. Die Ausrichtung der berechneten Bodenebene ist hier

ebenfalls jeweils in grün hervorgehoben.

Um die einzelnen Punkte der Punktwolke anstatt relativ zum Sensor bezüglich einer Bodenebene angeben zu können, muss zunächst die Bodenebene selbst bekannt sein. Dies ist insbesondere dann wichtig, wenn der Boden nicht eben ist bzw. die Blickrichtung des Sensors unbekannt und somit nicht zwingend parallel zum Boden ausgerichtet ist. Der Boden muss daher aus der Punktwolke selbst geschätzt und als Referenzebene approximiert werden. Für diesen Zweck wird ein iterativer Least-Squares-Ansatz verwendet, der ausgehend von einer initialen Ebene die Punkte der Punktwolke bezüglich ihrer Entfernung zur Ebene bewertet. Die Initialebene wird durch eine Hauptkomponentenanalyse der Punktwolke generiert<sup>2</sup>. Ausgehend von der Annahme, dass die beiden Hauptachsen dieser Analyse mit der größten Varianz der Ausdehnung der Ebene und somit die dritte Achse mit der geringsten Varianz dem Normalenvektor entsprechen, kann die Initialebene durch einmalige Anwendung der Singulärwertzerlegung (SVD) auf die Streumatrix  $K = 1/(n - 1) \sum_{i=1}^n (x_i - c)(x_i - c)^T$  der Punktwolke generiert werden. Hier bezeichnet  $c$  das arithmetische Zentrum der Punktwolke. Die Hauptachsen der Punktwolke können, wie schon beim Baumdetektor, nach der Singulärwertzerlegung  $K = U\Sigma V^T$  aus der Matrix  $U$  abgelesen werden. Die jeweils zugehörigen Varianzen entlang der Hauptachsen lassen sich der Spur von Matrix  $V$  entnehmen, anhand derer einzelne Punkte der Punktwolke als „Inlier“ oder „Outlier“ bewertet werden können. „Outlier“ sind die Punkte, die eine höhere Entfernung zur Ebene als die dreifache Standardabweichung aufweisen. Nach Entfernung dieser „Outlier“ wird die Ebene bezüglich der übrig gebliebenen „Inlier“ neu berechnet und wiederum alle Punkte bewertet. Dies wird iterativ wiederholt, bis die Anzahl der „Inlier“ stabil innerhalb eines vorgegebenen  $\delta$  bleibt. Wird dieser stabile Zustand nicht erreicht, bricht der Algorithmus nach einer festgelegten Maximalanzahl an Iterationen automatisch ab.

Durch eine affine Koordinatentransformation der Punkte in das durch die Hauptachsen der Singulärwertzerlegung aufgespannte Koordinatensystem mit Koordinatenursprung im Schwerpunkt der „Inlier“-Punkte lässt sich eine Punktwolke aus höhenkodierte Punkten erzeugen. Die

<sup>2</sup>Dieses Vorgehen ist mit der Segmentanalyse des Baumdetektors weiter oben vergleichbar. Lediglich die Dimension ist ein Grad höher, da hier mit 3D-Punktwolken gearbeitet wird und nicht auf der 2D-Tiefenkarte.

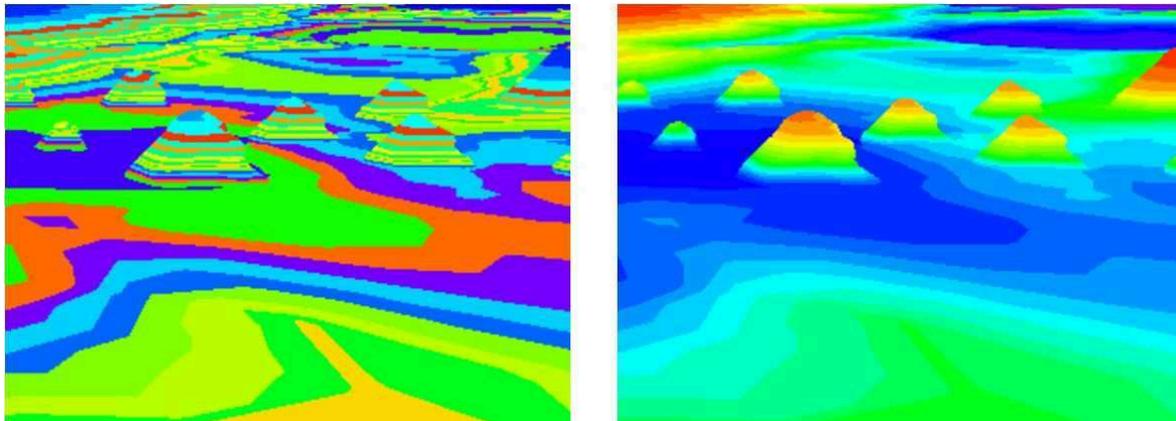


ABBILDUNG 4.11: Ergebnis der k-Means-Segmentierung auf eine relative Höhenkarte mit farbkodierten Höhenklassen. Links: Direktes Ergebnis der Segmentierung mit unsortierten Höhenklassen; rechts: Ergebnis nach der Sortierung der Cluster entsprechend der zugehörigen Höhenklassen.

Höhenwerte können nun normalisiert und in die Bildebene der ursprünglichen Tiefenkarte zurückprojiziert werden. Das Ergebnis ist ein Bild, das in jedem Pixel den normalisierten Abstand des jeweils zugehörigen 3D-Punkts zur geschätzten Bodenebene enthält. Die Darstellung als Bild erlaubt die Verwendung von effizienten Bildverarbeitungsalgorithmen zur Szenenanalyse.

**Höhenkartensegmentierung und Steinextraktion** Die weitere Analyse kann nun pixelbasiert in der Höhenkarte erfolgen, z.B. durch eine Analyse der Bildgradienten, also dem differentiellen Wertvergleich der Nachbarpixel. Gor et al. schlagen eine kMeans-Segmentierung der Höhenkarte vor, um alle Pixel auf eine begrenzte Anzahl von Klassen aufzuteilen. Es wird ebenfalls vorgeschlagen, die Anzahl der Klassen auf  $k = 21$  festzulegen, wobei dieser an sich willkürlich gewählt werden kann, solange ausreichend viele Klassen für die nachfolgende Analyse gewählt werden. Eine entsprechende Umsetzung der kMeans-Segmentierung unterteilt die feinstufig verlaufende Höhenkarte in gröbere Regionen, denen jeweils ein Wert  $g \in [0; k - 1] \subset N$  zugeordnete wird. Die resultierende Segmentierung ist zunächst unsortiert wie in Abbildung 4.11 links.

Gor et al. schlagen in [183] vor, eine regelbasierte Klassifizierung der Höhenklassen unter Betrachtung der Klassen-Nachbarschaften durchzuführen. Jedoch setzt dies eine ausführliche Analyse aller Nachbarschaften

jeder Region voraus. Mit Hilfe von bildverarbeitenden Algorithmen, lässt sich diese Analyse einfacher und schneller lösen. Daher wird an dieser Stelle vom Vorgehen von Gor et al. abgewichen. Dazu müssen zunächst die Segmentierungsklassen nach ihrer relativen Höhe sortiert werden. Die resultierende sortierte Segmentierung ist in Abbildung 4.11 rechts dargestellt.

Die sortierte und segmentierte Höhenkarte lässt nun folgende Analysen zu: Das letzte Segment der sortierten Höhenklassen muss die Spitze eines Steins repräsentieren, da wir davon ausgehen, dass die Steine auf der zuvor detektierten Bodenebene liegen und somit die höchsten Gegenstände der durch die Höhenkarte abgedeckten Szene darstellen. Von dieser Spitze lassen sich nun die Höhenklassen abwärts verfolgen, bis der Boden erreicht wird<sup>3</sup>. Genauer formuliert heißt das, die Spitze eines Steins besitzt keinen Nachbarn einer höheren Clusterstufe und nur genau einen Nachbarn nach unten mit einer Clusterstufe Unterschied. Innerhalb jedes Steins besitzt jede Region genau einen Nachbarn nach oben und genau einen Nachbarn nach unten, jeweils mit der Differenz von genau einer Clusterstufe. Der Boden besitzt weit ausgedehnte Clusterstufen und hat demnach mehrere Nachbarn nach oben und unten, die sich in einer Stufe unterscheiden. Ausgehend von dieser Analyse können nun von der Spitze eines Felsens die Clusterstufen nacheinander von oben nach unten durchlaufen werden bis zu dem Punkt, wo eine Stufe erreicht wird, die mehr als einen Nachbarn nach oben besitzt.

Im Folgenden werden einander angrenzende Regionen die sich in der Klassenhierarchie um genau eine Klasse unterscheiden „Nachbarn“ genannt. Ausgehend von der ersten Region höchster Klasse wird nun ein Floodfill-Algorithmus angewendet, der eine asymmetrische Einschränkung der Farbtoleranz nach oben und unten erlaubt, da die Klassen hier als Farbwerte kodiert werden, wie in Abbildung 4.11 zu sehen. Es wird auf diese Weise jeweils die absolute Differenz zwischen den Nachbarregionen und des „Seed“-Werts des „Floodfills“ berechnet, und dadurch nur die direkte Nachbarregion der „Seed“-Region markiert. Somit lassen sich die Cluster hierarchisch von der Spitze eines Felsen nach unten zum Boden durchwandern. Alle Regionen, die bis dahin durchlaufen wurden, werden vereinigt und als mögliche Felsregion gespeichert. Anschließend werden erneut von der Spitze des Felsens beginnend alle

---

<sup>3</sup>Da kein exakt planarer Boden vorausgesetzt wurde, kann dieser aus mehreren Höhenklassen bestehen

Nachbarn markiert, deren Differenz zum höheren Nachbarn einer Clusterstufe entspricht (im Gegensatz zum vorherigen Durchlauf wird hier die relative Differenz (Gradient) zwischen zwei benachbarten Regionen betrachtet). Durch den zweiten Durchlauf erhält man nun eine geschlossene Nachbarschaftskette von der Spitze des Felsen bis zur tiefsten von dieser erreichbaren Region. Diese Regionen werden als bereits besucht markiert und werden nicht weiter betrachtet. Diese Vorgehensweise wird solange durchgeführt, bis alle Regionen oberhalb einer festgelegten Schranke betrachtet wurden. Nach erfolgreicher Abgrenzung möglicher Regionen, müssen diese attribuiert werden, um danach entscheiden zu können, ob es sich um einen für die Lokalisierung geeigneten Felsen handelt oder nicht. Hierzu wird jede Einzelregion mit der metrischen Höhenkarte verschnitten, um die Menge von 3D-Szenenpunkten zur Region zu erhalten.

Aus diesen geclusterten Punktwolken können sowohl Höhe als auch Durchmesser der betrachteten Felsen ermittelt werden. Um unterscheiden zu können, ob es sich bei dem gegenwärtig betrachteten Cluster um einen Fels - im Sinne des Landmarkenbegriffs - oder etwas anderes handelt, wird an dieser Stelle ein Kompaktheitsmaß  $k$  eingeführt:

$$k = (2\sqrt{\pi A})/C. \quad (4.7)$$

Dieses Maß erlaubt eine zuverlässige Unterscheidung zwischen kompakten, und damit als Landmarken nutzbaren Objekten, und weniger kompakten, weitläufigen Erhebungen. Das Maß wird definiert als der Quotient zwischen dem Umfang eines Kreises mit gleicher Fläche  $A$  wie die Region des Clusters in der Bildebene und der Länge der Regionskontur  $C$ . Eine entsprechende Grenze kann vom Benutzer ebenso wie eine minimale und maximale Größe angegeben werden, um die Erkennungskriterien des Algorithmus im Betrieb ändern zu können. Im Folgenden werden diese Objekte Felsen bzw. Steine genannt, da dieser Algorithmus zur Detektion dieser in Planetenexplorationsmissionen angewendet werden soll.

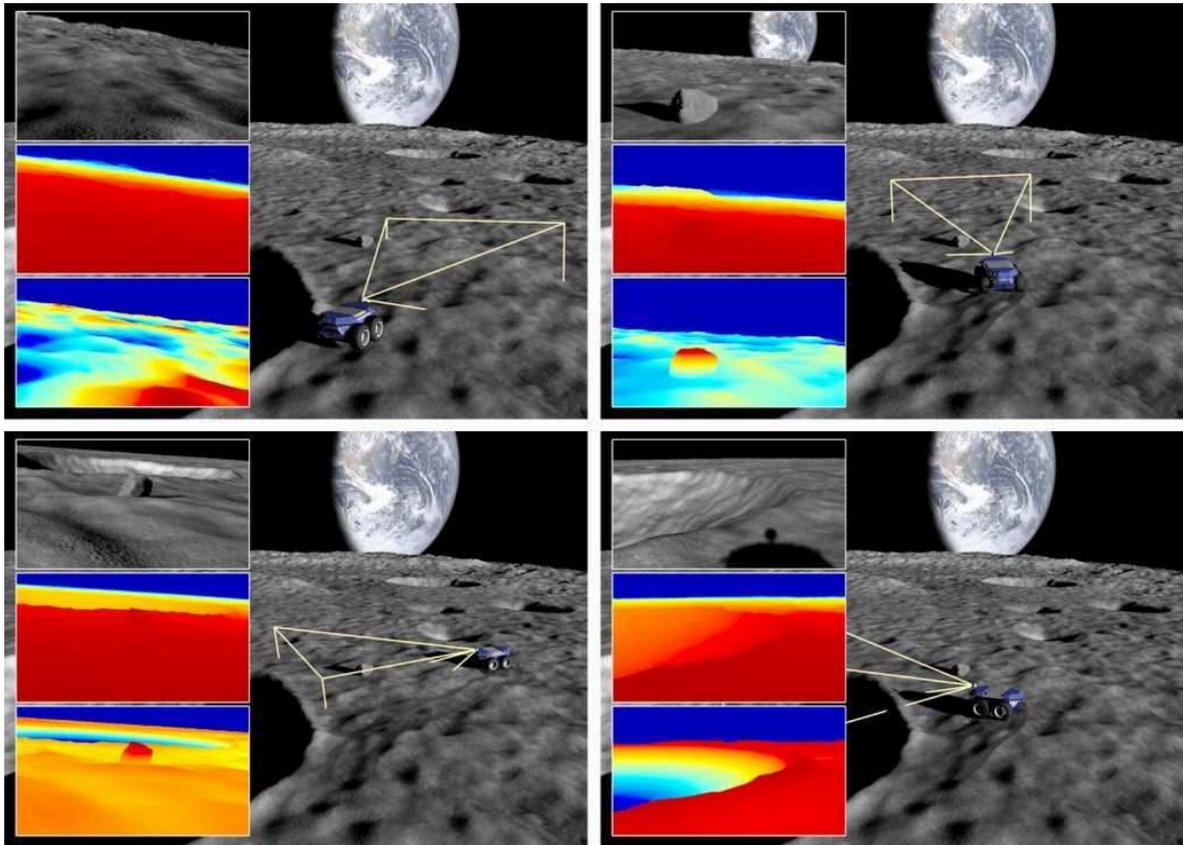


ABBILDUNG 4.12: Simulierte extraterrestrische, planetare Explorationsmission im Virtuellen Testbed. Die Einzelbilder stellen unterschiedliche Bereiche der Umgebung dar, die in der Höhenkarte deutlicher identifiziert werden können, als im Tiefenbild. In den vier Einzelbildern sind jeweils drei Overlays angezeigt. Von oben nach unten: Farbbild, Tiefenbild (simulierte ToF-Kamera), daraus berechnete Höhenkarte zur Steindetektion.

### Ergebnisse der Steindetektion

Eine Ausführliche Untersuchung der Steindetektion wurde im Virtuellen Testbed durchgeführt, wo eine Explorationsmission auf einer extraterrestrischen, planetaren Oberfläche simuliert wurde. Ausgehend von einer Steindetektion in einer sehr einfachen, flachen Umgebung wurde zunächst die Steindetektion unabhängig von der Bodenschätzung getestet. Durch geeignete Parametrierung der erlaubten Breite, Höhe und Kompaktheit der Landmarken war hier eine zuverlässige Detektion und

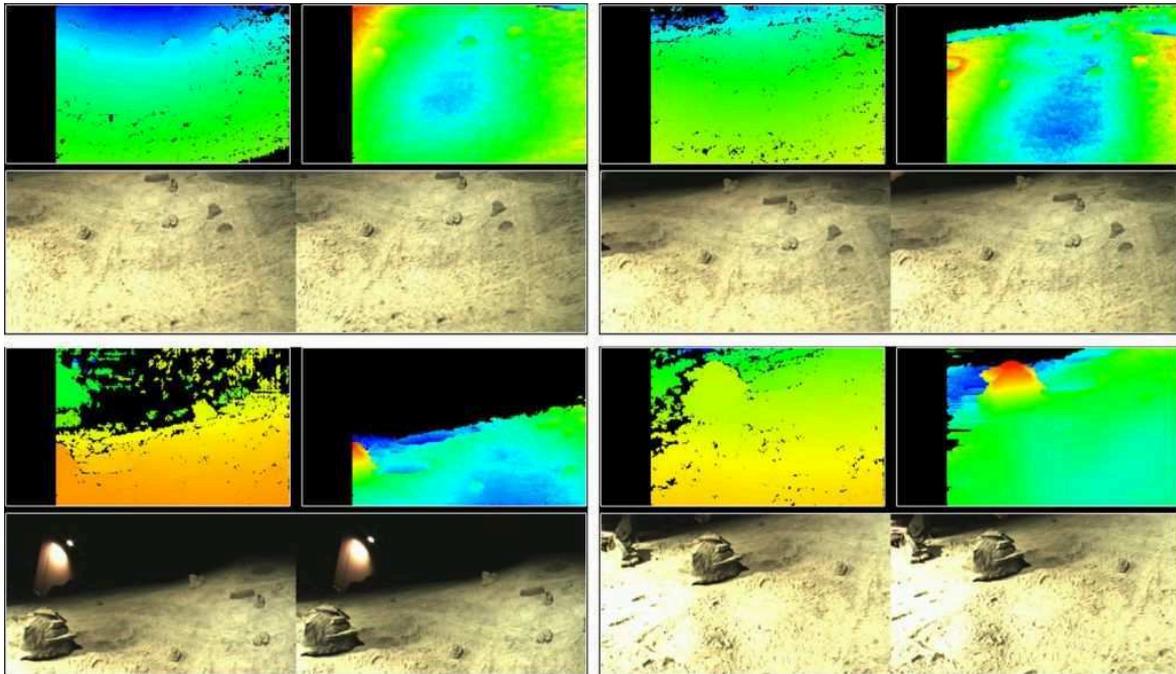


ABBILDUNG 4.13: Ergebnisse der Bodenschätzung und Höhenkartengenerierung im Testkrater des DFKI Bremen. Die vier Einzelbilder stellen unterschiedliche Bereiche der Testumgebung dar. Sie bestehen jeweils aus einem Stereodoppelbild (unten), der daraus berechneten Tiefenkarte (oben links) und der Höhenkarte nach erfolgter Bodenebenenschätzung.

Attribuierung (Höhe, Breite und Kompaktheit(siehe Gleichung 4.7))<sup>4</sup> möglich.

Nachdem eine zuverlässige Detektion von Erhebungen in der Punktwolke und eine Klassifikation als Steine gewährleistet wurde, muss für eine Anwendung in anspruchsvollen und unstrukturierten Umgebungen, die Detektion mit der Bodenebenenschätzung kombiniert werden. Auch hier soll zunächst die Bodenebenenschätzung gesondert betrachtet werden.

<sup>4</sup>Breite und Kompaktheit werden lediglich zur Unterscheidung zwischen Stein-Landmarken und anderen Hindernissen oder Erhebungen genutzt. Aufgrund der Beliebigkeit von Steingeometrien ist eine anderweitige Nutzung dieser Attribute, etwa zur Unterscheidung bei der Bildung von Paaren während der Lokalisierung, nicht ratsam.

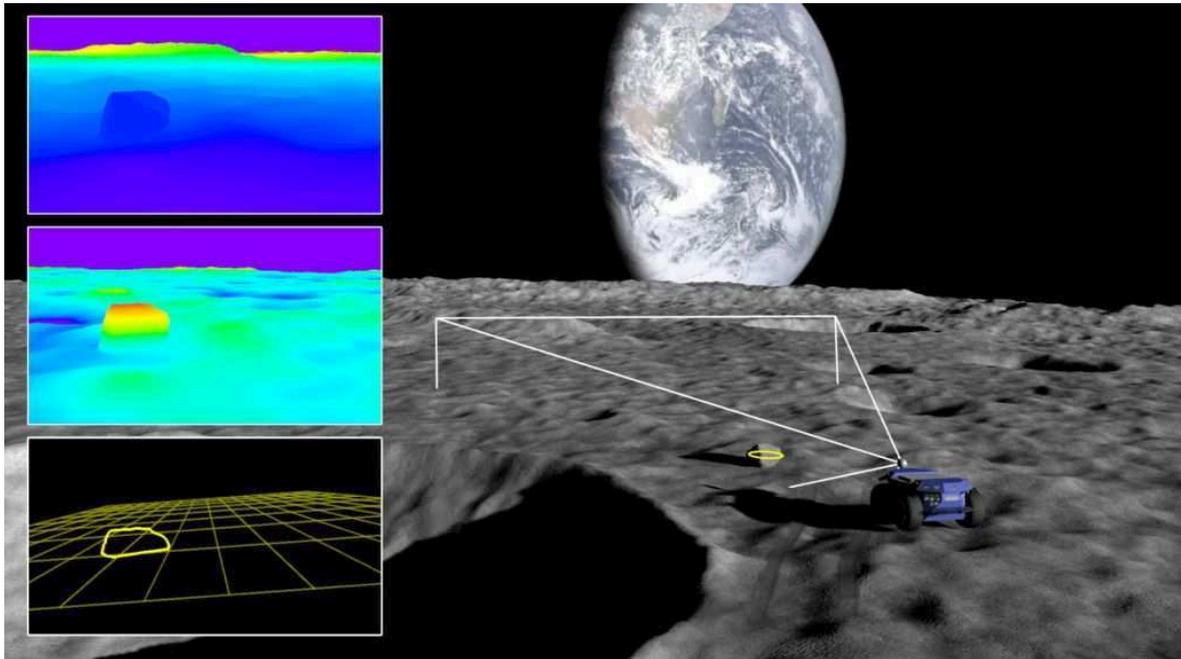


ABBILDUNG 4.14: Steindetektion im virtuellen Testbed zur extraterrestrischen Planetenexploration

Abbildung 4.12 zeigt einige Einzelaufnahmen einer Explorationsmission auf einer planetaren Oberfläche im virtuellen Testbed. Die jeweiligen drei Bildoverlays zeigen eine Farbaufnahme des Kamerasensors (oben), ein Tiefenbild einer simulierten ToF-Kamera (mitte) und einer, auf der Bodenebenenschätzung resultierenden, Höhenkarte (unten). Der Unterschied zwischen der entfernungskodierten Tiefenkarte und der Höhenkarte ist hier deutlich zu erkennen und unterstreicht die Notwendigkeit dieser Transformation für eine zuverlässige Steindetektion in den Sensordaten.

Die Bodenebenenschätzung wurde ebenfalls im realen Testgelände der „Weltraum-Explorationshalle“ am Deutschen Forschungszentrum für Künstliche Intelligenz (DFKI) in Bremen<sup>5</sup> durchgeführt, um die Ergebnisse auch an realen Sensoren verifizieren zu können. Einige repräsentative Einzelaufnahmen der Testsequenzen sind in Abbildung 4.13 zusammengestellt. Hier sind jeweils die Tiefenkarte (links) als Ergebnis der Stereobildverarbeitung und die aus der Bodenebenenschätzung resultierende Höhenkarte (rechts) als Overlays eingeblendet. Topologische

<sup>5</sup><http://robotik.dfki-bremen.de/de/forschung/testanlagen/weltraum-explorationshalle.html>

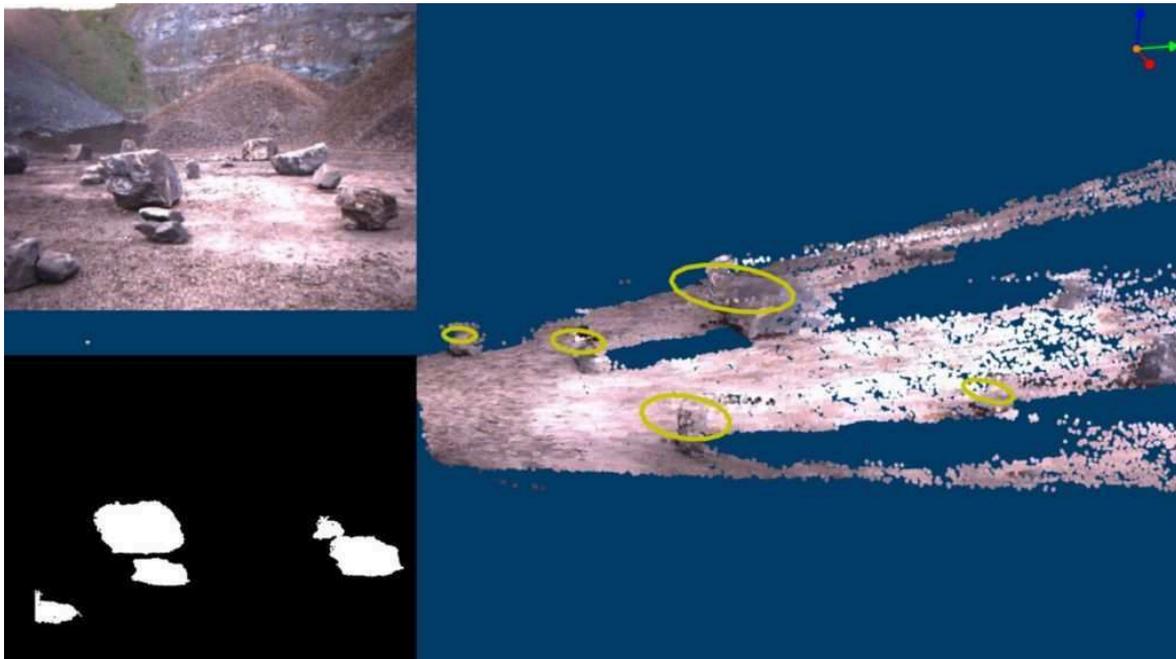


ABBILDUNG 4.15: Steindetektion im Steinbruch

Strukturen der Oberfläche werden auch in den realen Aufnahmen erst in der Höhenkarte deutlich sichtbar, sodass Hindernisse und Landmarken algorithmisch darin erfasst werden können.

Die Kombination aus Bodenenebenenschätzung und Steindetektion wurde zunächst wieder im Virtuellen Testbed für extraterrestrische Planetenexploration evaluiert. Das Ergebnis ist in Abbildung 4.14 dargestellt. Aus den Sensordaten (oberes Overlay) wird zunächst wieder die Bodenebene geschätzt und eine Höhenkarte berechnet (mittleres Overlay) aus der dann der Stein als Landmarke extrahiert wird. Das untere Overlay zeigt die Ausrichtung der geschätzten Ebene und den darin erkannten Stein, der sich bereits in der Höhenkarte deutlich von seiner Umgebung abhebt.

Realtests wurde mit der mobilen Roboterplattform SeekurJr und dem ersten Prototyp der SELMA-Einheit in einem Steinbruch durchgeführt. Abbildung 7.7 zeigt das Areal im Steinbruch mit den dort verteilten Steinen als Landmarken. Der zur Verifikation benötigte Ground-Truth (Steinpositionen und -umfang) wurde mit Hilfe eines 3D-Laserscanners vom Boden, und mit Hilfe einer Kameradrohne aus der Luft aufgenommen und anschließend manuell kartiert. Hier konnte die Tauglichkeit des Verfahrens für die zuverlässige Detektion von Steinlandmarken in

Explorationsszenarien verifiziert werden. Abbildung 4.15 zeigt eine Visualisierung der Sensordaten aus diesen Tests samt einer Darstellung der erkannten Steine sowie deren Positionen. In der Abbildung ist die aus der Stereobildverarbeitung berechnete und kolorierte Punktwolke in 3D dargestellt. Die erkannten Steine und deren Durchmesser sind als gelbe Kreise (ebenfalls in 3D, daher erscheinen sie elliptisch) über den Detektionen visualisiert. Eins der beiden Ausgangsbilder ist im Overlay links oben zu sehen. Die segmentierten Steinregionen aus dieser Aufnahme sind als binäre Maske im Overlay darunter abgebildet. Eine Auswertung der Lokalisierung auf den hier erkannten Landmarken folgt im Kapitel 5.4.3.

#### 4.2.4 Erkennung von künstlichen Markern

In natürlichen Umgebungen können, wie anhand von Bäumen und Steinen bereits gezeigt, verschiedene natürliche Strukturen als Landmarken genutzt werden. In der Regel sind diese Landmarken auf natürliche Weise verteilt und weisen damit verschiedene Abstände zueinander auf. Bei der Betrachtung ganzer Gruppen von Landmarken ermöglicht diese natürliche Verteilung eine Distinguierung der einzelnen Landmarken, auch wenn sie sich äußerlich nicht sonderlich von einander unterscheiden. Beispielsweise wachsen Bäume im Wald in unterschiedlichen Abständen zueinander. Im Gegensatz dazu werden in Städten Bäume entlang einer Straße meist in regelmäßigen Abständen eingepflanzt und da sie sich in Alter, Baumart und anderen Attributen oft sehr ähnlich sind, sehen sie in vielen Fällen sogar gleich aus. Gebäude werden in der Regel symmetrisch aufgebaut und bieten in ihrem Inneren meist noch weniger Unterscheidungsmerkmale, was die Verteilung zur Orientierung geeigneter Objekte angeht. Daher müssen die Objekte, die in künstlichen Szenarien (Städte, Indoor, etc.) zur Lokalisierung genutzt werden sollen, besser von einander unterscheidbar sein. Existieren mehrere verschiedene Objekttypen, wie Verkehrsschilder, Ampelanlagen, Fahrbahnmarkierungen, Bäume, etc. in Städten, so erreicht man diese Unterscheidung allein durch die Vielfältigkeit der verwendeten Landmarken. Stehen allerdings nicht genügend viele verschiedene Landmarken zur Verfügung, wie etwa bei der Indoor-Lokalisierung, so erlauben künstliche Marker eine kostengünstige und aufwandsarme Präparierung der Umgebung für eine semantische Kartierung und landmarkenbasierte Lokalisierung.

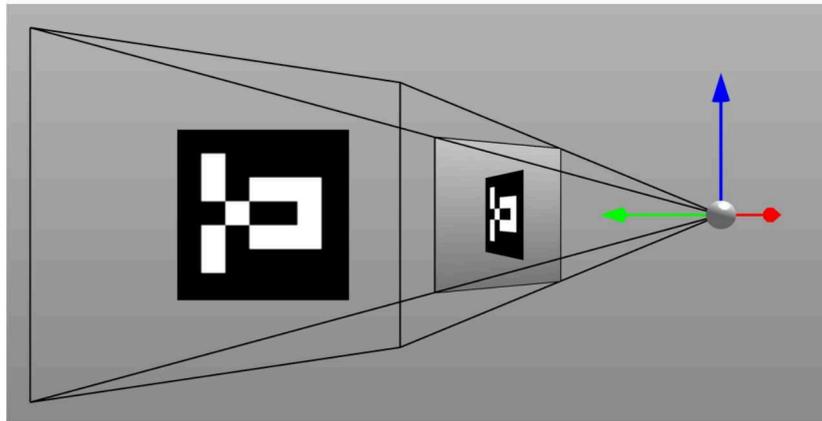


ABBILDUNG 4.16: Anhand des projizierten Abbilds des Markers im Bildraum der Kamera, lässt sich die relative Lage des Markers bezüglich des Kamerazentrums rekonstruieren.

Dieser Abschnitt widmet sich der Thematik der künstlichen Marker (im Englischen „fiducials“). Wird ein mit Informationen kodiertes Objekt in den Sichtbereich einer Kamera platziert, sodass dieses durch maschinelles Sehen wiedererkannt und ausgelesen werden kann, dann wird dieses Objekt als Marker bezeichnet. In der Regel sind diese zweidimensional und kodieren Informationen durch fest definierte Strukturen. Unterschieden werden soll hier zwischen Markern, welche nur zur Informationsübermittlung genutzt werden können (z.B. klassische Barcodes) und solchen, die ebenfalls die Bestimmung der relativen Lage zwischen Kamera und Marker erlauben (z.B. QR-Tags oder AR-Marker).

Die Verwendung von AR-Markern erlaubt neben einer Detektion und Beschreibung des Markers, anders als natürliche Landmarken, bereits eine direkte Lokalisierung der Kamera bezüglich eines Markers, vorausgesetzt die Geometrie des Markers sowie die intrinsischen Parameter der verwendeten Kamera sind bekannt. Denn in diesem Fall wird nicht nur die Existenz des Markers im erfassbaren Bereich der Kamera detektiert, sondern darüber hinaus ist es möglich das projizierte Abbild des Markers im Bildraum der Kamera dazu zu verwenden, die Lage des Markers im 3D-Raum relativ zur Kamera zu rekonstruieren. Abbildung 4.16 veranschaulicht die bei der perspektivischen Projektion entstehende Verzerrung der Markergeometrie.

Sind die intrinsischen Kameraparameter, d.h. die Brennweite, Linsenverzerrungseigenschaften, etc. (vgl. Gleichung 2.13 aus Kapitel 2.1.3), bekannt, so lassen sich in einem ersten Schritt diese Linsenverzerrungen

herausrechnen, sodass fortan wie mit einem Lochkameramodell (Kanoni-  
sche Projektionsmatrix  $P_0$  aus Gleichung 2.10) fortgefahren werden kann.  
An dieser Stelle sei noch einmal auf die Grundlagen zur Lochkamera und  
den intrinsischen Kameraparametern bei der Verwendung von Kameras  
mit Linsenoptiken in Kapitel 2.1.3 verwiesen.

Für die Detektion von AR-Markern wurde die in [141] vorgestellte  
Methode verwendet, deren Detektion und Erkennung in vier Stufen  
arbeitet:

1. Bildsegmentierung zur Extraktion der markantesten Konturen im  
Bild
2. Konturfilterung durch Schwellwertbildung und anschließendem  
Verwerfen aller Polygone mit mehr als vier Eckpunkten
3. Marker-Raster-Extraktion durch erneute Schwellwertbildung im In-  
nern der übrig gebliebenen Polygone und Detektion des schwarzen  
Randes im gerasterten Binärmuster
4. Markeridentifikation und Fehlerkorrektur durch den Abgleich des  
gerasterten Binärmusters mit den Einträgen im vorgegebenen Marker-  
Lexikon unter der Ausnutzung der Redundanzen im Markercode,  
um Detektionsfehler zu erkennen und ggf. zu korrigieren.

Eine Übersicht des Vorgehens ist in Abbildung 4.17 dargestellt.

Als Ergebnis erhält man pro gefundenen Marker die vier Eckpunkte  
der Projektion in Bildkoordinaten sowie die jeweilige im Marker binär  
kodierte Information. Bei dem hier verwendeten Markertyp wird jeweils  
eine eindeutige Identifikationsnummer (Id) kodiert, die zur eindeutigen  
Unterscheidung und Zuordnung detektierter Marker genutzt werden  
kann. Ab diesem Punkt ist unter der Voraussetzung, dass sowohl die  
Geometrie und Größe der Marker, als auch die intrinsischen Kamera-  
parameter bekannt sind, eine Posenschätzung der Marker relativ zur  
Kamera möglich. Bei künstlichen Markern wird meist eine sehr einfache  
Form, gewählt, d.h. symmetrisch mit regelmäßigen Außenkanten, sodass  
ein einziger Skalierungswert die Abmessungen des Markers vollständig  
beschreibt.

Da nun lediglich die Geometrie des Markers zur Posenrekonstruktion  
verwendet wird, ist dieses Verfahren losgelöst von der Markerdetekti-  
on selbst zu betrachten und kann auf alle Objekte (Im Folgenden wird

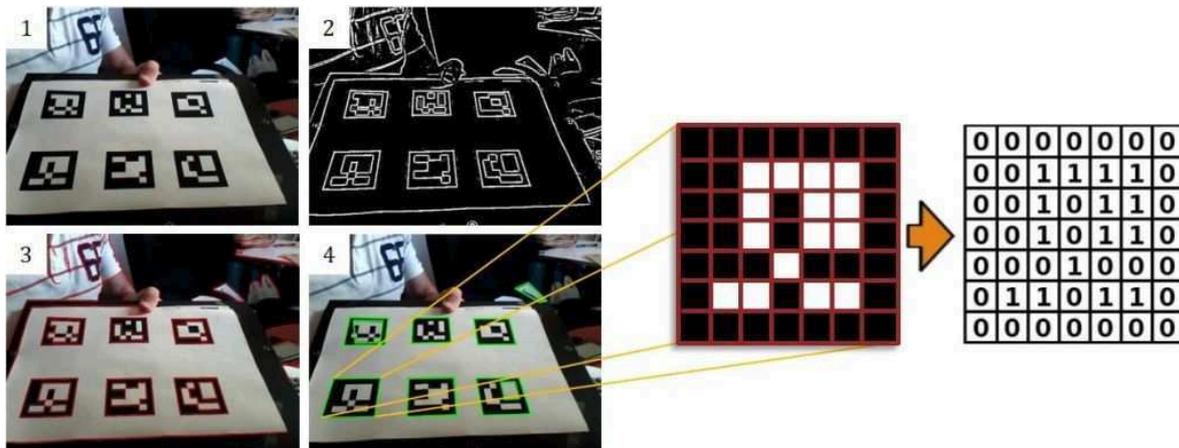


ABBILDUNG 4.17: Verarbeitungsprozess zur AR-Markerdetektion: 1 Originalbild, 2 Konturdetektion nach lokaler Schwellwertbildung, 3 Polygonale Approximation und Filterung, 4 Markierung erkannter Marker. Rechts: Beispiel eines Markers mit Bitraster. Quelle: [141]

der Einfachheit halber weiterhin von Markern gesprochen) angewendet werden, deren Geometrie a priori bekannt ist. Die Markerbasierte Lokalisierung wird in Kapitel 5.3 eingehend behandelt.

### Ergebnisse der Markerdetektion

Bei der Verwendung von 2D-Markern ist zu beachten, dass deren Detektierbarkeit aufgrund von Größe und perspektivischer Verzerrung im Bildraum der beobachtenden Kamera, limitiert ist. Eine schlichte Prüfung auf Sichtbarkeit ist hier nicht ausreichend. Aus diesem Grund wurde zunächst die Detektierbarkeit von AR-Markern<sup>7</sup> in einer kontrollierten Simulationsumgebung überprüft. In Abbildung 4.18 ist der Sichtbarkeitsbereich eines Markers in der Ebene auf Höhe des Markerzentrums angegeben. Hier ist deutlich zu erkennen, dass die Detektierbarkeit des Markers bei frontaler Sicht durch die Entfernung limitiert ist, und zum Rand hin durch den steigendem Betrachtungswinkel. Die maximale Distanz für eine erfolgreiche Detektierbarkeit hängt dabei sowohl von der Auflösung des Kamerabildes als auch von der Größe des Markers ab. Ist die Kameraauflösung gegeben, so steigt die maximale Entfernung, an der der Marker noch zuverlässig detektiert werden kann, linear mit der

<sup>7</sup>Es wurden Marker und Detektionsalgorithmus aus [141] verwendet.

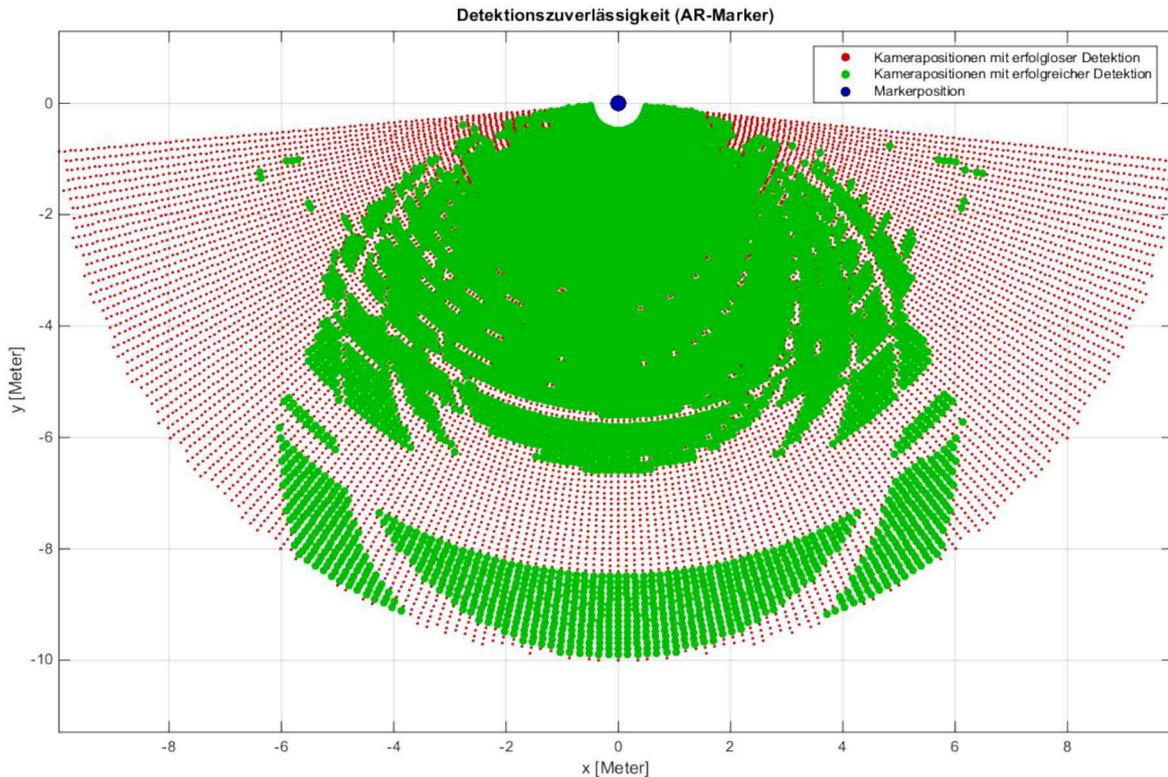


ABBILDUNG 4.18: Detektierbarkeit von AR-Markern<sup>6</sup> aus verschiedenen Entfernungen und mit unterschiedlichem Betrachtungswinkel

Kantenlänge des Markers. Es handelt sich hier um ein allgemeines Skalierungsproblem, das sich direkt aus dem einfachen Lochkameramodell für Bildprojektionen als  $\lambda \vec{x} = P \vec{X}$  schreiben lässt. Genauer wurde dies in Abschnitt 2.1.3 behandelt (siehe dort Gleichung (2.9)).

Die Analyse der Detektierbarkeit wurde in der Simulation unter optimalen Bedingungen durchgeführt, sodass eine unabhängige Betrachtung von sämtlichen äußeren Einflüssen erreicht werden kann. Dazu wurde ein ideales Lochkameramodell (mit einem Öffnungswinkel von  $60^\circ$  und einer Bildauflösung von  $640 \times 480$  Pixeln) und ein Marker mit Kantenlänge von  $0,2m$  verwendet. Die hier angegebenen Werte liefern damit eine obere Schranke, da bei der Verwendung realer Hardware optische Effekte (Linsenverzeichnung, radiale Verzerrung, etc.) und äußere Einflüsse (Beleuchtung, Reflexionen, etc.) die Detektierbarkeit zusätzlich reduzieren können. Somit lassen sich diese Auswertungen als Entscheidungsunterstützung zur Auswahl von Größe und Positionierung von

Markern und die Eigenschaften der zu nutzenden Kamera verwenden.

In Abbildung 4.18 sind, neben dem grundsätzlichen Detektionsbereich des Markers, deutlich regelmäßige Detektionslücken zu erkennen. Diese Detektionslücken lassen sich auf die Diskretisierung des Bildes im Pixelraster zurückführen, das dazu führt, dass bei bestimmten Entfernungen und Betrachtungswinkeln die Abstände während der Diskretisierung, ähnlich des Aliasing-Effekts in der Computergrafik, nicht eingehalten werden können. Dieser Effekt ist ebenfalls bei der Wahl von Markergrößen und Kameraauflösungen zu berücksichtigen.

## 4.3 Semantische Beschreibung durch Attribuierung

Wie in den letzten Abschnitten zur Detektion verschiedener Landmarken aus Sensordaten gesehen, bauen sich die einzelnen Detektionsalgorithmen aus verschiedenen Einzelteilen zusammen, die jeweils der Vorverarbeitung, (Teil-) Detektion oder Disambiguierung dienen. Die temporären Teilergebnisse dieser einzelnen Verarbeitungsschritte werden nach der erfolgreichen Detektion einer Landmarke i.d.R. nicht mehr benötigt und deswegen verworfen. Allerdings enthalten diese Teilergebnisse bereits zusätzliche Informationen, welche die jeweils detektierte Landmarke näher beschreiben. Somit ist es oftmals möglich durch nur wenig zusätzlichen Aufwand neben der Detektion an sich auch eine Beschreibung des detektierten Objekts zu erhalten.

Eine ausführliche Betrachtung der semantischen Beschreibung durch Attribuierung soll im folgenden für Bäume gegeben werden, da hier auch der Fokus der späteren Anwendungen liegen wird.

### 4.3.1 Attribuierung von Baumlandmarken

Bei der Baumdetektion mittels Laserscannern und aus (Stereo-)Punktwolken ist die horizontale Beschränktheit der Detektion ein entscheidender Faktor zur Unterscheidung zwischen einem Baum und anderen Strukturen in der Umgebung. Im Laserscan wird bereits nach kreisförmigen Strukturen gesucht. Daher sind die notwendigen Informationen zum Baumdurchmessers bereits implizit durch die Detektion gegeben. Sowohl

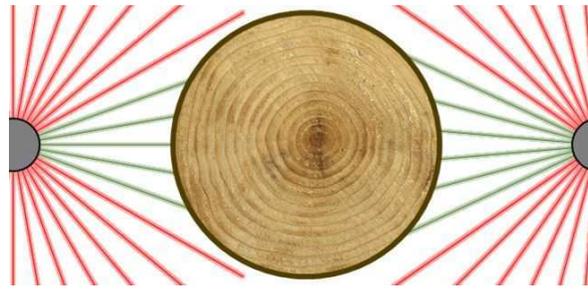


ABBILDUNG 4.19: Analyse einer Durchmesserbestimmung mittels 2D-Laserscanner. Unterscheidung zwischen einer Frontalmessung (links) und einer um die halbe Winkelauflösung gedrehte Messung (rechts)

die Hough-Transformation für Kreise als auch die allgemeine Kreisgleichung ermitteln neben der Position des Mittelpunkts des detektierten Baumquerschnitts auch den zugehörigen Radius. Bei der Detektion von homogenen, vertikaldominanten Bereichen in gerasterten Punktwolken, z.B. aus Stereokameradaten, wird durch die Hauptkomponentenanalyse dieser Bereiche sowohl die Richtungen als auch die Varianzen der beiden Hauptachsen ermittelt. Die erste Hauptachse entspricht hierbei der Wuchsrichtung des Baumes und erlaubt die Bestimmung des Neigungswinkels aus Blickrichtung der Kamera. Durch die verhältnismäßig hohe horizontale Auflösung von Stereokameras (vgl. Abbildung 4.4) kann die Durchmesserbestimmung des Baumes orthogonal zur Hauptachse durch Abstandsbildung der beiden Baumkanten ebenfalls durchgeführt werden.

Eine etwaige Boden­neigung kann mit Hilfe des vorgestellten Ansatzes zur Boden­ebenenschätzung ermittelt werden. Sind sowohl Boden­neigung als auch die Neigung des Sensors zum Zeitpunkt der Aufnahme bekannt, so lässt sich der Durchmesser auch, wie in der Forstwirtschaft üblich, auf einer Höhe von 1,3 Metern über dem Boden als sogenannten Brusthöhendurchmesser bestimmen.

### Ergebnisse der Baumattribuierung

Die in Kapitel 4.2.2 vorgestellten Ansätze zur Detektion von Bäumen in 2D-Laserscanprofilen, liefern gerade in Bezug auf die Attribuierungsgenauigkeit stark unterschiedliche Ergebnisse. Zur Gegenüberstellung dieser Unterschiede, wurden in einer Simulationsumgebung mehrere

Testreihen durchgeführt, in denen ein idealer Zylinder mit 0,2 m Durchmesser aus Entfernungen zwischen eins und zehn Metern detektiert werden sollte.

Als Sensor soll ein idealer 2D-Laserscanner mit  $0,25^\circ$  Winkelauflösung betrachtet werden, der einmal frontal auf den Baum gerichtet wurde, d.h. ein Messstrahl wird direkt in Sichtrichtung des Baumes geworfen, und einmal um  $0,125^\circ$  um die Hochachse gedreht, sodass der Baum nicht frontal getroffen wird, sondern die Strahlen den Baum mit halber Winkelauflösung rechts und links von der Sichtachse des Laserscanners treffen. Den Einfluss dieser minimalen Drehung auf die Detektionsergebnisse ist in Abbildung 4.19 skizziert. Man sieht deutlich, dass im frontalen Fall (links) ein Messstrahl weniger vom Baum reflektiert wird, als im gedrehten Fall (rechts). Die Auswirkung dieser Drehung ist in Abbildung 4.20 ebenso ersichtlich, wie die allgemeine Genauigkeit der einzelnen Verfahren in diesem einfachen Testaufbau. Die Messungen dienen deshalb ausschließlich dem Zweck des Vergleichs der einzelnen Verfahren und treffen keine Aussage über die tatsächliche Genauigkeit in der realen Anwendung. Etwa wurden bei diesen Vergleichsmessungen die Hochachse des Laserscanners sowie der Messzylinder exakt lotrecht ausgerichtet. Bei Messungen mit realer Hardware und natürlich gewachsenen Bäumen, trifft diese Vereinfachung selbstverständlich nicht zu.

Im Folgenden sollen Messungen analysiert werden, die mit der in Kapitel 3.5 vorgestellten SELMA-Einheit in der Marteloskopfläche [184] „Meielewald“ der Hochschule für Agrar-, Forst, und Lebensmittelwissenschaften (HAFL) in Bern aufgenommen wurden. Bei Marteloskopflächen sind im Allgemeinen alle Bäume mit einem Brusthöhendurchmesser von über 7,5 Zentimeter eingemessen und kartiert. Diese Daten können als Referenzmessungen verwendet werden, um die aufgenommenen Beobachtungen zu evaluieren. Zur Aufnahme der Daten wurde ein Teil der Fläche in mehreren Teildurchläufen, ausgehend von einem gemeinsamen Startpunkt, mit der SELMA-Einheit durchschritten, und die Sensordaten jeweils aufgezeichnet. Da die Daten dazu genutzt wurden, Bäume zu detektieren, in eine Karte zu schreiben und sich anschließend darin zu lokalisieren und parallel die Karte weiter aufzubauen, kann die Durchmesserbestimmung in diesem Zusammenhang nicht vollständig losgelöst von den anderen beteiligten Komponenten beurteilt werden. Beispielsweise können Fehldetektionen in dieser komplexen Verarbeitungskette mit den

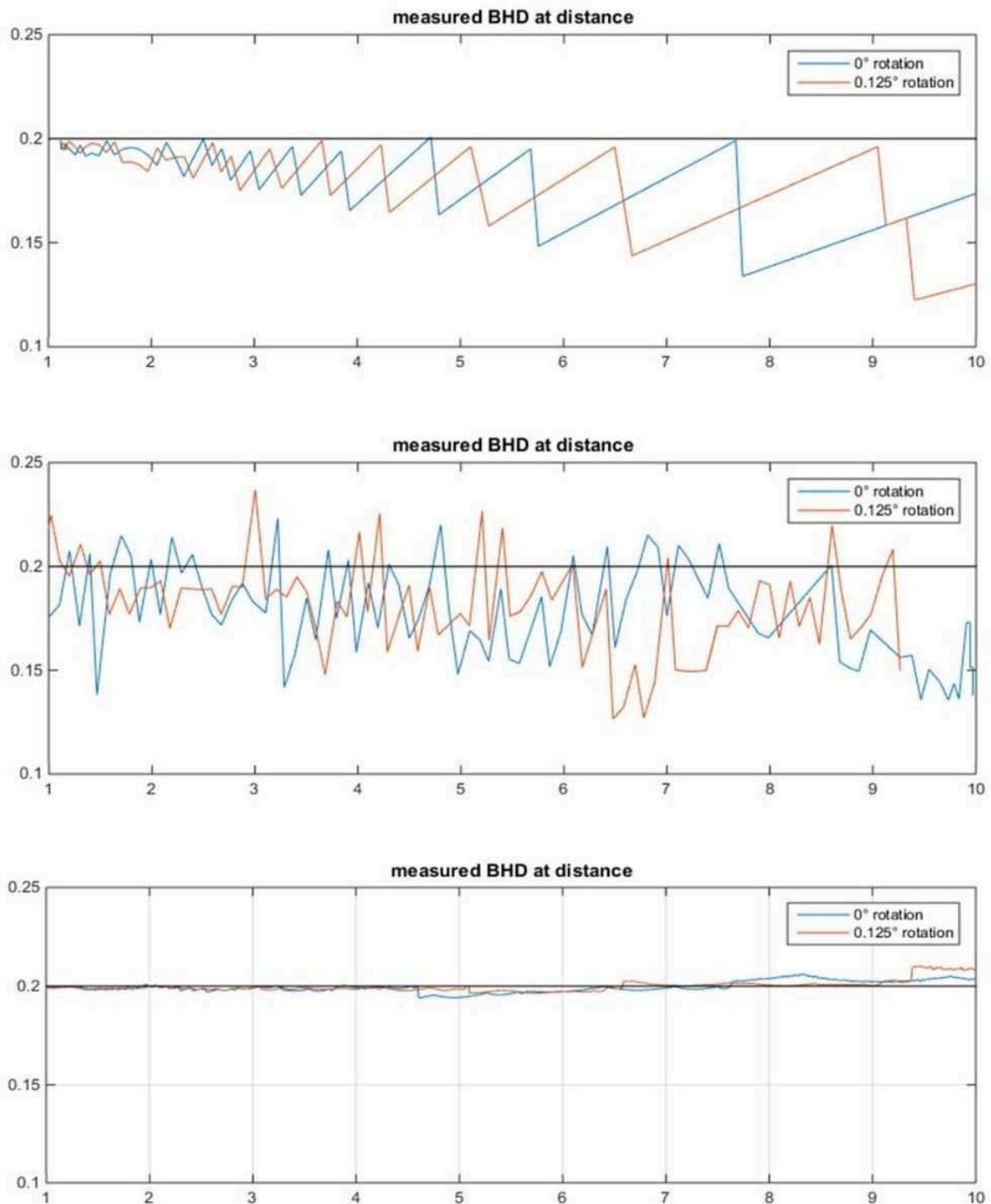


ABBILDUNG 4.20: Gegenüberstellung der vorgestellten Algorithmen zur Bestimmung von Baumdurchmessern aus 2D-Laserscans: Einfache Durchmesserbestimmung aus Differenzbildung (oben), Verwendung einer Kreis-Houghtransformation (mitte), Verwendung der allgemeinen Kreisgleichung (unten)

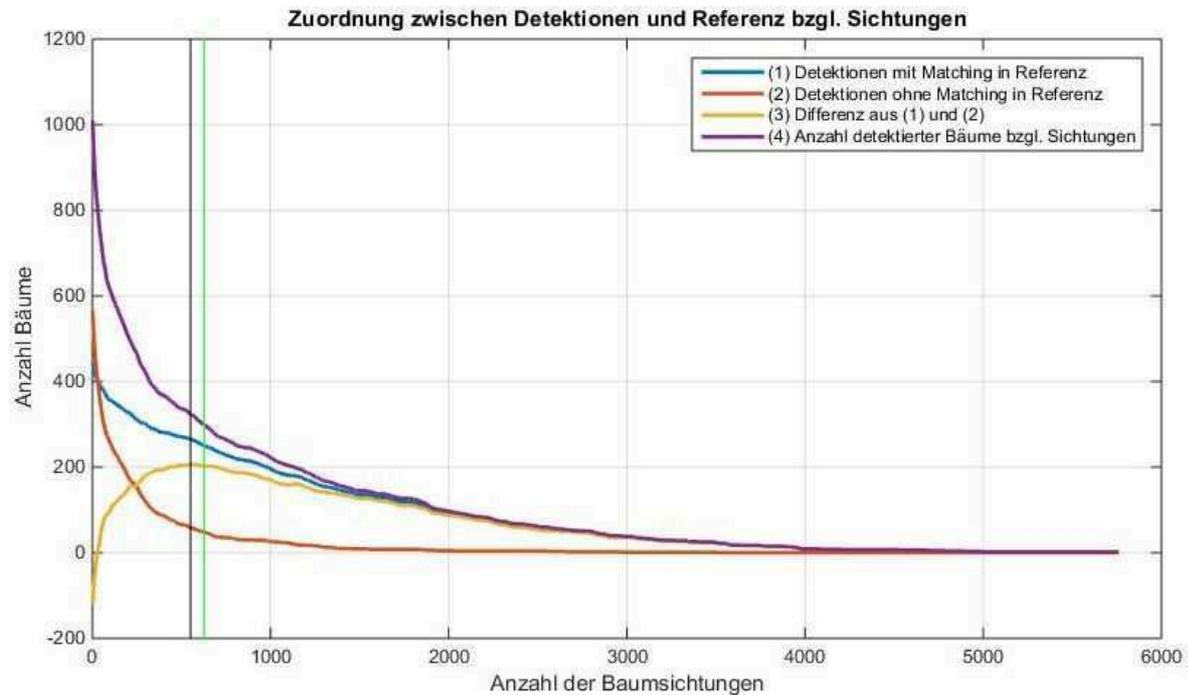


ABBILDUNG 4.21: Bestimmung des idealen Werts an Sichtungen mit Hilfe von Ground Truth aus Zollikofen. In der Anwendung (ohne Ground Truth) soll der Mittelwert aller Baumsichtungen verwendet werden. Dieser liegt in diesem Testgebiet (schwarz) nur knapp über dem idealen Wert (grün).

korrekt erkannten Bäumen während der Lokalisierung und Kartenaktualisierung verschmolzen werden, sodass es mitunter zu Abweichungen in der Bestimmung der Baumattribute geben kann. Aus diesem Grund ist die Anzahl der Sichtungen eines jeden Baumes besonders wichtig, da diese Information bereits während der Kartenerzeugung genutzt wird, die Einzelmessungen mit der Häufigkeit ihres Auftretens zu gewichten. Später können dann alle Bäume entfernt werden, die nicht ausreichend viele Sichtungen aufweisen, um Messfehler (hier: *false positives*) zu verwerfen.

Die Angabe „ausreichend viele“ ist in Abhängigkeit von der Aufnahme und vom erwarteten Ergebnis zu betrachten. Eine Aufnahme sollte möglichst kontinuierlich Überschneidungen der Messregionen aufweisen, um erstens semantische Landmarken aus verschiedenen Perspektiven zu sehen und zweitens einem Drift entgegenzuwirken, der bei relativen Lokalisierungsverfahren zwangsläufig auftritt. In der Regel reicht es aus, den Mittelwert der Sichtungen als Schwellwert zu nutzen,

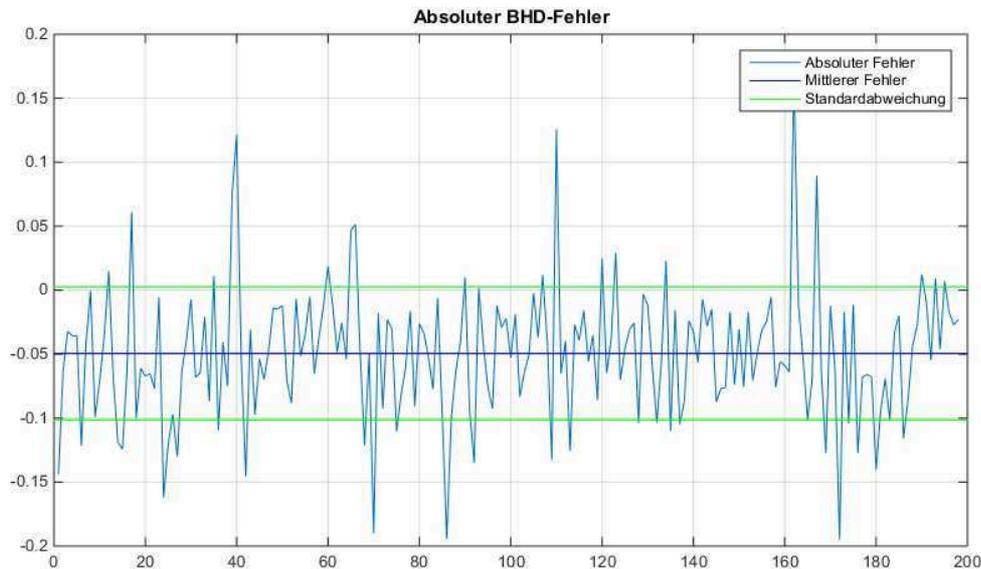


ABBILDUNG 4.22: Absolute Genauigkeitsabweichung der BHD-Messungen im Testgebiet Zollikofen. Der mittlere Fehler ist auf eine systematische Überschätzung des Durchmessers um ca. 5 cm zurückzuführen.

um zwischen *true positives* und *false positives* zu unterscheiden.

Abbildung 4.21 zeigt den Zusammenhang zwischen der Anzahl an Sichtungen und den detektierten Bäumen. Hier ist zu sehen, dass der Mittelwert aller Baumsichtungen (grüne Linie) eine gute Annäherung für die ideale Anzahl an Sichtungen (schwarze Linie) darstellt. Der ideale Wert wurde ermittelt, indem die Differenz zwischen Bäumen mit passendem Matching in den Ground Truth Daten und Bäumen ohne ein Matching maximiert wurde. In der Praxis, wo keine Ground Truth Daten zur Verfügung stehen, ist der Mittelwert eine geeignete Annäherung.

Nach der Kartenerzeugung ist es erforderlich, Paarungen zwischen den aufgenommenen Bäumen und denen aus der Referenzkarte zu bilden („matchen“), um anschließend die jeweiligen Attribute vergleichen zu können. Bei einer eindeutigen Zuordenbarkeit ist dies kein Problem, allerdings können hin und wieder Bäume der Detektion nicht eindeutig den Referenzdaten zugeordnet werden. Um die Qualität der Baumattribuierung möglichst unabhängig von Detektion, Lokalisierung und Kartenerzeugung evaluieren zu können, wurden in der folgenden Auswertung uneindeutige Zuordnungen verworfen.

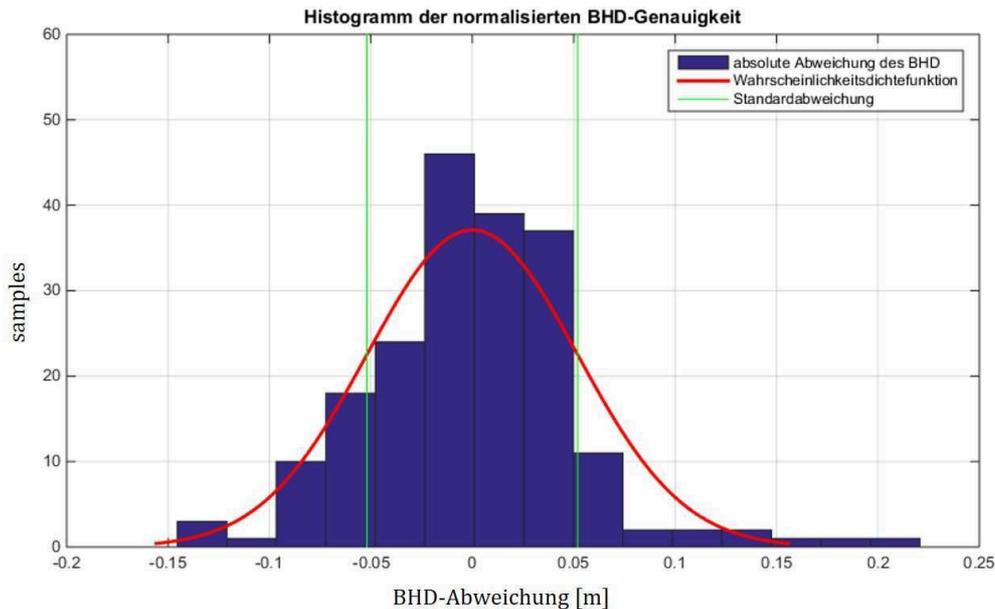


ABBILDUNG 4.23: Das normalisierte Histogramm sowie die angepasste Wahrscheinlichkeitsdichtefunktion zeigen die annähernde Normalverteilung des Fehlers über alle BHD-Messungen.

Werden die Werte der automatischen Baumattribuierung mit den händisch aufgenommenen Werten verglichen, erhält man die in Abbildung 4.22 dargestellte Fehlerverteilung. Man sieht hier eine systematische Abweichung des mittleren Fehlers um etwa 5 cm nach oben, d.h. die Bäume wurden von der SELMA-Einheit, bis auf einige Ausreißer, kontinuierlich dicker angenommen als sie in den Referenzdaten verzeichnet sind. Zum einen lässt sich dieser Unterschied schlicht darauf zurückführen, dass zwischen der Referenzmessung und der automatisierten Aufnahme weitere Wachstumsperioden lagen, in der die Bäume weiter an Durchmesser gewonnen haben. Zum anderen lässt sich an Abbildung 4.20 unten erkennen, dass der zur Durchmesserbestimmung verwendete Algorithmus ab einer Entfernung von etwa sieben Metern den Durchmesser auch bei idealen Kreisschnittflächen geringfügig überschätzt. Da die Bäume in der Regel keinen idealen Kreisquerschnitt aufweisen, kann die Durchmesserbestimmung hier auch Abweichungen von einigen wenigen Zentimetern aufweisen. Im normalisierten Histogramm der BHD-Abweichungen in Abbildung 4.23 lässt sich darüber hinaus eine gauss'sche Fehlerverteilung um den, hier auf Null gelegten, Mittelwert

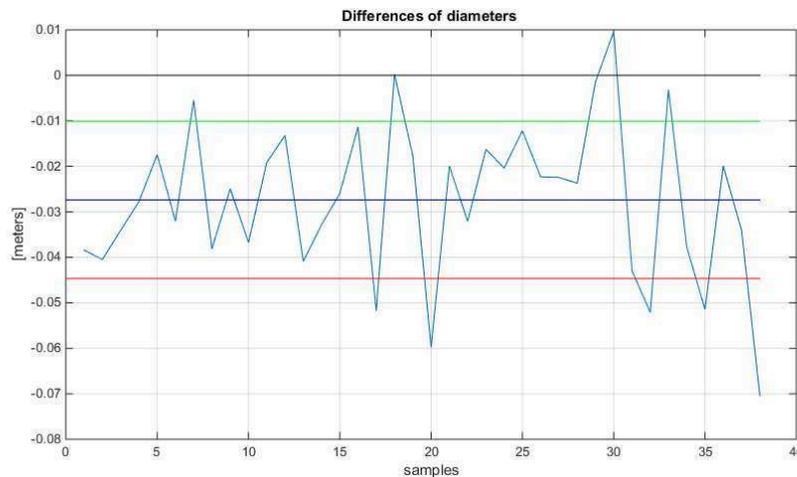


ABBILDUNG 4.24: Absoluter Fehler der BHD-Bestimmung auf Basis von Stereokameradaten (Auflösung 640x480).

erkennen. Die Standardabweichung liegt hier bei ungefähr fünf Zentimetern, sodass die Genauigkeit der Durchmesserbestimmung in einem neu zu erfassenden Gebiet mit +/-fünf Zentimetern angegeben werden kann. Dieses Ergebnis erfüllt ziemlich genau die theoretischen Erwartungen aus Abbildung 4.4 links, woraus sich ein Strahlenabstand von 9 Zentimetern auf einer Entfernung von 10 Metern bei der hier verwendeten Winkelauflösung, ablesen lässt.

Eine Erhöhung der Genauigkeit der Durchmesserbestimmung wäre demnach durch eine Erhöhung der Laserscannerauflösung gegeben. Für die Aufnahmen wurde ein leichtgewichtiger Scanner mit einer Winkelauflösung von 0,25 deg verwendet. Winkelauflösungen von 0,125 deg sind aktuell ebenfalls üblich, aktuell aber nur bei weitaus schwereren und teureren Modellen.

Der Einsatz von Stereokameras zur Durchmesserbestimmung von Bäumen ist aufgrund der höheren horizontalen Messwertauflösung eine sinnvolle Ergänzung. Insbesondere bei der Durchmesserbestimmung in Fichtenreinbeständen sind Genauigkeiten mit Standardabweichungen im Bereich von zwei bis drei Zentimetern möglich (vgl. Abbildung 4.24), insbesondere, da Fichtenstämme in der Regel recht gerade sind und im unteren Bereich des Baumes (im Messbereich der Kamera) wenig Beastung aufweisen.

Durch die mit der Entfernung schnell abfallende Tiefenaufklärung von

Stereokameras kann nicht, wie beim Laserscanner, die Krümmung genutzt werden, um die kreisförmige Schnittfläche zu rekonstruieren, und daraufhin den Durchmesser zu bestimmen. Deshalb hängt die Genauigkeit der Durchmesserbestimmung stark von der vorrangig vertretenen Baumart ab.

### 4.3.2 Attribuierung von Felsen und Steinen

Im Anwendungsgebiet einer planetaren Explorationsmission können, wie bereits diskutiert, Steine als Landmarken herangezogen werden. einen entsprechenden Detektor auf Basis von höhenkodierten Punktwolken aus Kamerasystemen wurde bereits vorgestellt.

Beim behandelten Detektionsalgorithmus wurden alle Punkte der Punktwolke zunächst in Höhencluster eingeteilt und anschließend Cluster zusammengefügt, die wahrscheinlich einen Stein darstellen. Zur Unterscheidung zwischen Steinen und anderweitigen Erhebungen des Bodens wurde ein Kompaktheitsmaß (Gleichung 4.7) eingeführt und herangezogen, das die Größe der sichtbaren Seitenansicht im Bild und der zugehörigen Konturlänge ins Verhältnis setzt. Damit war es möglich, kompakte Steine von langgezogenen Erhebungen erfolgreich zu trennen. Ebenso ist es möglich, zur weiteren Verfeinerung der Detektionsergebnisse, die Höhe und Breite der Steine zu nutzen, um nur Landmarken einer bestimmten Größenklasse nutzen zu können. Die Höhe ist dabei sehr einfach und mit hoher Genauigkeit messbar, da die Bodenebene bereits zum Anfang des Detektionsprozesses geschätzt wurde und die Messpunkte in das Referenzsystem der Ebene transformiert wurden, somit stehen zu jedem Messpunkt auch dessen Entfernung zur Ebene zur Verfügung.

Die Bestimmung der Breite einer Landmarke kann dem gegenüber nicht einfach ermittelt werden, da die Geometrie, nicht wie bei Bäumen, als annähernd symmetrisch angenommen werden kann. Besser ist es eine minimale und maximale gemessene Breite der Landmarke aus allen Beobachtungen zu protokollieren. Dies erlaubt zum einen eine aussagekräftigere Beschreibung der Landmarkengeometrie als lediglich den Mittelwert zu bilden und zum anderen erleichtert es die Aussage, ob ein aktuell beobachteter Stein einer Landmarke aus einer Karte entspricht. Wurde ein Stein bereits aus verschiedenen Richtungen betrachtet, muss die gemessene Breite zwischen der minimalen und maximalen Breite der

Landmarke liegen. Ist dagegen nur der Mittelwert protokolliert worden ist eine solche Aussage mit nur einer Messung schwer zu treffen.

### 4.3.3 Attribuierung von künstlichen Markern

Bei der Verwendung von künstlichen Landmarken kann die Art der Attribuierung mit der Auswahl der zu verwendenden Marker oder Objekte frei bestimmt werden. So verwenden die in Abschnitt 4.2.4 behandelten AR-Marker zur Informationsspeicherung einen Bit-Code der für jeden Marker eine eigene Identifikationsnummer (kurz: Id) kodiert. Aufgrund dieser Ids können Marker, sofern sie nicht mehrfach verwendet werden, eindeutig identifiziert werden. Über die Bittiefe der Marker wird, unter Berücksichtigung der für Fehlererkennung bzw. Fehlerkorrektur reservierte Redundanz, die Anzahl der unterschiedlichen Marker-Ids festgelegt. Berücksichtigt werden muss hier, dass sich die Geschwindigkeit und Zuverlässigkeit der Detektion mit zunehmender Bittiefe verringert.

Anstatt die Bittiefe der zu detektierenden Marker auf Kosten einer schnellen und zuverlässigen Detektion zu erhöhen, kann die Informationsübertragung ebenso von der Detektion entkoppelt werden, wie die in Abschnitt 5.3.1 beschriebene Posenschätzung aus der Markergeometrie. Dazu wird ein einfacher Marker verwendet, der in den Sensordaten detektiert werden soll. Dieser Marker kann auch ein einfacher Reflektor sein, der beispielsweise hohe Reflektanzwerte bei der Nutzung eines licht-emittierenden und -empfangenden Sensors aufweist und dadurch überaus schnell zu detektieren ist. Nach erfolgreicher Detektion des Markers in den Sensordaten, kann nun eine weitere Technik zur Informationsübertragung genutzt werden, auch etwa ein Marker mit höherer Bittiefe, der sich an einer fest vorgegebenen Position relativ zum ersten Marker befindet. Dadurch kann die Informationsübertragung beschleunigt werden, denn der Suchbereich für den zweiten Marker wird durch die erste Detektion entsprechend eingegrenzt.

Mit Hilfe dieser Herangehensweise kann nicht nur die Id eines Markers kodiert und bei der Detektion ausgelesen werden, sondern zusätzlich können noch weitere Informationen gespeichert werden, wie etwa die Koordinaten des Markers in einem übergeordnetem Koordinatensystem (Weltkoordinatensystem), oder eine Karte mit weiteren Markerpositionen

in der Umgebung. Damit wird eine, bezüglich eines globalen Koordinatensystems, absolute Lokalisierung ermöglicht.

Darüber hinaus sind selbstverständlich auch weitere Möglichkeiten zur Datenübertragung denkbar, wie z.B. RFID-Leser oder andere kontaktlose Datenübertragungsarten. Allerdings werden dazu separate Sensoren benötigt, was die Nutzbarkeit wiederum einschränkt. Daher ist eine visuelle Datenübertragung, die mit derselben Sensorik funktioniert wie die Detektion selbst eher zu bevorzugen.



## Kapitel 5

# Landmarkenbasierte Lokalisierung

Da die Lokalisierung wichtiger Bestandteil bei der Erzeugung semantischer Weltmodelle ist, wird nun die Nutzung der semantisch erfassten Objekte zur eigenen Posenbestimmung behandelt. Wie bereits im letzten Kapitel erarbeitet, müssen die erkannten Objekte gewisse Anforderungen erfüllen (vgl. Kapitel 4.1) damit sie als Landmarken zur Lokalisierung geeignet sind. Im Folgenden wird angenommen, dass die erfassten Landmarken diese Anforderungen erfüllen. Nachdem die zur Lokalisierung taugenden Objekte aus den jeweiligen Sensordaten erfasst wurden, stehen sie als sensorunabhängige Instanzen einer abstrakten Landmarkenklasse zur Verfügung, d.h. der zur Erfassung eingesetzte Sensor ist nicht mehr von Bedeutung. Die Unterschiede zwischen Merkmalen im Sensorraum und Objekten im 3D-Raum wurde bereits in Abbildung 3.4 veranschaulicht. Im direkten Vergleich zur Herangehensweise mit sensorabhängigen Lokalisierungsverfahren, wie z.B. Visual Odometry (VO) oder Spatial Mapping Ansätze, lässt sich dieser Gesichtspunkt wie in Abbildung 5.1 dargestellt beschreiben.

Sei in beiden Fällen  $S$  ein Sensor, der zu den Zeitpunkten  $t$  und  $t + 1$  Sensordaten  $d(S_t)$  bzw.  $d(S_{t+1})$  liefert. Im Sensorraum bleibend können jeweils Merkmale  $F_t$  und  $F_{t+1}$  gesucht, und anschließend Paarungen  $M(F_t, F_{t+1})$  zwischen diesen gebildet werden. Die Paarungen entsprechen lediglich einer abgeleiteten Bewegung der Merkmale im Sensorraum. Die tatsächliche Bewegung hat in der 3D-Umgebung stattgefunden, weshalb die Merkmals-Paarungen der Projektion von Einzelbewegung aus der 3D-Umgebung in den Sensorraum entsprechen. Somit muss die anschließende Bewegungsschätzung zwei Aufgaben zugleich

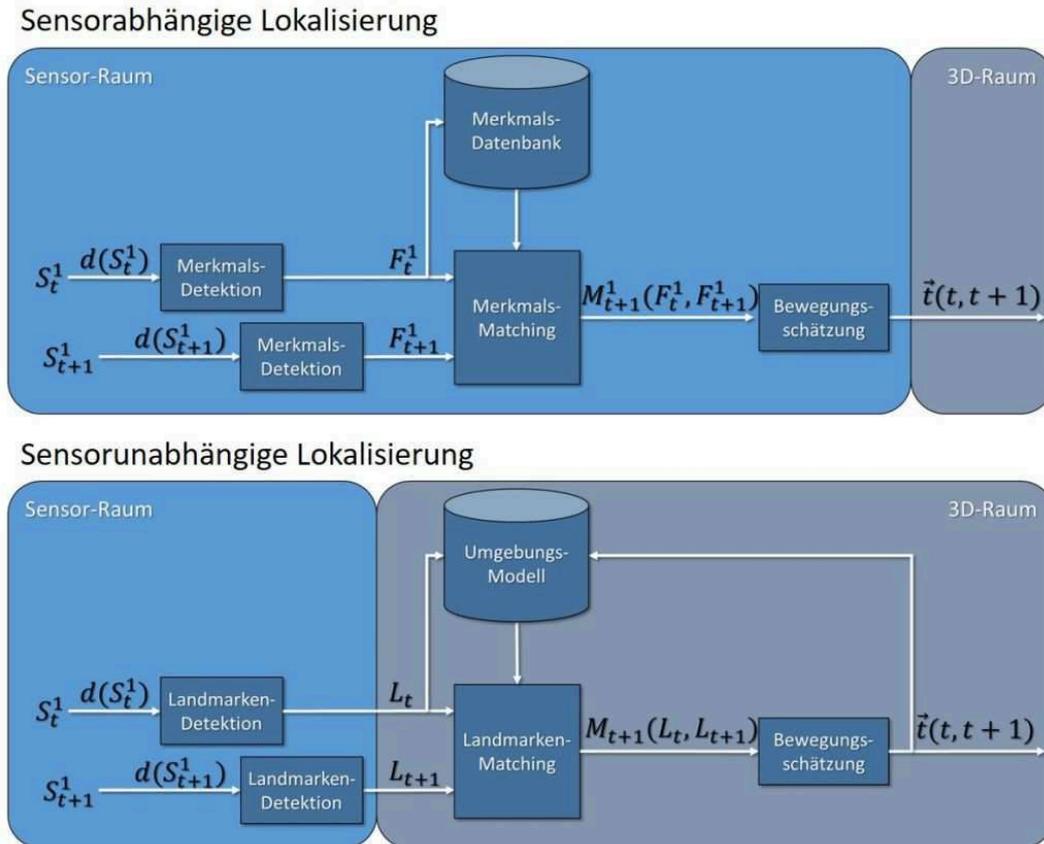


ABBILDUNG 5.1: Gegenüberstellung zwischen der Lokalisierung auf Basis sensorabhängiger Features (oben) und auf Basis von semantischen Landmarken im 3D-Raum (unten)

erfüllen. Zum einen muss die Rückprojektion der Merkmale aus dem Sensorraum in die 3D-Umgebung durchgeführt, und zum anderen die Einzelbewegungen zu einer plausiblen Gesamtbewegung zusammengefasst werden. Wird Entfernungsmesssensorik verwendet, so ist diese Rückprojektion recht einfach und eine rechenaufwändige Rückprojektion auf Basis von Epipolargeometrie, wie bei Kameras, entfällt. Die Schätzung einer Eigenbewegung aus verschiedenen Einzelbewegungen ist aufgrund eines fehlenden Umgebungsverständnisses jedoch fehleranfällig und bedarf einer geeigneten und rechenintensiven Filterung.

Durch die Detektion und Extraktion von 3D-Objekten aus den Sensordaten, um diese als Landmarken  $L_t$  zur Lokalisierung zu nutzen, wird die Grenze zur Sensorabhängigkeit direkt zu Beginn überschritten, d.h.

anschließend werden ausschließlich die von der Aufnahme unabhängigen Landmarken selbst, jedoch nicht mehr die zu Grunde liegenden Sensordaten betrachtet. Liefert der Sensor zum Zeitpunkt  $t + 1$  weitere Sensordaten  $d(S_{t+1})$ , so werden daraus ebenfalls Landmarken  $L_{t+1}$  erzeugt, die mit den Landmarken  $L_t$  in Beziehung gebracht werden können. Ähnlich dem Feature-Matching, entstehen so Paarungen  $M(L_t, L_{t+1})$  von Landmarken, die zu den Zeitpunkten  $t$  und  $t + 1$  erfasst wurden. Aus den Landmarken-Paarungen lassen sich nun unmittelbar räumliche Bewegung in der 3D-Umgebung ablesen, da die Landmarken bereits selbst eine räumliche Darstellung von Umgebungsobjekten sind. Die Rekonstruktion der Eigenbewegung  $\vec{t}(t, t + 1)$  geschieht nun anhand von Objekten in der Umgebung, was aufgrund eines höheren Umgebungsverständnisses bereits einer Filterung entspricht.

Über das Problem der Posenschätzung eines Systems anhand von Landmarken existieren bereits zahlreiche Veröffentlichungen. An dieser Stelle sei nochmal darauf hingewiesen, dass in der Literatur eine sehr weitläufige Verwendung des Begriffs „Landmarke“ existiert. Darauf wurde bereits in Kapitel 3.2 näher eingegangen. Zur Lösung des Problems werden oftmals Ansätze verfolgt, die stark auf das jeweilige Anwendungsszenario, das verwendete mobile System oder die eingesetzten Sensoren zugeschnitten sind. Fokus ist daher zumeist die Optimierung eines bereits bekannten Verfahrens durch die gezielte Abstimmung auf die gegebenen Randbedingungen. Auf diese Weise können Lokalisierungsergebnisse mit hoher Genauigkeit, Zuverlässigkeit oder Geschwindigkeit erzielt werden. Allerdings sind diese Lokalisierungsverfahren meist stark auf einen bestimmten Sensor zugeschnitten oder setzen bestimmte Anforderungen an die mobile Plattform voraus, sodass eine anwendungsübergreifende Nutzung zumindest nur sehr eingeschränkt möglich ist.

Nur durch die Entkopplung des eigentlichen Lokalisierungsverfahrens von jeglichen anwendungsbezogenen Eigenschaften, ist eine allgemeine Herangehensweise möglich. Die semantische Umgebungserfassung und Umweltmodellierung erlaubt genau diese Herangehensweise, da hier nicht direkt auf speziellen Sensordaten und Fahrzeugeigenschaften gearbeitet, sondern auf einer allgemeinen Umgebungsbeschreibung aufgebaut wird. Die Abstraktion der Umgebungserfassung macht dies möglich. So wird die eigentliche Posenschätzung nicht direkt auf den Sensordaten bzw. im Sensordatenraum durchgeführt sondern anhand



ABBILDUNG 5.2: Ein Anwendungsunabhängiges Lokalisierungskonzept, das auf einer landmarkenbasierten Umgebungserfassung operiert.

von Landmarken der semantischen Umgebungsbeschreibung. Abbildung 5.2 verdeutlicht diese Abstraktion des Problems zu einem anwendungsunabhängigen Lokalisierungskonzept.

Für die Lokalisierung anhand semantisch erfasster Objekte stehen mehrere Ansätze zur Verfügung. Die Entscheidung, welches Verfahren angewendet werden soll, hängt dabei vom jeweiligen Anwendungsszenario bzw. von der Umgebung. Dabei ist zu unterscheiden, ob eine Lokalisierung anhand einem einzelnen erfassten Objekt möglich ist, da sich alle Objekte eindeutig von einander unterscheiden lassen (i.d.R. durch die Verwendung künstlicher Marker), oder ob es mehrerer Detektionen

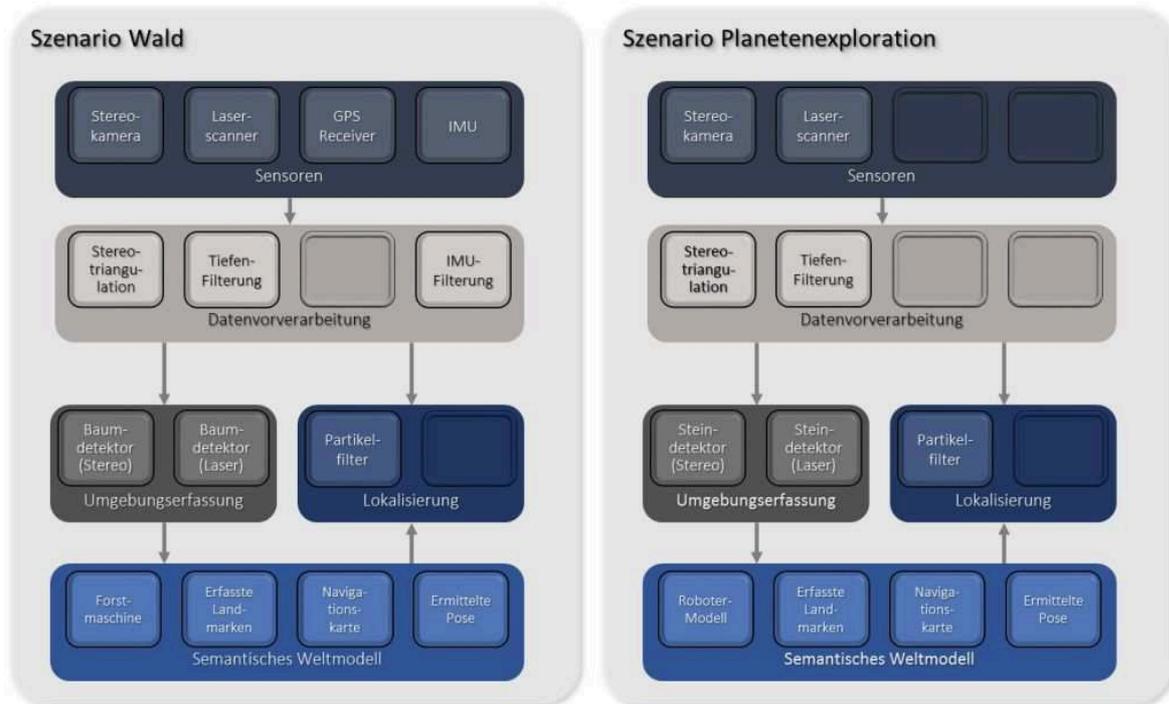


ABBILDUNG 5.3: Adaption des allgemeinen Lokalisierungskonzepts auf zwei Anwendungen

für die Lokalisierung bedarf, da eine eindeutige Unterscheidung der einzelnen Objekte nicht möglich ist (beispielsweise Bäume im Wald), dafür jedoch weitere, räumlichen Informationen herangezogen werden können, um dennoch eine eindeutige Zuordnung zu erreichen. Darüber hinaus ist eine Unterscheidung anhand der Ausgangssituation vorzunehmen, d.h. ob bereits eine Karte für die Lokalisierung vorliegt, ob eine Karte aufgebaut werden, oder ob vollständig auf die Verwendung einer Karte verzichtet werden soll. Stehen die Randbedingungen fest, kann ein geeignetes Lokalisierungsverfahren gewählt werden. Das bereits vorgestellte, allgemeine Konzept erlaubt diese verschiedenen Herangehensweisen, sowohl im Einzelnen als auch in der Kombination.

Daraus ergeben sich je nach Anwendungen bestimmte Verfahren die sich als Komponenten in das allgemeine Konzept per Baukastenprinzip einfügen und miteinander kombinieren lassen. Abbildung 5.3 zeigt dies exemplarisch an den beiden Anwendungen im Wald und bei der Planetenexploration. In beiden Fällen wird derselbe Lokalisierungsalgorithmus verwendet, der im Wald mit Bäumen und bei der Planetenexploration mit Steinen als Landmarken arbeitet. In beiden Fällen werden

die Landmarken sowohl per Laserscanner als auch per Stereokamera detektiert und erkannt.

## 5.1 Sensorfusion

Durch die Abstraktion des Landmarkenbegriffs ist es möglich eine mehrere verschiedene Sensoren simultan zu verwenden, denn für jeden dieser Sensoren können in jeder Aufnahme, unabhängig voneinander, die Landmarken erfasst und weitergegeben werden. Es muss also nicht mit jedem neuen Datenframe eine neue Lokalisierung im jeweiligen Sensorraum durchgeführt und anschließend fusioniert werden. Daraus entstehen gleich mehrere Vorteile:

- Es muss auch bei der Verwendung (vieler) verschiedener Sensoren nur eine gemeinsame Posenschätzung durchgeführt werden.
- Die Methode der Posenschätzung ändert sich nicht durch die Verwendung anderer Sensortypen.
- Die geschätzte Pose bezieht sich bereits auf das Gesamtsystem und nicht auf den jeweiligen eingesetzten Sensor und dessen Koordinatensystem.
- Aus der geschätzten Pose, den lokal erfassten Landmarken und dem Umgebungsmodell ergibt sich automatisch eine Möglichkeit zur simultanen Umweltmodellierung.

Aufbauend darauf lässt sich auf einfache Weise eine Lokalisierung realisieren, die nicht nur einen bestimmten Sensor erlaubt, sondern den flexiblen Einsatz mehrerer, verschiedener Sensoren ermöglicht. Abbildung 5.5 illustriert beide Fälle. Bei der Verwendung der jeweiligen Sensoren als Bezugssysteme der Umgebungserfassung wird jede Landmarke relativ zum jeweiligen Sensor beschrieben, woraus sie detektiert wurde (Abbildung 5.5a)). In diesem Fall wird ein anschließender Fusionschritt notwendig, der die Detektionen bzw. die Bewegungsschätzungen zusammenfügt. Die direkte Beschreibung der Umgebung aus Sicht des Roboters (Abbildung 5.5b)) macht eine zusätzliche Fusion überflüssig. Jeder Sensorpose  $S^i$  wird durch eine entsprechende Transformation  ${}^R T_{S^i}$  relativ zum Roboter  $R$  angegeben.

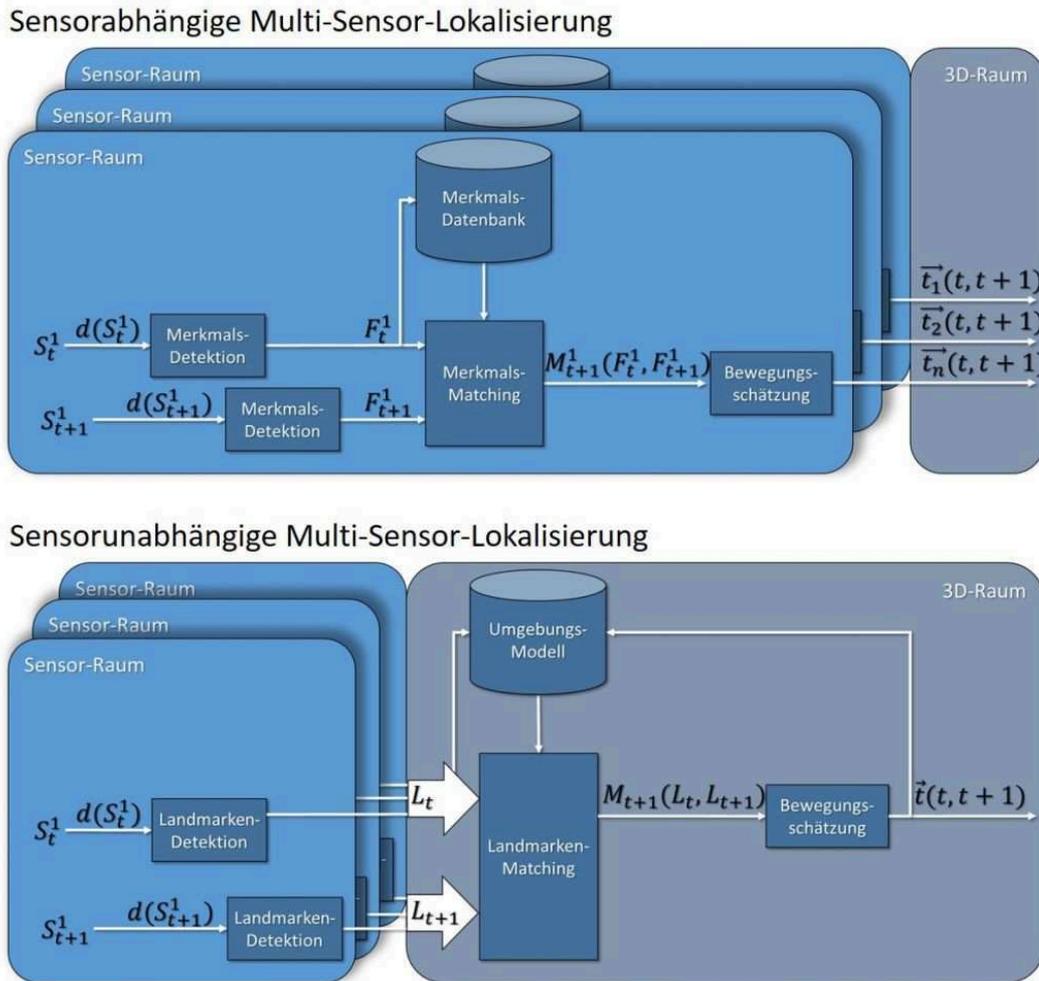


ABBILDUNG 5.4: Gegenüberstellung der sensorabhängigen und sensorunabhängigen Lokalisierung im Multi-Sensor-Betrieb

Da die Transformationen zwischen Sensoren und Roboterzentrum  ${}^R T_{S_i}$  nicht bekannt sind, müssen diese in einem Kalibrierungsschritt zunächst einmalig bestimmt werden. Die Bestimmung lässt sich auf verschiedene Arten realisieren. Wird ein globales Weltkoordinatensystem  $\mathfrak{W}$  angenommen, so sind sowohl die Pose des Roboters  $R_{\mathfrak{W}}$  als auch die Pose der jeweiligen Sensoren  $S_{\mathfrak{W}}^i$  bezüglich dieses Koordinatensystems zu bestimmen, um anschließend  ${}^R T_{S_i}$  zu erhalten aus:

$${}^R T_{S_i} = S_{\mathfrak{W}}^i R_{\mathfrak{W}}^{-1} \quad (5.1)$$

Das bedeutet, die Roboter- und Sensorposen müssen *von außen* gemessen

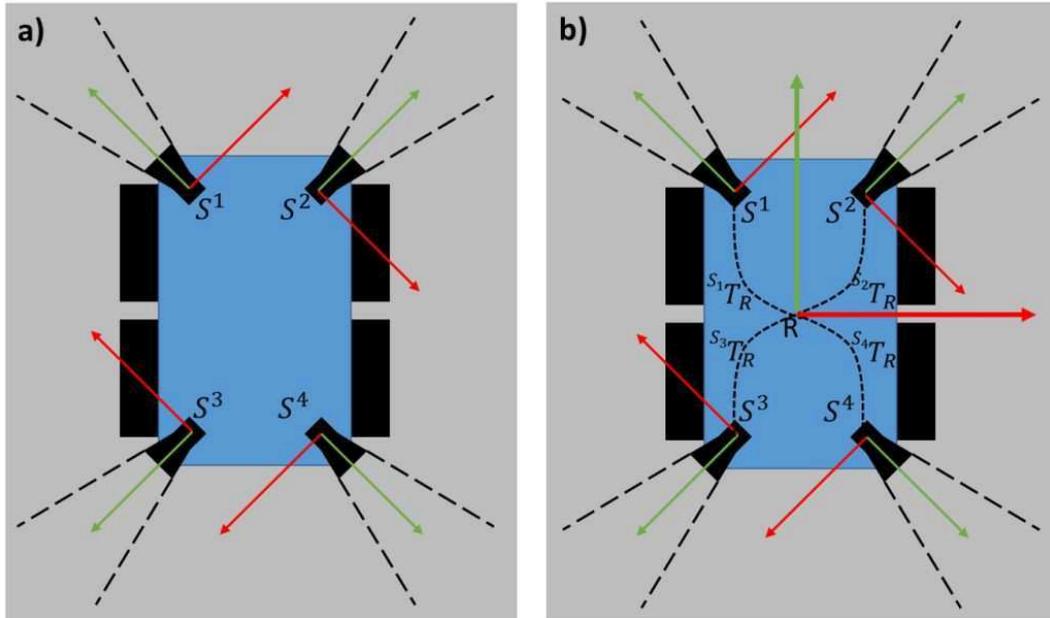


ABBILDUNG 5.5: Nicht die einzelnen Sensoren selbst dienen als Bezugssystem der Umgebungserfassung (a)), sondern der Roboter. Das ermöglicht eine einheitliche Umgebungsbeschreibung aus Sicht des Roboters und ist somit flexibel bezüglich der Anzahl und Art der verwendeten Sensoren.

werden. Dazu muss ein Punkt des Roboters als Zentrum definiert und anschließend dessen exakte Pose eingemessen werden. Schwierig wird die Bestimmung des Ursprungs *von außen* bei den Sensoren, da etwa das optische Zentrum bei Laserscanner- und Kamerasystemen i.d.R. nicht von außen erkennbar ist. Aus diesem Grund bietet sich an, die Kalibrierung der Sensoren *von innen* durchzuführen, d.h. die Sensorposen werden nicht durch Vermessung von außen sondern direkt anhand der Sensordaten ermittelt. Genauer gesagt werden hierzu künstliche Marker  $M^j$  in den Sichtbereich des Sensors eingebracht, die von außen eingemessen werden um deren Posen bezüglich des Weltkoordinatensystems  $M_{\mathfrak{W}}^j$  zu bestimmen. Durch anschließende Erfassung der Marker aus den Sensordaten und Bestimmung der Markerposen im Sensorraum  $M_{S_i}^j$  lassen sich daraus die Sensorposen in  $\mathfrak{W}$  bestimmen durch:

$$S_{\mathfrak{W}}^i = M_{S_i}^j{}^{-1} M_{\mathfrak{W}}^j. \quad (5.2)$$

Überdecken oder überlappen sich die Erfassungsbereiche der Sensoren sogar in ausreichendem Maße, sodass die zur Kalibrierung verwendeten Marker jeweils in mehreren Sensoren erfasst und detektiert werden können, so ist es, unter der Voraussetzung einer geschlossenen Kette (d.h. mit überlappenden Sensorbereichen, sodass alle Sensoren gegeneinander kalibriert werden können) möglich die Sensoren *vollständig von innen* untereinander und ohne Einbeziehung eines Weltkoordinatensystems direkt zu kalibrieren.

$$S_{S_{i+1}}^i = S_{M_j}^i S_{M_j}^{i+1-1} \quad (5.3)$$

Um sich propagierende Fehler bei der verketteten Kalibrierung untereinander zu minimieren, können die Sensoren derart angeordnet werden, dass sie nicht nur eine Kette sondern einen Kreis bilden, sodass neben Gleichung 5.4 noch zusätzlich gilt:

$$S_{S_1}^n = S_{M_j}^n S_{M_j}^1{}^{-1}. \quad (5.4)$$

Auf diese Weise werden benachbarte Sensoren paarweise gegeneinander kalibriert, ohne die eigentliche Pose der Merkmale in einem Referenzkoordinatensystem kennen zu müssen. Durch das Einbringen mehrerer eindeutig voneinander unterscheidbarer Merkmale in den jeweiligen Überlappungsbereich von zwei oder mehr Sensoren, kann die Kalibrierung stabilisiert werden, da sie sich auf gleich mehrere Messungen stützt. Wird darüber hinaus a priori-Wissen bzgl. der Merkmalsverteilung zur Verfügung gestellt, wie etwa bei der Verwendung von AR-Markern die jeweiligen Abstände der einzelnen Marker, so lässt sich ebenfalls eine Stabilisierung der Kalibrierung erreichen, wenn diese Informationen in die Merkmalsdetektion der einzelnen Sensoren einfließt.

Eine geeignete Auswahl der zu verwendenden Merkmale ist hier letztlich entscheidend, insbesondere dann, wenn verschiedene Sensortypen gegeneinander kalibriert werden sollen. So lassen sich AR-Marker gut zur Kalibrierung zweier Kameras nutzen, jedoch nicht, um Kameras gegen Tiefensensoren wie PMD- oder LIDAR-Sensoren zu kalibrieren. In solchen Fällen müssen Merkmale verwendet werden, die sich in beiden Sensortypen detektieren lassen, also konkret in diesem Beispiel Merkmale, die sowohl über hohe Kontrast- als auch Strukturgradienten verfügen. Ein Beispiel hierfür wäre ein dreidimensionaler AR-Marker, wie in Abbildung 5.6 gezeigt.

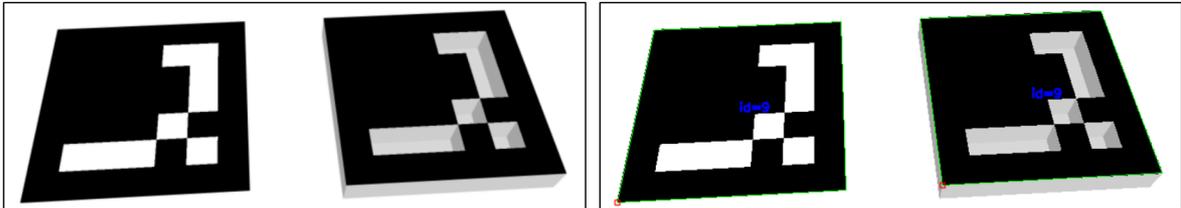


ABBILDUNG 5.6: 3D-Variante eines AR-Markers zur sensorübergreifenden Kalibrierung. Links: Vergleich zwischen 2D- und 3D-Marker, rechts: Ergebnis der bildbasierten Markerdetektion.

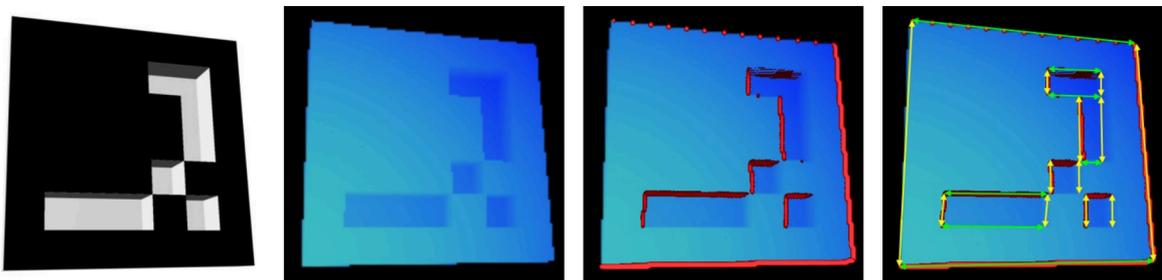


ABBILDUNG 5.7: 3D-AR-Marker-Detektion mittel ToF-Kamera. Links: 3D-AR-Marker, 2.v.l.: Tiefenbild des Markers, 2.v.r.: Gradientenbildung, rechts: Parallelverschiebung der detektierten Kanten.

Der hohe Bildgradient an den Übergängen zwischen schwarz und weiß fällt auf den ebenso hohen Gradienten der Tiefenwerte zwischen Vordergrund (schwarz) und Hintergrund (weiß). Die Abbildung des 3D-AR-Markers in einer ToF-Kameras ist in Abbildung 5.7 dargestellt. Durch eine Bildglättung und anschließende Gradientenbildung erhält man die Kanten, die in den Sensordaten die größten Tiefsprünge verursachen. Dieses Beispiel funktioniert bestens bei einer frontalen Beobachtung, bei erhöhter Neigung des Betrachtungswinkels werden die Übergänge zwischen Vordergrund und Hintergrund zumindest auf der entgegengesetzten Seite zur Kamera abgeschwächt. Durch eine Parallelverschiebung der starken Kanten innerhalb des Markers, entlang ihrer orthogonalen, benachbarten Kanten, wird jedoch auch eine Rekonstruktion der schwachen Kanten ermöglicht.

Eine andere Möglichkeit ist die Verwendung von Materialien mit unterschiedlichen Reflektanzeigenschaften sodass bei den Übergängen ein deutlicher Sprung im Empfangssignal detektierbar ist. Dies erfordert jedoch eine ToF-Kamera, die die Aufnahme von Intensitätswerten erlaubt.

## 5.2 Allgemeines Landmarkentracking

Im Gegensatz zu künstlichen Markern, die meist direkt eine eindeutige Identifikation anhand einer eingebetteten Id erlauben, sind natürliche Landmarken nicht so einfach von einander zu unterscheiden. Durch die Erfassung zusätzlicher Erscheinungsmerkmale oder beschreibender Attribute, kann eine Unterscheidung zwar begünstigt werden, jedoch ist eine eindeutige Erkennung und damit die Möglichkeit zur Lokalisierung anhand nur einer Landmarke nur in Fällen möglich, wo die Merkmale der Landmarke Rückschlüsse über Entfernung und relative Ausrichtung zur Kamera ermöglichen, wie etwa bei den in Kapitel 4.2.4 behandelten künstlichen Markern. Für den Fall, dass Landmarken nur bedingt oder überhaupt nicht von einander unterschieden werden können, muss der Lokisierungsalgorithmus die notwendigen Informationen selbst aufbringen, um die erfassten Landmarken von einem Zeitpunkt zum Nächsten zu verfolgen. Nimmt man beispielsweise an, dass sich die Eigenbewegung des Sensors im Verhältnis zu seiner Aufnahmegeschwindigkeit verhältnismäßig langsam bewegt, und die relative Translation der Landmarken im Sensorraum in einem Zeitschritt  $\delta t$  geringer ist, als der Abstand dieser Landmarke zu ihren Nachbarn, so lässt sich eine Landmarke über die Zeit verfolgen oder *tracken*, indem man von ihrer Position zum Zeitpunkt  $t_i$  im Sensorraum die nächstgelegene Landmarke zum Zeitpunkt  $t_{i+1}$  sucht. Jedoch ist diese Einschränkung nicht immer leicht zu erfüllen. Beispielsweise führt die Rotation eines mobilen Systems, das sich in einer Ebene fortbewegt, dazu, dass Landmarken mit zunehmender Entfernung zum Sensor durch diese Drehung zu großen Translationen der Landmarken im Sensorraum führen. Das liegt daran, dass sich bei einer reinen Rotation des Sensorsystems um einen Winkel  $\phi$  Landmarken mit einer Entfernung von  $d$  Metern um  $2d \cdot \sin(\phi/2)$  verschieben. Bei einem Winkel von  $\phi = 6^\circ$  und einer Entfernung von  $d = 10\text{m}$  entspricht das bereits einer Verschiebung um einen Meter.

Alternativ können gleichzeitig mehrere Landmarken betrachtet werden. Eine Möglichkeit hierfür ist die Betrachtung der relativen Lage der Landmarken zueinander, z.B. die paarweisen Abstände oder Winkel zwischen Verbindungslinien. Unter der Voraussetzung, dass sich die Landmarken selbst nicht bewegen und somit die relativen Beziehungen untereinander fix bleiben, lassen sich hier mehrere Landmarken zu Mustern zusammenfassen, die als Gruppe und somit nicht mehr einzeln

getrackt werden. Durch diese Maßnahme kann ein im Gegensatz zum Verfolgen einzelner Landmarken, robusteres Tracking realisiert werden, da ein Trackingschritt auf der Kombination mehrerer Messungen<sup>1</sup> beruht und nicht nur auf einer einzigen. Somit können Fehler, die bei der Messung ggf. auftreten, bereits implizit herausgefiltert werden.

Sind nun mehrere Objekte über die Zeit und während der Bewegung des Roboters detektierbar, so kann die relative Bewegung des Roboters anhand dieser Beobachtungen rekonstruiert werden. Dazu sind zwei wesentliche Aufgaben zu erfüllen. Die erste Aufgabe besteht darin, die Beobachtungen zum Zeitpunkt  $t_0$  mit denen vom Zeitpunkt  $t_1$ , mit  $t_1 = t_0 + \Delta t$  wieder zu finden oder zu *matchen*. Setzt man voraus, dass die Zeitpunkte der jeweiligen Detektionen in  $t_0$  und  $t_1$  so dicht an einander gewählt sind, dass die Bewegung des Roboters zwischen diesen Zeitpunkten ebenfalls gering ausfällt, so lässt sich das Matching der Detektionen dahingehend vereinfachen, dass lediglich im näheren Umfeld der individuellen Landmarken  $L_{t_0}$  zum Zeitpunkt  $t_0$  nach den korrespondierenden Landmarken  $L_{t_1}$  zum Zeitpunkt  $t_1$  gesucht werden muss. Die zweite Aufgabe besteht dann darin, die Bewegung aus mehreren Landmarkenkorrespondenzen zwischen den beiden Zeitpunkten  $t_0$  und  $t_1$  zu rekonstruieren.

Das Problem lässt sich einfach anhand Abbildung 5.8 zusammenfassen und verdeutlichen. Dabei beschreibt der linke Teil das Problem als Ganzes, also die unbekannte Bewegung des Roboters im Referenzsystem und die resultierenden Landmarkenbeobachtungen zu den jeweiligen Zeitpunkten, die im rechten Teil von Abbildung 5.8 noch einmal separat dargestellt sind. Die Landmarkenkorrespondenzen sind rechts durch gestrichelte Linien dargestellt. Sowohl die Korrespondenzen als auch die Bewegung des Roboters sind unbekannt und müssen bestimmt werden. Das Matching lässt sich vereinfachen, wenn es gelingt die Zeitabstände derart zu verkleinern, dass die relative Landmarkenbewegung zum Roboter gering ist, vor Allem sollte der Betrag der Verschiebung nicht größer sein, als der mittlere Abstand zwischen den einzelnen Landmarken. Im Grenzfall ist der Zeitabstand derart klein gewählt, dass keine Bewegung zwischen den Roboterposen  ${}^R P_{t_0}$  zum Zeitpunkt  $t_0$  und  ${}^R P_{t_1}$

---

<sup>1</sup>Hier wird die Detektion einer Landmarke als Messung verstanden, d.h. wenn in den Daten einer Sensoraufnahme  $n$  Landmarken erfasst werden können, so werden diese auch an dieser Stelle auch als  $n$  Messungen verstanden

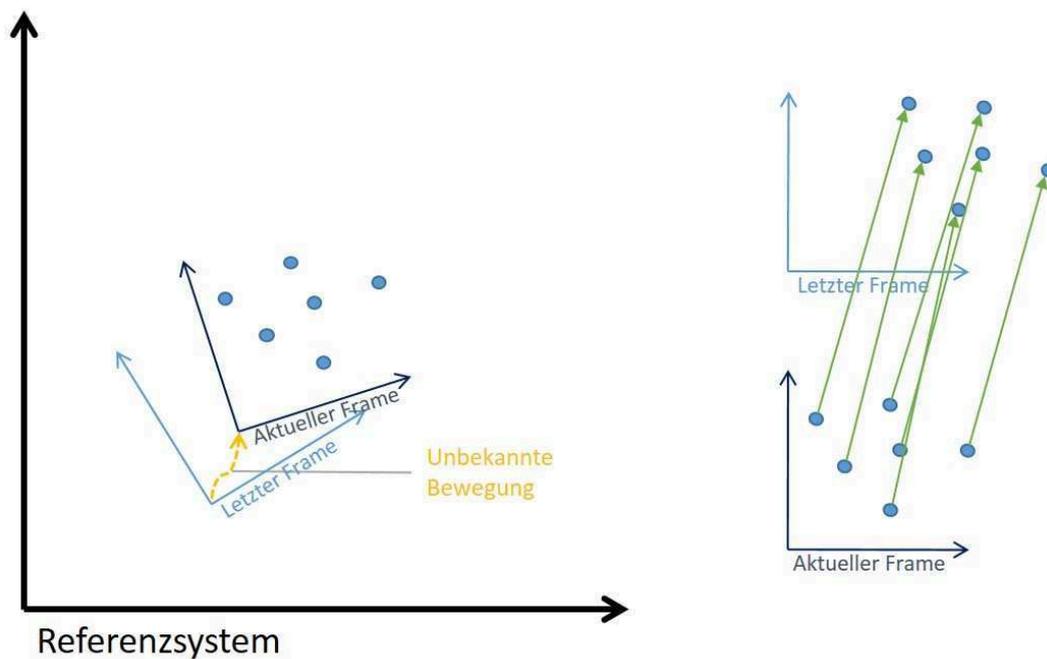


ABBILDUNG 5.8: Illustration des Problems der Bewegungsrekonstruktion (links) durch Landmarkentracking (rechts)

zum Zeitpunkt  $t_1$  stattfindet, also

$$d = \lim_{\Delta t \rightarrow 0} {}^R P_{t_1} - {}^R P_{t_0} = 0.$$

Dann befinden sich die beobachteten Landmarken  $L_{t_1}$  zum Zeitpunkt  $t_1$  an derselben Stelle wie die zuvor betrachteten Landmarken  $L_{t_0}$  zum Zeitpunkt  $t_0$ . In der Praxis wird man aus Performance- und Genauigkeitsgründen Zeitintervalle  $\Delta t$  wählen, die ausreichend klein sind, um den Suchraum für die Korrespondenzsuche gering zu halten, aber auch groß genug, um die Anzahl der Berechnungen zu beschränken. Bei relativen Lokalisierungsverfahren, wie bei diesem Ansatz, muss damit gerechnet werden, dass bei jeder neuen Bewegungsschätzung Fehler entstehen, die sich mit der Zeit akkumulieren.

Nach der Korrespondenzenbildung kann nun jeweils die relative Bewegung der Landmarken für den Zeitschritt  $\Delta t$  wie folgt ermittelt werden. Für alle Landmarkenpositionen in lokalen Koordinaten  $l \in L_{t_0}$  und  $l' \in L_{t_1}$  gilt:

$$Rl + t = l', \quad (5.5)$$

wobei  $R$  die relative Rotation und  $t$  die zugehörige Translation der Landmarken in der Zeit zwischen  $t_0$  und  $t_1$  bezeichnet, mit

$$R = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix}, \quad t = (u, v)^T. \quad (5.6)$$

Diese Transformation kann mit ausreichend vielen korrespondierenden Landmarken mittels Methode der kleinsten Quadrate (*least-squares-Verfahren*) aufgelöst werden. Dazu wird ein überbestimmtes, lineares Gleichungssystem der Form  $Ax = b$  aufgestellt, mit

$$A = \begin{pmatrix} p_x^1 & -p_y^1 & 1 & 0 \\ p_y^1 & p_x^1 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ p_x^n & -p_y^n & 1 & 0 \\ p_y^n & p_x^n & 0 & 1 \end{pmatrix}, \quad (5.7)$$

wobei  $p_x^i$  und  $p_y^i$  die  $x$ - oder  $y$ -Koordinate der  $i$ -ten beobachteten Landmarke, mit  $i \in 1..n$  und  $n$  der Anzahl der Matches zwischen den Landmarken  $l' \in L_{t_1}$  und Landmarken  $l \in L_{t_0}$ . Wenn  $b$  gegeben ist durch

$$b = (l_x^1, l_y^1, \dots, l_x^n, l_y^n)^T,$$

so stellt  $x \in \mathbb{R}^4$  die Lösung des Gleichungssystems dar in der Form

$$x = (\cos(\alpha), \sin(\alpha), u, v)^T.$$

Das überbestimmte Gleichungssystem lässt sich dann beispielsweise durch eine Singulärwertzerlegung lösen. Das Ergebnis ist die Bewegung im Zeitintervall  $[t_0, t_1]$ , das auf Basis des zuvor ermittelten Matching den geringsten quadratischen Fehler aufweist. Da die Bewegungsschätzung sehr stark an die Qualität des Matchings geknüpft ist, leidet die Genauigkeit an fehlerhaften Korrespondenzen. Daher empfiehlt es sich ein Verfahren zur Ausreißer-Detektion und Eliminierung zu verwenden, um möglichst keine Ausreißer als Berechnungsgrundlage zu verwenden. Idealerweise sollten Ausreißer direkt schon bei der Korrespondenzsuche vermieden werden. Wenn eine Zuordnung allein auf räumlichen Informationen (s.o.) dafür nicht ausreicht, können weitere Attribute der

Landmarken (siehe Abschnitt 4.3) zur Differenzierung verwendet werden. Ebenso kann eine Bewegungsprädiktion auf Basis von vorherigen Bewegungsberechnungen verwendet werden, um die vermuteten Posen der korrespondierenden Landmarken zu schätzen und die Suche zu vereinfachen. Da die Rotation Ursache für große Translationen von weit entfernten Landmarken ist, kann eine IMU, oder ein Gyroskop dazu verwendet werden, diese Rotationen zu messen und diese in die Prädiktion direkt einfließen zu lassen. Bei der Annahme einer Messung mit hoher Genauigkeit, kann das Problem somit auf eine Translation zurückgeführt werden, was eine zusätzliche Bewertung der Korrespondenzen ermöglicht, da nun alle Verbindungslinien zwischen korrespondierenden Landmarken möglichst parallel und dieselbe Länge aufweisen müssten. Wenn eine fehlerfreie Korrespondenzenbildung nicht sichergestellt werden kann, bietet sich ein RANSAC-Ansatz an, der randomisiert Untermengen der gefundenen Korrespondenzen zur Berechnung der Bewegung auswählt und diese am Komplement der Untermenge validiert. Mit Hilfe einer geeigneten Bewertungsfunktion können auf diese Weise Ausreißer von korrekten Matchings getrennt und die gewählte Untermenge aufgrund des Verhältnisses zwischen Inlier und Outlier bewertet werden. Das Abbruchkriterium ist dann eine festzusetzende Bewertungsschwelle, die überschritten werden muss oder eine maximale Anzahl an Iterationen so, dass, wenn diese erreicht wird, das bis dahin beste Ergebnis verwendet wird. Die Verwendung einer unteren Grenze für die Güte der Schätzung sei hier empfohlen, damit fehlerhafte Detektionen in der Umgebungserfassung nicht auch zu falschen Bewegungsannahmen führen, damit kein Abriss der Detektionskette erfolgt und daraufhin keine weitere relative Lokalisierung mehr möglich ist. Dann müsste durch ein absolutes Verfahren zunächst versucht werden, eine aktuelle Pose zu bestimmen von welcher durch Landmarkentracking erneut fortexploriert werden kann.

## 5.3 Markerbasierte Lokalisierung

Die zur Lokalisierung günstigste Ausgangssituation ist die Verwendung künstlicher und eindeutig unterscheidbarer Landmarken. Die Detektion von AR-Markern wurde bereits in Kapitel 4.2.4 behandelt. Für eine lückenlose Lokalisierung müssten genügend Marker in der Umgebung

vorhanden sein, sodass permanent ein oder mehrere Marker von der Kamera erfassbar sind. Eine derartige Realisierung ist offensichtlich nur in beschränkten Szenarien möglich und eignet sich daher vor allem für Indoor-Lokalisierung in Gebäuden oder Lagerhallen. Durch die Verwendung von Kameras mit Fischaugen-Objektiven oder Verbunden aus mehreren Kameras lässt sich der erfassbare Bereich zwar erhöhen, jedoch können Verdeckungen und Unterschiede in der Beleuchtung noch immer dazu führen, dass nicht durchweg mindestens ein Marker erfassbar bleibt. Dies ist dann für die jeweilige Anwendung zu berücksichtigen und sollte ggf. ausgemessen werden. Ist hingegen eine lückenlose Lokalisierung nicht zwingend erforderlich, reicht eine weniger dichte Verteilung von Markern aus, was die Anwendbarkeit auch in größeren Anwendungsszenarien erhöht. Dies wäre beispielsweise für ein Besucherleitsystem in öffentlichen Gebäuden ausreichend, wo man bei Bedarf die aktuelle Position mit Hilfe dieser Technik bestimmen kann. Die markerbasierte Lokalisierung kann kombiniert mit anderen Verfahren aber auch für eine initiale Ortsbestimmung eines Systems genutzt werden, welches anschließend einen relativen Lokalisierungsansatz verwendet, um sich von dieser Startposition fortzubewegen. Ein Beispiel hierfür wird in Kapitel 7.1.1 vorgestellt. In diesem Fall dient die markerbasierte Lokalisierung zur Bestimmung eines absoluten Fixpunktes für eine relative Exploration von diesem Startpunkt aus. Sobald erneut Marker in den Sichtbereich der Kamera fallen, lassen sich diese als absolute Stützstellen nutzen, um den „Drift“, der bei relativen Lokalisierungsverfahren typischerweise auftritt, zu eliminieren.

### 5.3.1 Posenschätzung durch Rückprojektion bekannter Geometrien

Ist die Geometrie des abgebildeten und detektierten Markers bekannt, so lassen sich dessen Eckpunkte zurückprojizieren in den dreidimensionalen Raum und daraus die Translation  $\vec{t} \in \mathbb{R}$  und Rotation  $R \in \mathbb{R}^{3 \times 3}$  des Markers relativ zum Referenzkoordinatensystem der Kamera bestimmen. In der Schreibweise als homogene Koordinaten erhält man den Frame des Markers  $M$  relativ zum Referenzkoordinatensystem der Kamera  $C$  als

$${}^C F_M = \begin{pmatrix} R & t \\ 0 & 1 \end{pmatrix}.$$

Ist wiederum die Pose der Kamera in Weltkoordinaten bekannt, so lässt sich diese ebenfalls als Frame  $F_{C_W}$  ausdrücken und eine Multiplikation beider Frames ergibt die Weltkoordinaten des Markers

$${}^W F_M = {}^W F_C \cdot {}^C F_M. \quad (5.8)$$

Diese Sicht stimmt mit der eigentlichen Anwendung der Marker für Augmented Reality-Anwendungen überein, wo an die Stelle des Markers eine künstlich erzeugte (3D-)Geometrie angezeigt werden kann. Für die Verwendung der Marker zur Selbstlokalisierung ist genau die gegenteilige Sicht zu betrachten, also die Posen der Marker in der Umgebung  ${}^W F_M$  sind bekannt und die Kamerapose  ${}^W F_C$  ist gesucht. Durch das Tauschen der Frames von Marker und Kamera in Gleichung 5.8 erhält man die Pose der Kamera durch Invertierung der relativen Lage des Markers zur Kamera  ${}^M F_C = {}^C F_M^{-1}$  und der bekannten Pose des Markers, beispielsweise aus einer Karte

$${}^W F_C = {}^W F_M \cdot {}^M F_C.$$

Weil diese Methode lediglich die Eckpunkte des Markers in der Projektion, d.h. im Bild, und die Kantenlänge des Markers verwendet, ist dieses Verfahren ohne weiteres auch für andere bekannte Objekte anwendbar. Beispielsweise können auch Verkehrsschilder nach demselben Prinzip verwendet werden. Benötigt wird ein Detektionsalgorithmus, der die Außenkanten der Schilder detektiert und anschließend erkennt, um welche Art von Schild es sich handelt. Der Schildtyp (Vorfahrt, Stopp, Bushaltestelle, etc.) gibt Aufschluss über die Kantenlängen, bzw. bei Kreisen über den Durchmesser. Zusammen mit der erfassten Geometrie lässt sich nun auch wieder die relative Lage des Schildes zur Kamera ermitteln und so weiter.

Das zuletzt beschriebene Szenario greift implizit gleich zwei Punkte auf, die nun explizit und im Detail betrachtet werden sollen. Zum Einen ist dies die für die Klassifizierung der Verkehrsschilder verwendete Attribuierung von Landmarken, also der Beschreibung von detektierten Instanzen einer Landmarkenklasse. Zum Anderen wurden die detektierten bzw. a priori bekannten Geometrieinformationen der Landmarken dazu verwendet, die Lage der Kamera zu ermitteln.

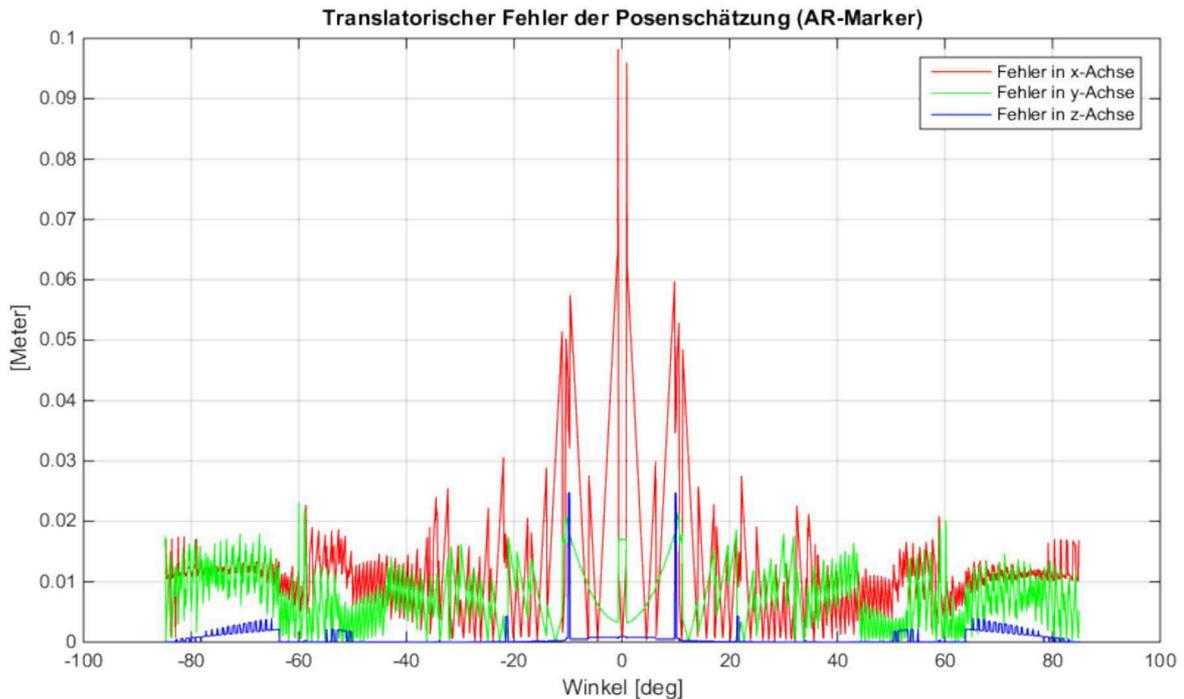


ABBILDUNG 5.9: Translatorische Abweichung in der Posenschätzung von AR-Markern

### 5.3.2 Ergebnisse der markerbasierten Lokalisierung

Unmittelbar mit der Detektierbarkeit von zweidimensionalen Markern (siehe Kapitel 4.2.4) ist die Posenschätzung aus detektierten AR-Markern verbunden. Hier ist zu beachten, dass eine korrekte Detektion des Markers nicht gleichbedeutend ist mit einer korrekten Posenschätzung. Während eine erfolgreiche Detektion lediglich die Erfassung und Erkennung des Markers im Kamerabild beinhaltet, gehört zur Posenschätzung eine korrekte Erkennung der perspektivischen Verzerrung des Markers durch die Projektion in die Bildebene. Gerade bei geringen perspektivischen Verzerrungen können aufgrund des diskreten Pixelrasters im Bild Singularitäten entstehen, die zu fehlerhaften Ergebnissen in der Posenschätzung führen. In [144] wird dieses Problem zwar erkannt und auch eine Erweiterung der 2D-Marker durch die Verwendung einer Mikrolinsenfolie vorgeschlagen, um dem Problem zu begegnen, jedoch wird das Problem an sich nicht weiter quantifiziert.

Durch die Verwendung der Simulationsumgebung soll eine Quantifizierung des Posenschätzungsfehlers an dieser Stelle nachgeholt werden.

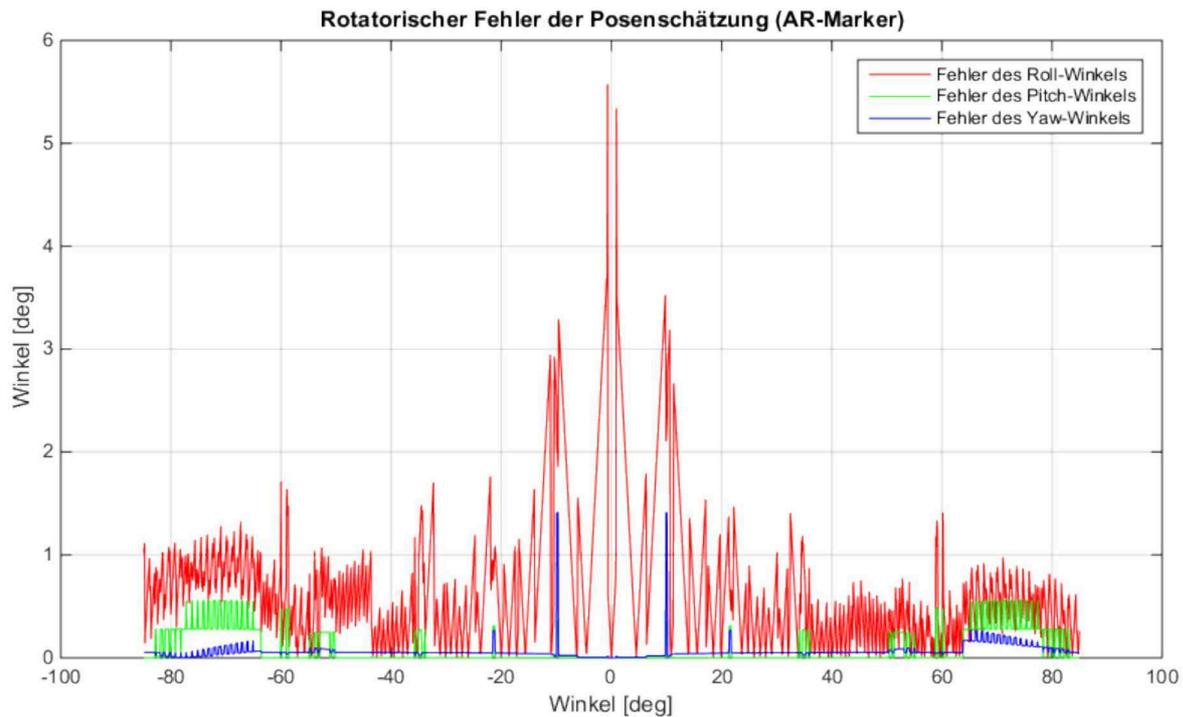


ABBILDUNG 5.10: Rotatorische Abweichung in der Posenschätzung von AR-Markern

Mit demselben Testaufbau, wie in Kapitel 4.2.4, wurde eine simulierte Kamera um einen Marker bewegt und die berechnete Pose mit der in der Simulation bekannten, tatsächlichen Pose verglichen. Abbildung 5.9 zeigt die translatorische Abweichung und Abbildung 5.10 die rotatorische Abweichung in der Posenschätzung des Markers aus dem Kamerabild. Deutlich erkennbar ist, dass sich die Genauigkeit der Posenschätzung umgekehrt proportional zur Größe des Betrachtungswinkels verhält. Ein wichtiges Ergebnis ist, dass sich der Zuverlässigkeitsbereich im Randbereich, d.h. unter höheren Betrachtungswinkeln, nur wenig unterscheidet, wobei die Zuverlässigkeit bei der Annäherung an den Frontalbereich des Markers drastisch absinkt und dort einen großen Unterschied zur Detektierbarkeit aufweist, die gerade in diesem Bereich maximal ist. Daraus ergibt sich, dass Marker wenn möglich derart angebracht werden sollten, dass eine frontale Betrachtung ausgeschlossen oder zumindest selten auftreten sollte. Im Anwendungsfall der Indoor-Lokalisierung für mobile Roboter bedeutet das beispielsweise, dass die Marker möglichst nicht auf Höhe der Kamera angebracht werden sollten. Im Fall einer fest montierten Kamera kann damit auf einfache Weise dieser Problemfall

ausgeschlossen werden, indem die optische Achse der Kamera parallel zum Boden ausgerichtet wird und die Marker oberhalb dieser Achse angebracht werden.

## 5.4 Monte Carlo Lokalisierung (MCL)

Ein auf die Verfügbarkeit einer Umgebungskarte angewiesener, absoluter Lokalisierungsansatz stellt die in der Robotik seit geraumer Zeit verbreitete Monte Carlo Lokalisierung (MCL) dar (siehe Kapitel 2.3.2). Ein wesentliches Merkmal der sequenziellen Monte-Carlo-Methode (SMC) ist die stochastische Herangehensweise, die keine globale, optimale Lösung garantiert sondern lediglich ein Ergebnis, das die zugrunde liegenden Messungen mit hoher Wahrscheinlichkeit rechtfertigt. Oftmals gehen aus diesem Ansatz gleich mehrere unterschiedliche Lösungen mit gleichen Wahrscheinlichkeiten hervor, die im Verlauf des sequenziellen Verfahrens weiterhin betrachtet und fortwährend anhand folgender Messungen bewertet werden, sodass es mit der Zeit gegen die korrekte Lösung konvergiert.

### 5.4.1 Partikelfilter

Eine explizite Filterung bietet oftmals bessere Ergebnisse, als implizite Verfahren wie oben beschrieben. Ein in der Literatur oft beschriebener Ansatz ist die Nutzung eines Partikelfilters für dieses Problem [155]. Damit werden parallel viele verschiedene Hypothesen betrachtet, einzeln bewertet und die jeweils beste Lösung verwendet. Die neuen Hypothesen werden dann im nächsten Schritt gewichtet nach den Bewertungen des vorherigen Schritts entweder beibehalten oder verworfen und in der Nähe<sup>2</sup> von besser bewerteten Hypothesen verteilt. Somit beeinflussen die aktuellen Bewertungen die zukünftige Hypothesen dahingehend, dass ab einem Zeitpunkt alle Partikel um bestenfalls ein Maximum ausgestreut werden, also der Partikelfilter konvergiert. Konkret für den Anwendungsfall des zuvor behandelten Landmarkentrackings können damit verschiedene, bewegungsabhängige Prognosen aufgestellt werden, welche die Lokalisierung stark verbessern können. Mit der Hilfe von Partikelfiltern können bei jedem neuen Berechnungsschritt neben der

---

<sup>2</sup>hier ist der euklidische Abstand des Hypothesenraums gemeint

reinen Translationen auch parallel mehrere Winkel in Betracht gezogen werden und damit auch Landmarken, die für ein Matching eigentlich schon zu weit von ihrer letzten Position abgewichen sind. Da die Partikel in jedem Schritt neu bewertet werden, können auch abrupte Bewegungsänderungen schnell erfasst und auf diese reagiert werden.

### 5.4.2 Kalmanfilter

Ein weiterer Filter, der häufig im Kontext der Lokalisierung Verwendung findet ist der Kalmanfilter bzw. der nichtlineare oder erweiterte Kalmanfilter. Letzterer hat sich bereits nahezu als Standard für die Nutzung von satellitengestützten Navigationssystemen in Fahrzeugen etabliert, da mit Hilfe dieses Filters die nichtlineare Bewegung eines Fahrzeugs sehr gut nachgebildet werden kann.

Für eine landmarkenbasierte Lokalisierung gibt es grob zwei unterschiedliche Herangehensweisen. Die eine ist das Detektieren und anschließende Verfolgen der Landmarken (Tracking), was außer einer zuverlässigen Detektion keine zusätzlichen Informationen der Umgebung benötigt, und das andere ist die absolute Lokalisierung mit Hilfe einer a priori gegebenen Karte, wo alle Landmarken bereits eingetragen sind und die Detektionen mit der Karte abgeglichen werden, um die Pose zu rekonstruieren von wo aus die Beobachtungen gemacht wurden. Diese beiden Herangehensweisen lassen sich auch gut kombinieren, was eine simultane Lokalisierung und Kartenerzeugung (SLAM) erlaubt.

Zunächst soll auf die relative Lokalisierung durch Landmarken-Tracking eingegangen werden. Die Überlegung ist hier, die Bewegung der detektierten Landmarken relativ zum Sensor zu ermitteln und mit Hilfe der Voraussetzung, dass Landmarken stationär an einer Stelle verbleiben, die tatsächliche Bewegung des Sensors zu rekonstruieren.

Das Tracking von Landmarken hat den Vorteil, dass über die Zeit semantische Objekte im Raum verfolgt werden und keine losen Sammlungen von Merkmalspunkten, wie beim Featuretracking. Die Objekterfassung bietet eine zuverlässigere Detektion über die Zeit und verhindert damit die hohe Anzahl an Ausreißern, wie sie beim Featurematching üblich ist. Außerdem werden die Objekte bereits in Weltkoordinaten an die Lokalisierung übergeben und erlauben auf diese Weise eine Bewegungsschätzung im Bezugsraum des Roboters anstatt einer Merkmalsbewegung im jeweiligen Sensorraum. Wurde die Bewegung der Landmarke

relativ zum Sensor erfasst so lässt sich durch eine einfache Transformation daraus direkt die Bewegung des Sensors relativ zur Landmarke ermitteln. Um daraus die Bewegung des Roboters zu rekonstruieren muss die Position aller Sensoren am Roboter bestimmt werden. Dazu ist eine einmalige Sensorkalibrierung erforderlich.

### 5.4.3 Ergebnisse der MCL mittels Partikelfilter

Im Folgenden sollen die Ergebnisse der landmarkenbasierten Monte-Carlo Lokalisierung vorgestellt werden. Die Lokalisierung wurde in zwei unterschiedlichen Anwendungsszenarien ausgeführt und getestet. Der erste Anwendungsfall ist der Wald, wo Bäume als Landmarken verwendet werden. Zur Detektion der Bäume wurden die Detektionsalgorithmen für Laserscanner und Stereokamera aus Kapitel 4.2.2 verwendet. Der zweite Anwendungsfall ist eine extraterrestrische Planetenexplorationsmission, wo sich ein mobiler Roboter anhand von Felsen in der Umgebung lokalisiert. Hier wurden die Detektionsalgorithmen aus Kapitel 4.2.3 verwendet.

#### Lokalisierung im Wald

Im Anwendungsszenario „Wald“ wurden zur Erfassung von Referenzdaten zunächst die einzelnen Bäume (Position und Durchmesser) eines Testgebiets, in der Waldfläche von Lattenberg in Nordrhein-Westfalen (NRW), von einem Vermessungsingenieur und einem Forsteinrichter eingemessen und kartiert. Während einer Durchfahrt des kartierten Testgebiets mit der SELMA Einheit auf der mobilen Roboterplattform „SeekurJr.“ wurden die Roboterposen an verschiedenen Zeitpunkten von außen durch den Vermessungsingenieur ermittelt. Dazu wurden auf dem Roboter vorne und hinten Markerplatten befestigt, die durch den Vermesser von einem Einmesspunkt aus mit Hilfe eines Theodolits anvisiert und verortet wurden (Abbildung 5.11). Durch das Einmessen beider Markerplatten steht nicht nur die Position des Roboters sondern auch dessen Orientierung als Referenzmessung zur Verfügung.

Abbildung 5.12 zeigt die Trajektorie einer Testfahrt mit den vermessenen Positionen der Markerplatten. Die Messungen sind als rote bzw. gelbe Punkte in der Abbildung zu sehen. Das Ergebnis der Lokalisierung



ABBILDUNG 5.11: Durch Vermessungsingenieure werden die Lokalisierungsergebnisse validiert. Der Roboter ist dazu mit zwei Markerplatten (rot und gelb) ausgestattet, die mit einem Theodolit angepeilt und eingemessen werden. Dadurch wird ebenfalls die Orientierung des Roboters berücksichtigt.

über die gesamte Testfahrt ist als blaue Linie dargestellt und die Roboterposen zu den Zeitpunkten der Referenzmessungen sind jeweils durch die entsprechende Darstellung des Robotermodells visualisiert. Während der Testläufe wurde der Roboter regelmäßig in Abständen von einigen Metern angehalten, sodass die Vermessungsingenieure die jeweiligen Positionen der Markerplatten einmessen konnten.

Eine Auswertung der Lokalisierungsgenauigkeit für das Szenario „Wald“ ist in der Abbildung 5.13 abzulesen und wie folgt zu interpretieren. Die Lokalisierung wurde während der Messung nicht unterbrochen, sondern lieferte weiterhin die regelmäßig getakteten Ergebnisse. Dadurch ist in den Auswertungen zu den einzelnen Messzeitpunkten (in den Diagrammen jeweils voneinander getrennt durch eine vertikale Linie) ein Rauschen zu erkennen, das in erster Linie von der probabilistischen Methodik der Monte-Carlo-Lokalisierung (insbesondere das zufällige Auswerfen von Hypothesen) herrührt, aber auch durch Sensorrauschen und die davon abhängige Landmarkendetektion beeinflusst wird.

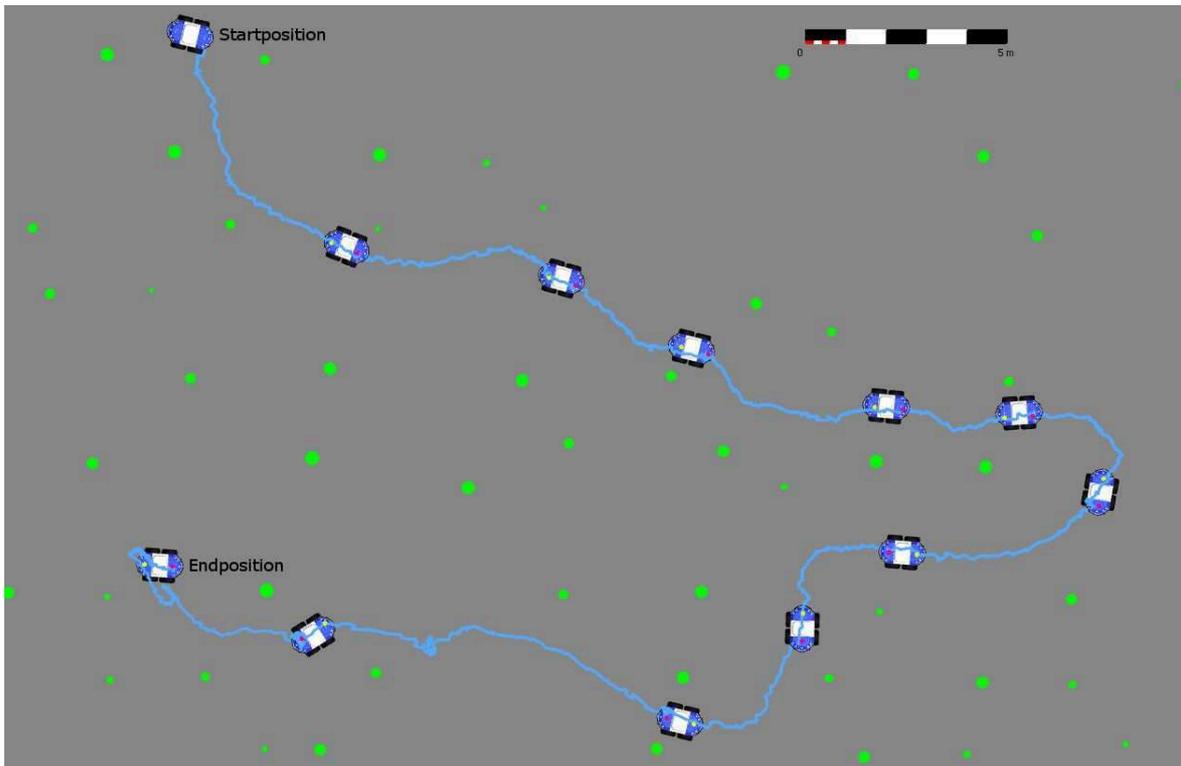


ABBILDUNG 5.12: Trajektorie der mobilen Roboterplattform im Wald. Die von den Vermessern extern gemessenen Referenzmarker auf dem Roboter (vorne rot, hinten gelb) sind als entsprechend farbige Punkte eingezeichnet, ebenso wie die aus der Lokalisierung stammenden Roboterpose.

Allgemein lässt sich zur Lokalisierung sagen, dass der absolute, translatorische Fehler im Mittel deutlich unterhalb von 10 Zentimetern und die Abweichung in der Orientierung (Betrachtet wurde nur die Rotation um die Hochachse) unter einem Grad liegen. Damit stellt die Lokalisierung selbst eine außerordentlich geeignete Basis für höhere Aufgaben, wie das später betrachtete SLAM-Verfahren, dar.

Jetzt muss die Lokalisierung erneut ohne im Vorfeld bekannte Karte evaluiert werden, um die Verknüpfung beider Verfahren zu evaluieren und den Grundstein für die Evaluation der Karte im nächsten Abschnitt zu legen. Das Ergebnis hierzu befindet sich in Abbildung 5.14. In dem Fehler erkennt man, dass zunächst mit steigender Entfernung zum Startpunkt die Lokalisierung aufgrund akkumulierender Fehler schlechter wird. Jedoch sieht man auch, dass ab dem Punkt, wo der Roboter wieder

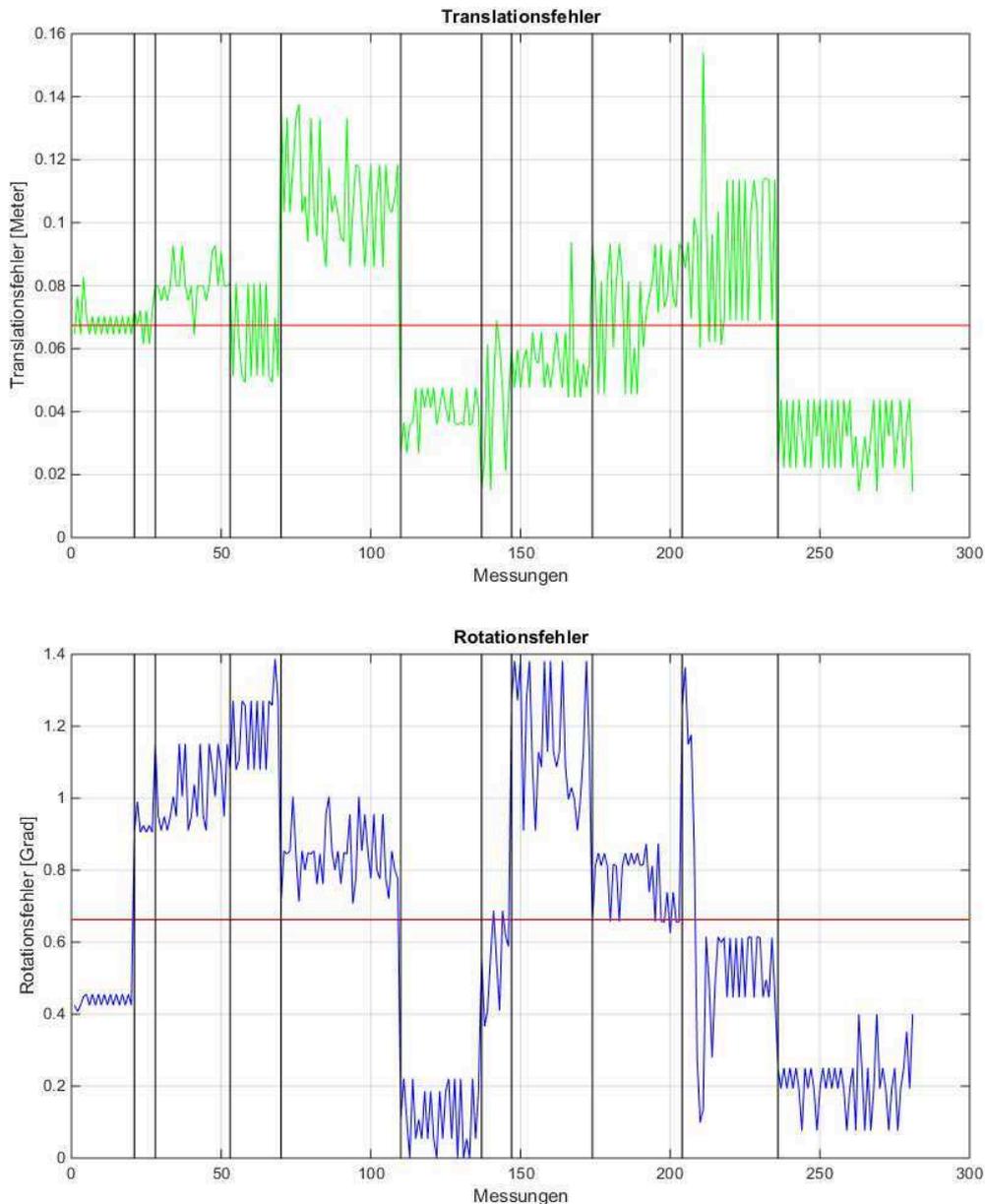


ABBILDUNG 5.13: Lokalisierungsgenauigkeit - Translationsfehler/Rotationsfehler

in den Bereich einfährt, der bereits vorher „gesehen“ wurde und sich entsprechend schon Landmarken in der Karte befinden, der Fehler wieder kleiner wird. Das zeigt zum Einen, dass das sogenannte „Loop-Closing“ mit der Landmarkenkarte funktioniert und zum Anderen gibt es eine Strategie zur Exploration unbekannter Gebiete vor, nämlich mit einem fortwährenden Wiederkehren in bereits besuchte Gebiete, um den sich

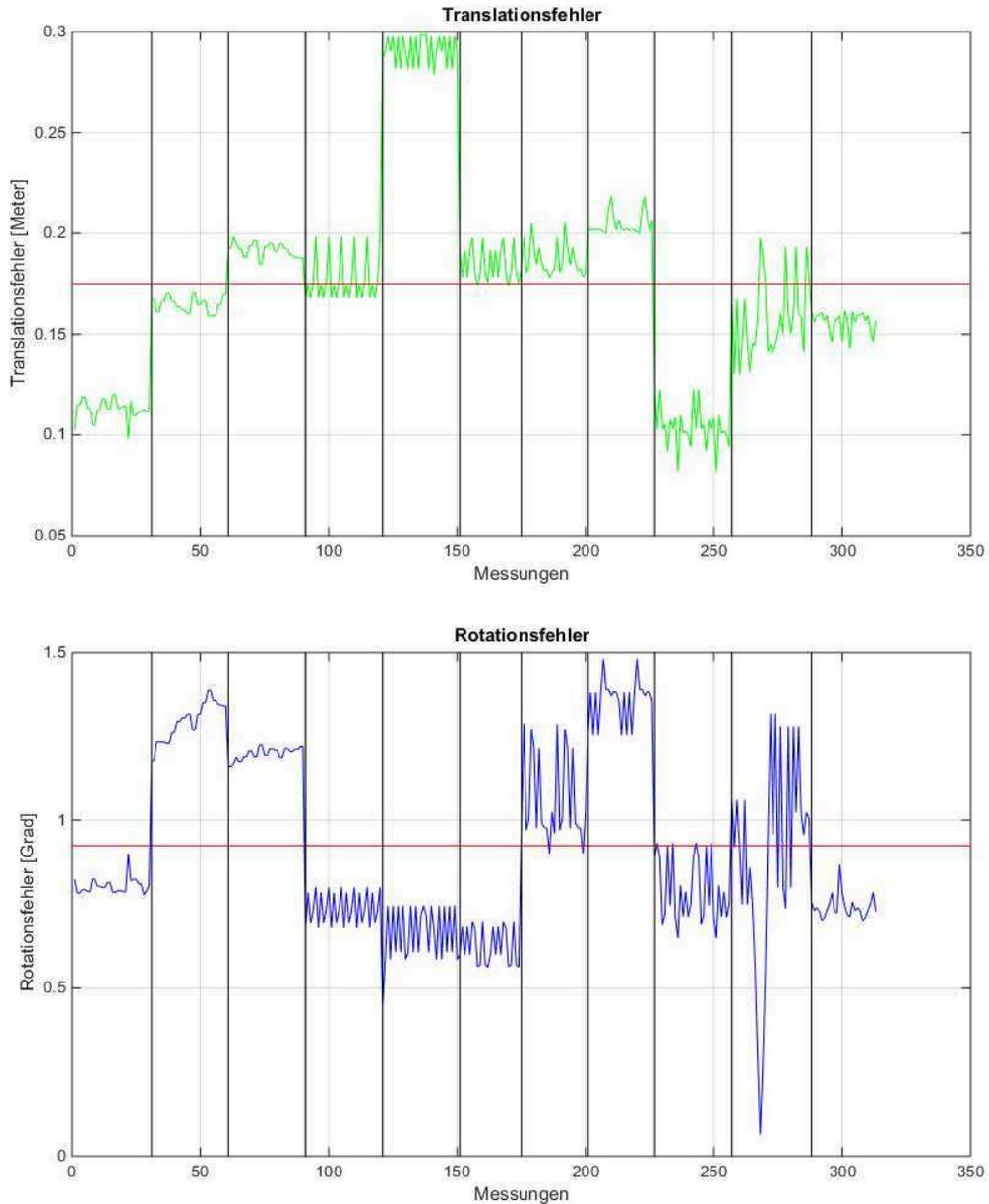


ABBILDUNG 5.14: Relative Lokalisierungsgenauigkeit - Translationsfehler/Rotationsfehler

akkumulierenden Fehler erneut zu verringern.

Doch zunächst soll noch ein weiteres Anwendungsgebiet des hier verwendeten Lokalisierungsverfahren betrachtet werden, das im Rahmen von Forschungsprojekten näher untersucht wurde.

### Lokalisierung in planetaren Explorationsmissionen

Das nächste Anwendungsszenario ist eine extraterrestrische Planetenexplorationsmission. Die Hauptidee dieses Szenarios ist das Gebiet der Mission während der Landephase großflächig von oben zu erfassen und anschließend ein Umgebungsmodell aufzubauen, das sowohl als Datenbasis zur Missionsplanung als auch als Karte zur Landmarken-basierten Lokalisierung dient. Das Szenario wurde zunächst in der Simulation durchgeführt um insbesondere die Steindetektion zu verifizieren (gezeigt in Kapitel 4.2.3). Zur Verifikation des Szenarios anhand eines realen Testbeds wurde die Lokalisierung mit der mobilen Roboterplattform Seekur Jr in einem Steinbruch getestet. Dazu wurden zunächst Steine im Testgebiet verteilt und anschließend mit einer Drohne und einem 3D-Laserscanner eingemessen. Eine genaue Beschreibung des Anwendungsszenarios findet sich in Kapitel 7.2.1.

Bei der Lokalisierung im Steinbruch wurde zur Verifikation die Roboterposition zu Beginn und zum Ende verschiedener Bewegungsabfolgen eingemessen. Ein Vermessungsingenieur stand hier nicht zur Verfügung, daher wurde die Position des Roboters jeweils durch zwei einfache Entfernungsmessungen mit Hilfe eines Laserentfernungsmessgeräts von zwei zuvor festgelegten Fixpunkten  $F_1$  und  $F_2$  bestimmt, wie in Abbildung 5.15 dargestellt. Durch diese Messungen sind zwar jeweils beide Roboterpositionen  $R_1$  und  $R_2$  möglich, jedoch kann durch die Wahl zweier Randpunkte des Testareals als Fixpunkte jene Roboterposition ausgeschlossen werden, die außerhalb des Areals liegt. Aufgrund des geringeren rotatorischen Fehlers im Vergleich zu den translatorischen Abweichungen im Szenario „Wald“, wurde auf das Einmessen der Roboterorientierung verzichtet, denn die erwarteten Messfehler mit einem unnivellierten, von Hand geführten, optischen Messgerät liegen im Bereich der zu erwarteten Fehler in der Orientierung.

Für die Erstellung der Referenzkarte wurde das Testgebiet mit einem stationären 3D-Laserscanner eingemessen und zusätzlich mit einer Drohne befliegen, die mit einem GPS-Receiver und einer Farbkamera bestückt ist. Aus den Kameradaten wurde ein orthorektifiziertes Bild erzeugt, das als Grundlage für die Landmarkenkarte diente. Anschließend wurden die einzelnen Felsen manuell in der Karte segmentiert und entsprechend der 3D-Punktwolke des Laserscanners attribuiert. Für die Position der Landmarke in der Karte wurde jeweils der Mittelpunkt der

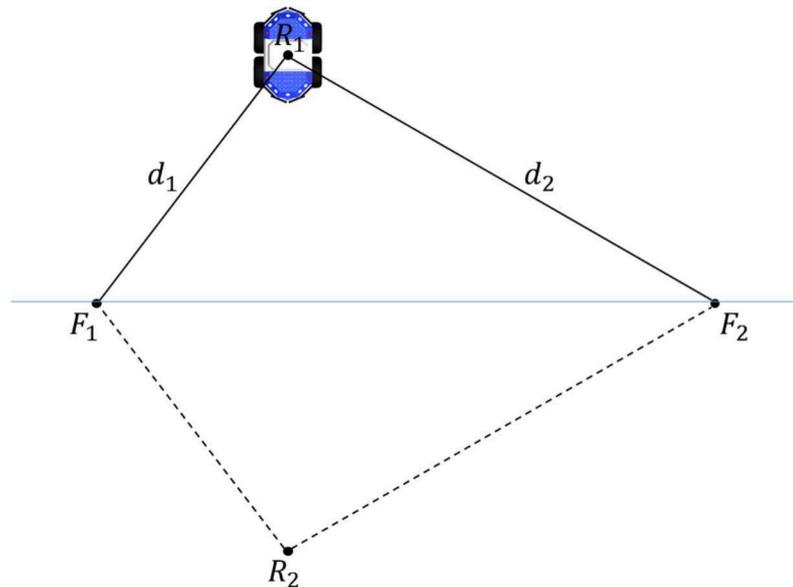


ABBILDUNG 5.15: Positionsbestimmung der Roboterplattform anhand von Distanzmessungen  $d_1$  und  $d_2$  aus den Fixpunkten  $F_1$  und  $F_2$ .

segmentierten Landmarkenregion gewählt.

Es wurde eine Reihe von zehn Testsequenzen aufgenommen. Vier davon sind in Abbildung 5.16 zusammengestellt. Die eingemessenen Start und Endpunkte der Testsequenzen sind dort jeweils mit einem grünen bzw. roten Punkt markiert.

Die Abweichungen zwischen der eingemessenen Endposition und dem entsprechenden Ergebnis des Lokalisierungsalgorithmus' lag im Durchschnitt aller Testfahrten unter 20 Zentimetern und damit im Vergleich zu den verwendeten Steindurchmessern (zwischen 30 und 140 Zentimetern) deutlich im Rahmen der Detektionsgenauigkeit der Steine. Somit bestätigt sich die Zuverlässigkeit des verwendeten Lokalisierungsansatzes auch unter schwierigen Bedingungen und unterstreicht die Robustheit des Verfahrens für eine Verwendung mit verschiedensten Landmarkentypen.

Wird die Landmarkenkarte unabhängig von der Lokalisierung im Vorfeld erzeugt - beispielsweise mit der Hilfe von Luftbildern, Lasermessungen oder aus Planungsdaten - so spiegelt die Genauigkeit der jeweiligen Landmarkendetektion zusammen mit der Zuverlässigkeit der Karte die Größenordnung der Lokalisierungsqualität wider. Wie in diesem Abschnitt gezeigt wurde, sind Genauigkeiten in der Translation

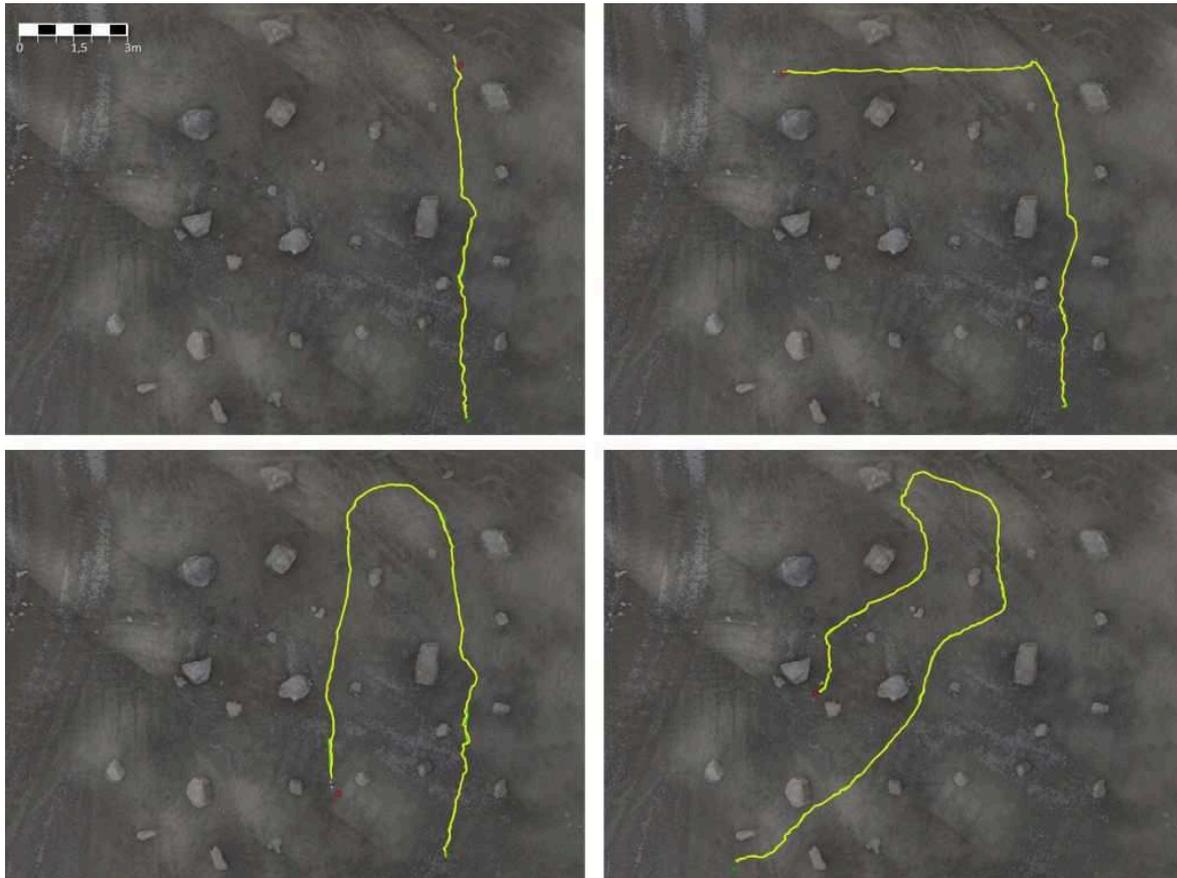


ABBILDUNG 5.16: Ergebnisse der Lokalisierung im Steinbruch anhand von Felsen als Landmarken. Die jeweilige Startposition wurde mit einem grünen Punkt markiert, die Endposition mit einem roten Punkt.

unter 10cm möglich und im Bereich von 20 cm auch unter schwierigen Bedingungen zu erreichen. Damit eignet sich das Verfahren ebenfalls zur simultanen Lokalisierung und Kartierung, was nun im Anschluss genauer untersucht werden soll.



## Kapitel 6

# Anwendungsübergreifende Umweltmodellierung

Die Anwendungsübergreifende Umweltmodellierung verknüpft nun die semantischen Informationen der Umgebungserfassung mit den Ergebnissen der Lokalisierung. Auf diese Weise werden die relativ zur Sensorik aufgenommene Umgebungsinformationen gegenüber einem globalen Referenzsystem verortet und in ein Umweltmodell integriert. Neben einer allgemeinen Behandlung der Umweltmodellierung werden in diesem Kapitel Ergebnisse aus Anwendungen im Wald und bei der Indoor-Lokalisierung vorgestellt.

### 6.1 Simultane Lokalisierung und Kartenerzeugung (SLAM)

Mit dem Landmarkentracking und der Monte Carlo Lokalisierung wurden bisher zwei unterschiedliche Ansätze der Lokalisierung behandelt. Auf der einen Seite steht ein Dead-Reckoning-Ansatz, der ausgehend von einer initial gegebenen Pose und ohne die Verwendung einer Karte die Eigenbewegungen anhand von aufeinanderfolgenden Umgebungsbeobachtungen ermittelt und diese zur jeweilig aktuellen Pose akkumuliert. Auf der anderen Seite steht ein Verfahren, das auf Basis einer im Vorfeld gegebenen Karte eine absolute Posenschätzung vornimmt, die im Laufe der Zeit sukzessive bewertet und zu einer Lösung konvergiert. Gemein haben beide Verfahren die Umgebungsbeschreibung in Form von Landmarken gemäß der Überlegungen aus Kapitel 5. Durch eine

geschickte Verknüpfung beider Verfahren lässt sich aus der reinen Lokalisierung ein Verfahren ableiten, dass die Detektionen während des Landmarkentrackings nicht nur für den Moment vorhält, sondern mit Hilfe der aktuellen Lokalisierungsergebnisse in eine Karte einträgt.

Diese Karte kann parallel bereits für die Lokalisierung mit Partikelfilter genutzt werden. Detektionen, die beim Matching nicht zugeordnet werden können, werden in die Karte hinzugefügt. Durch diese Art der Verknüpfung wird ein simultaner Lokalisierungs- und Kartierungs-Ansatz (SLAM) erzeugt, der gleichzeitig von den Vorteilen beider beinhalteter Verfahren profitiert. Konkret heißt das, dass ausgehend von einer initialen Pose, zunächst eine relative Bewegungsschätzung durchgeführt wird, die parallel alle detektierten Landmarken in einer globalen Karte protokolliert und über die Zeit aktualisiert. Da relative Lokalisierungsverfahren den Nachteil haben, Fehler über die Zeit zu akkumulieren, müssen diese durch spezielle Mechanismen abgesichert werden. Einer dieser Mechanismen baut auf dem sogenannten Loop-Closing-Problem [185] auf, also dem Problem, das auftritt, wenn ein zuvor bereits besuchtes Gebiet erneut aufgesucht wird. Dann muss das Lokalisierungsverfahren bereits in Vergangenheit detektierte Strukturen (hier: Landmarken) wieder erkennen müssen, um nicht auf Basis der aktuell mit akkumulierten Fehlern behafteten Posenschätzung weiter zu rechnen, sondern die zuvor gemachten Beobachtungen als Stützpunkt zur Eliminierung dieser Fehler nutzen zu können. Dafür ist es erforderlich, dass zum einen alle zuvor erfassten Landmarken in einer Karte vorgemerkt werden, und zum anderen sichergestellt wird, dass diese Landmarken auch dann wieder erkannt werden, wenn die aktuelle Pose aufgrund von Fehlern um einen gewissen Grad abweicht. Durch die absolute Lokalisierungskomponente des Partikelfilters wird diese Anforderung erfüllt und ermöglicht eine zuverlässige Fehlerkorrektur durch das erneute Durchfahren bereits besuchter und kartierter Umgebungen.

Ein wichtiger Punkt bei der Kartengenerierung ist das zugrundeliegende Koordinatensystem sowie die darin bestimmte initialen Pose. Die einfachste Herangehensweise an dieses Problem ist die Verwendung eines relativen Koordinatensystems worin die Startpose des Roboters als Ursprung dient. Dann müssen keine weiteren Anpassungen vorgenommen werden, und die Lokalisierung kann sofort starten. Sofern das Verfahren lediglich zur Lokalisierung in einem unbekanntem Gebiet genutzt wird, ist dieser Ansatz tatsächlich ausreichend. Sobald aber ein

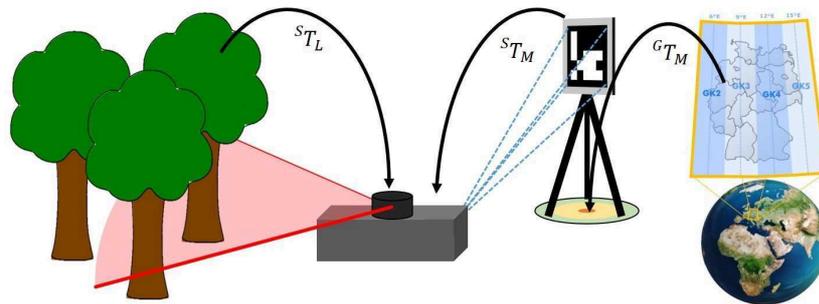


ABBILDUNG 6.1: Kombination aus Umgebungserfassung und Lokalisierungsverfahren

Bezug zwischen der eigenen Pose und der Umwelt aufgebaut werden soll, indem die Karte mit weiteren Datenquellen verschnitten oder kombiniert wird (Beispiele werden in Kapitel 7 eingehend behandelt), so ist das Koordinatensystem anderweitig bereits vorgegeben und man steht vor dem Problem, die initiale Pose in Bezug zum Referenzkoordinatensystem zu bringen. In der Regel sind in diesem Referenzsystem keine natürlichen Landmarken verzeichnet, die man für die Lokalisierung nutzen könnte, also ist eine Initialisierung nicht ganz trivial. Man muss sich also an anderen Punkten orientieren. Eine Möglichkeit hierfür ist die Nutzung von Referenzpunkten (z.B. Landvermessungspunkte, Referenzpunkte aus Katasterdaten, oder Building Information Models (BIM) in Indoor-Szenarien oder einfach mehrere GPS-Messungen) in Kombination mit künstlichen Markern. Hierfür eignen sich hervorragend AR-Marker, da sie einfach erzeugt werden können, bereits eindeutig identifizierbar sind, sowie die Lokalisierung anhand eines einzelnen Markers möglich ist. In Kapitel 7.1.1 wird eine konkrete Anwendung vorgestellt, die die absolute Verortung des während der Lokalisierung erzeugten semantischen Umgebungsmodells, durch die Nutzung von AR-Markern ermöglicht.

Ein Beispiel für die Verknüpfung mehrerer Lokalisierungsverfahren mit einer semantischen Umgebungserfassung zur simultanen Umweltmodellierung ist in Abbildung 6.1 skizziert. Das Beispiel stammt aus der Stichprobenaufnahme im Anwendungsgebiet der forstlichen Stichprobeninventur<sup>1</sup>. Zur absoluten Verortung wird hier ein geodätischer Einmesspunkt verwendet, über dem ein künstlicher Marker positioniert und eingenordet wird. Der Position des Einmesspunkts wird im jeweiligen Geokoordinatensystem  $G$  angegeben  $M T^G$ . Der Marker  $M$  dient zur

<sup>1</sup><https://www.bundeswaldinventur.de/>

initialen Verortung der Sensoreinheit  $S$  zum Einmesspunkt. Durch die im letzten Kapitel behandelte markerbasierte Lokalisierung kann mit Hilfe einer einfachen monooptischen Kamera die Lage der Sensoreinheit bezüglich des Markers, und damit zum Einmesspunkt, ermittelt werden, indem die Transformation  ${}_M T^S$  bestimmt wird. Gleichzeitig wird die Lage der Landmarken in der Umgebung  $L$  mit den Detektionsalgorithmen aus Kapitel 4.2.2 relativ zur Sensoreinheit erfasst und durch die Transformation  ${}_L T^S$  beschrieben. Durch die Verknüpfung aller beteiligter Transformationen können die Landmarken in absoluten Geokoordinaten  ${}_L T^G$  angegeben werden:

$${}_L T^G = {}_L T^S \cdot ({}_M T^S)^{-1} \cdot {}_M T^G. \quad (6.1)$$

Die Sensoreinheit dient hier als Bindeglied zwischen dem geodätischen Einmesspunkt und den erkannten Objekten der semantischen Umgebungserfassung.

Neben dem Hinzufügen neuer Landmarken ist auch die Verwaltung und Aktualisierung bereits vorhandener Landmarken von hoher Bedeutung. Gerade in Bezug auf die Landmarkenattribuierung ist die Möglichkeit der nachträglichen Erweiterung der Karte erforderlich. Wird ein dreidimensionales Objekt aus verschiedenen Perspektiven beobachtet, so lässt sich dieses meist besser und vor allem genauer beschreiben, als wenn es nur von einer Seite beobachtet wurde. Aus diesem Grund wird hier für jede Landmarke vorgehalten, wie oft sie gesichtet wurde. Somit ist es möglich bei Sichtungen aus verschiedenen Perspektiven die Position der Landmarke sowie deren Merkmale über die Zeit zu mitteln. Auf diese Weise werden alle Merkmale einer Landmarke mit der Anzahl ihrer Sichtungen gewichtet, d.h. wird die Landmarke zu einem späteren Zeitpunkt erneut gesehen, wirkt sich eine Änderung nur anteilmäßig aus, sodass einmalige fehlerhafte Beobachtungen die Landmarken in der Karte nicht entscheidend verändern. Das gilt in gleicher Weise für die Pose einer Landmarke sowie für ihre beschreibenden Attribute.

Darüber hinaus stellt die Anzahl der Sichtungen einer Landmarke gleichzeitig ein gewisses Qualitätsmaß dar. Die Wahrscheinlichkeit, dass eine Landmarke korrekt aus den Sensordaten extrahiert wurde („true positive“) steht im proportionalem Verhältnis zu ihrer Detektionshäufigkeit. Im Umkehrschluss lassen sich Fehldetektionen („false positives“) anhand deutlich geringerer Detektionshäufigkeiten erkennen und durch

das Festlegen einer unteren Grenze einfach ausfiltern. Die Bestimmung dieser Grenze wurde in Kapitel 4.3.1 beschrieben (siehe auch Abbildung 4.21).

## 6.2 Kartenerzeugung und Aktualisierung

Bei der reinen Lokalisierung werden die Sensordaten sowie die darin extrahierten Merkmale oftmals direkt nach der Berechnung verworfen, um den Aufwand der Datenhaltung und -organisation zu beschränken. Wenn das Ziel des jeweiligen Ansatzes ausschließlich aus der Positionsbestimmung bestehen soll, ist dieses Vorgehen auch durchaus gerechtfertigt. Allerdings wäre es auch möglich die Informationen zu sammeln und vorzuhalten, um sie bei einem erneuten Besuch derselben Region zu nutzen, um die aktuell gesammelten Daten zu ergänzen. Die Ergänzung ist dabei auf verschiedenen Ebenen möglich. Zunächst ist es möglich, eine bereits vorhandene Karte parallel zur Lokalisierung zu ergänzen oder zu korrigieren, wenn festgestellt wird, dass darin verzeichnete Landmarken fehlerhaft bzw. nicht vollständig attribuiert sind, oder sie nicht mehr existieren und daher aus der Karte gelöscht werden können. Darüber hinaus können detektierte Landmarken, die aber aktuell noch in der Karte fehlen, nachträglich ergänzt werden.

Unter der Betrachtung des Anwendungsfalls der Lokalisierung, kann ein detailliertes Umweltmodell zur Verbesserung der Zuordnung von Landmarken beitragen. Insbesondere durch die Attribuierung können markante Landmarken oder Bereiche in der Karte bestimmt, und diese als Ankerpunkte markiert werden. In Abbildung 6.2 ist die Durchmesser-Verteilung eines Baumbestandes farbkodiert dargestellt. Man sieht hier deutlich einige mögliche Ankerpunkte, wo sich vereinzelt Bäume mit deutlich höherem Durchmesser als die umgebenden Bäume befinden. Diese Ankerpunkte können beim SLAM-Verfahren beispielsweise dazu genutzt werden, gezielt das mehrfache Besuchen dieser Bereiche zu planen, um Drift-Fehler bei der Kartenerzeugung zu minimieren. Speichert man bei jeder Beobachtung zusätzlich die Unsicherheiten der Lokalisierung ab, so lassen sich bei der wiederholten Detektion einer Landmarke nicht nur der Fehler zwischen der letzten und jetzigen Sichtung korrigieren, sondern die gesamte Kette an Detektionen, die zwischen diesen beiden Sichtungen stattgefunden haben.

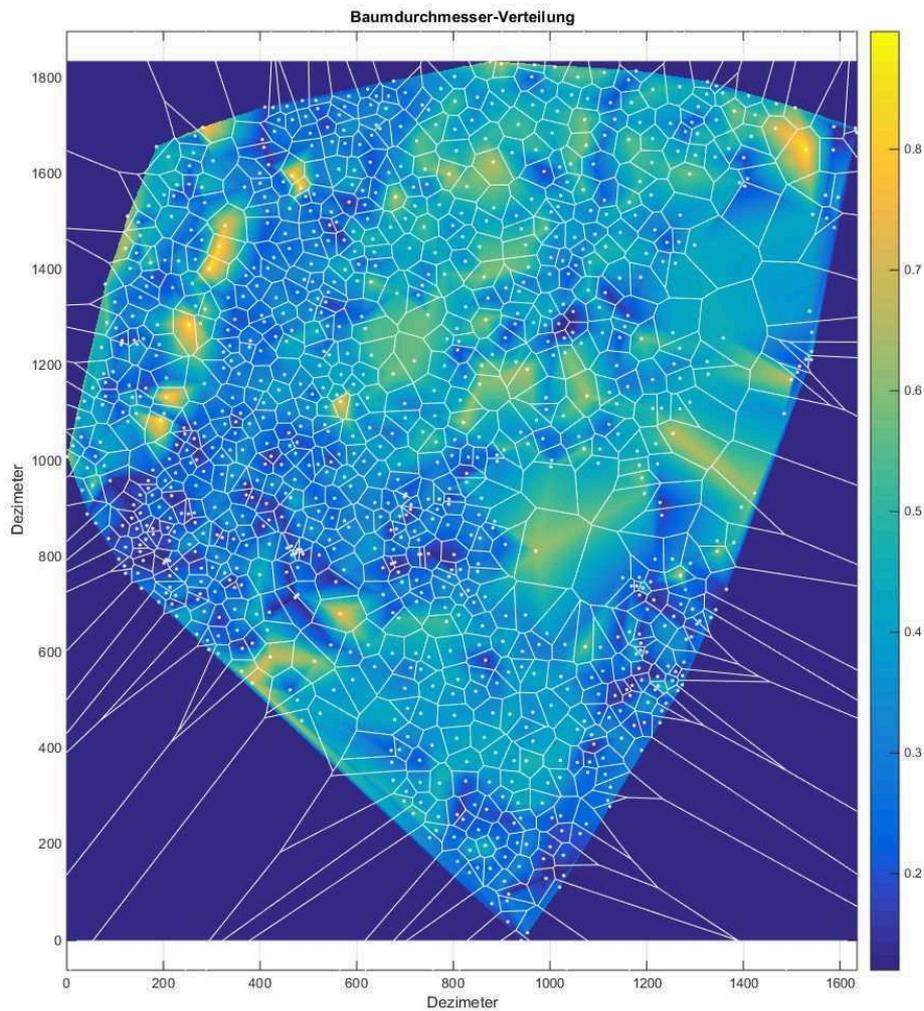


ABBILDUNG 6.2: Farbkodierte Durchmesser-Verteilung eines Baumbestandes mit überlagertem Voronoi-Diagramm zur Darstellung der Baumabstände.

### 6.3 Ergebnisse der Kartenerstellung

Die Kombination, der lokal durch die Umgebungserfassung detektierten und erkannten Objekte mit der anschließenden relativen Lokalisierung, erlaubt eine simultane Erzeugung eines Umweltmodells aus semantischen Objekten. Diese resultierende Karte aller erfassten Objekte kann

nachfolgend wieder zur Lokalisierung (diesmal global bezüglich dieser Karte) verwendet werden, oder für weiterführende Aufgaben, wie sie bereits im Kapitel 7 „Anwendungen“ näher erörtert wurden. Die Qualität der nachgelagerten und auf dem Umweltmodell basierenden Anwendung, sei es die Lokalisierung oder eine Umweltanalyse, hängt nun unmittelbar von der Qualität des Umweltmodells ab. In diesem Abschnitt soll daher diese Qualität in Bezug auf Genauigkeit der jeweiligen Umweltmodelle in den einzelnen Anwendungsszenarien evaluiert werden.

### 6.3.1 Wald

Das Anwendungsgebiet „Wald“ ist nicht nur für die Lokalisierung interessant sondern auch aus forstwirtschaftlicher Sicht. Die Erstellung von einzelbaumbasierten Bestandesaufnahmen ermöglicht gegenüber einer sonst üblichen bestandesbasierten Aufnahme, die automatisierte Ableitung forstwirtschaftlich relevanter Attribute, wie etwa der Holzvorrat in der Fläche. Dafür muss das Umweltmodell bestimmte Qualitätskriterien erfüllen. Daher sollen in diesem Abschnitt die Ergebnisse aus Aufnahmen in verschiedenen Testgebieten ausgewertet und analysiert werden. Abbildung 6.3 zeigt die jeweiligen Gegebenheiten und Ausprägungen der drei, stark unterschiedlichen Testgebiete Lattenberg (NRW), Traunstein (Bayern) und Zollikofen (Schweiz). Insbesondere die Testgebiete in Lattenberg und in Zollikofen konnten genau untersucht werden, da hier sowohl die Baumpositionen als auch die Baumdurchmesser zum Zeitpunkt der Tests bekannt und die Gegebenheiten für die sensorielle Erfassung hervorragend geeignet waren.

#### Lattenberg (NRW)

Zunächst sollen erneut die Testaufnahmen aus dem Waldstück in Lattenberg (NRW) analysiert werden, die bereits für die Evaluation der Lokalisierung herangezogen wurden. Da die Baumpositionen und Durchmesser hier bereits, für die Erzeugung einer Navigationskarte zur Bewertung der kartenbasierten Lokalisierung, aufgenommen wurden, eignet sich dieses Testgebiet hervorragend für eine Auswertung der automatisch generierten Karte von Baumpositionen. Es handelte sich bei dem Gebiet um einen - für diesen Zweck - sehr einfachen Fichtenreinbestand mit



ABBILDUNG 6.3: Testgebiete mit unterschiedlichem Bewuchs für die Anwendung im Wald. Linke Spalte: Fichtenreinbestand mit wenig Unterstand und geringer Beastung auf Messhöhe, Mitte: Großflächiges Testgebiet mit verschiedenen Baumarten, zumeist viel Unterstand und Jungwuchs und starker Verästung, Rechte Spalte: Marteloskopfläche der HAFL in Bern mit sehr alten Beständen in der Mitte und Jungwuchs mit z.T. unzugänglichem Unterstand am Rand.

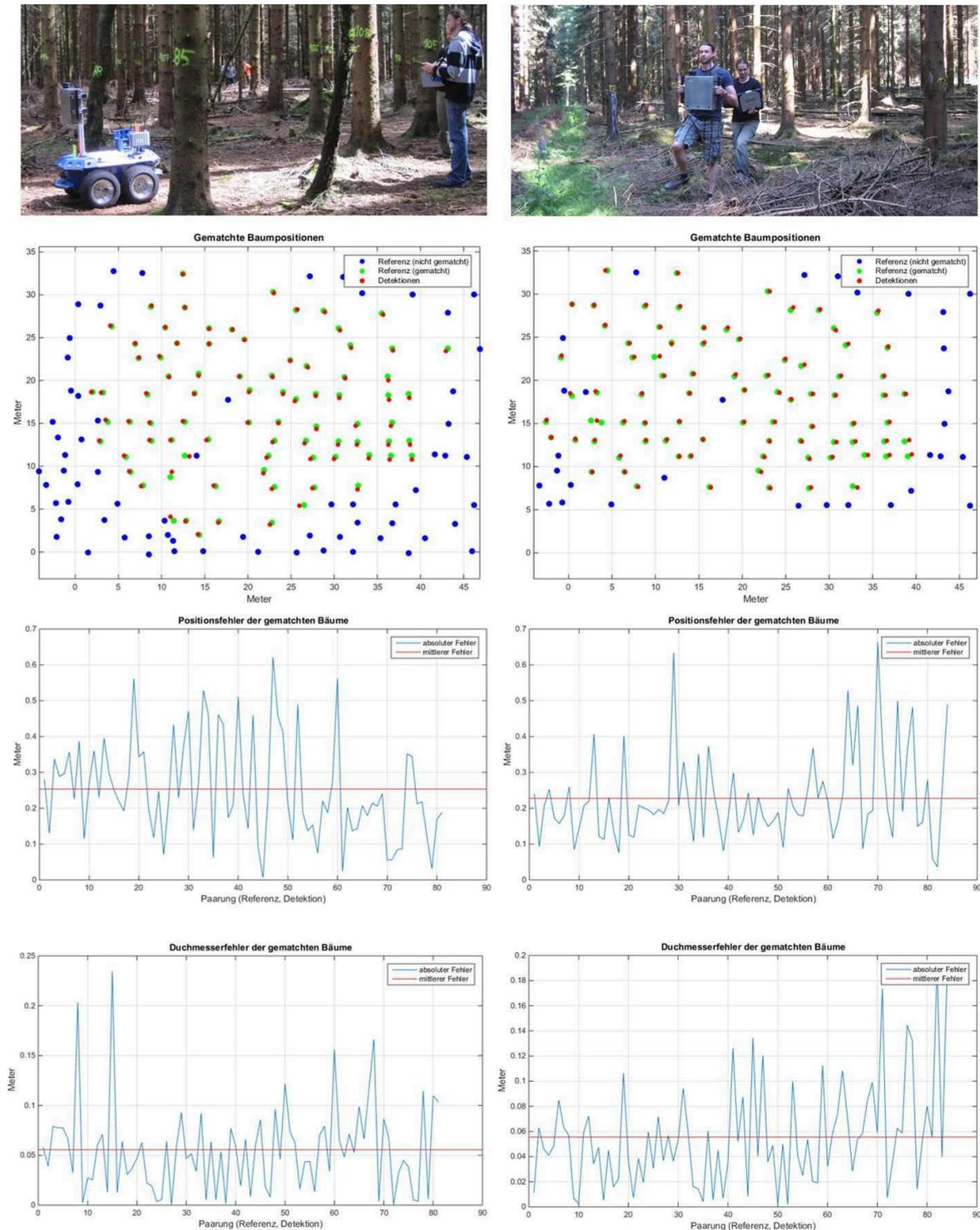


ABBILDUNG 6.4: Evaluation des SLAM-Ergebnisses in Lattenberg (NRW). Linke Spalte: Testlauf mit SELMA-Einheit auf mobilem Roboter, rechte Spalte: SELMA-Einheit getragen. Oben: Ergebnis der automatisierten Kartenerzeugung. Detektierte Bäume (rot), manuell erfassten Referenzdaten (grün), Mitte: Positionsfehler detektierteter Bäume, unten: Fehler des bestimmten Baumdurchmessers.

Baumdurchmessern zwischen 0,1 und 0,6 Metern. Diese Baumdurchmesser, der Umstand, dass die Fichten über kaum Beastung im unteren Bereich des Stamms verfügen und den nahezu nicht vorhandenen Unterstand im Testgebiet, machten die Aufnahmen besonders einfach und störungsunanfällig. Das Ergebnis der Kartenerzeugung für den bereits zur Evaluation der Lokalisierung genutzten Testlauf mit der mobilen Roboterplattform SeekurJr ist in Abbildung 6.4 in der linken Spalte dargestellt. Ein weiterer Testlauf ohne Roboter, in dem die Sensoreinheit händisch vor der Brust getragen wurde ist in der rechten Spalte von Abbildung 6.4 ausgewertet.

Die Karte oben zeigt neben den detektierten Bäumen (rot) auch die manuell eingemessenen Bäume, wobei bei letzteren noch einmal unterschieden wird, ob ein Baum mit einer Detektion gematcht (grün) werden konnte, oder ob dieser Baum nicht sensoruell erfasst (blau) wurde. Die nicht erfassten Bäume im Randbereich des Gebiets wurden aufgrund ihrer Entfernung zur Sensoreinheit während des Testlaufs nicht erfasst. Der blaue Punkt in beiden Karten an der lokalen Position (17,7;17,7) wurde hingegen detektiert, jedoch jeweils mit einer Abweichung von einem Meter in y-Richtung und konnte deshalb nicht korrekt gematcht werden. Hier liegt ein Messfehler oder ein falscher Übertrag von Messwerten bei der manuellen Baumerfassung nahe. Eine Auswertung des Positionierungsfehlers der automatisch erfassten gegenüber der manuell eingemessenen Bäume ist in Abbildung 6.4 in den beiden Diagrammen in der mittleren Zeile dargestellt. Hier ist zu erkennen, dass der Fehler der Baumpositionen im Mittel deutlich unter 0,3 Metern liegt und einen maximalen Fehler von etwa 0,6 Metern aufweist. Die Evaluierung des automatisch erfassten Baumdurchmessers ist im Diagramm unten von Abbildung 6.4 zu sehen. Obwohl hier mitunter große Abweichungen zum tatsächlichen Durchmesser auftreten können, liegt die mittlere Abweichung bei ungefähr 5 Zentimetern. Dieser Fehler ist in erster Linie auf die Winkelauflösung des Laserscanners zurückzuführen (siehe Abschnitt 4.3.1) und auf einen im Detektionsalgorithmus angenommenen, ideal kreisförmigen Querschnitt des Baumstamms.

### **Traunstein (Bayern)**

Der Testbestand in Traunstein ist im direkten Vergleich zur Fläche in Lattenberg besonders anspruchsvoll. Aufgrund der hohen Diversität an

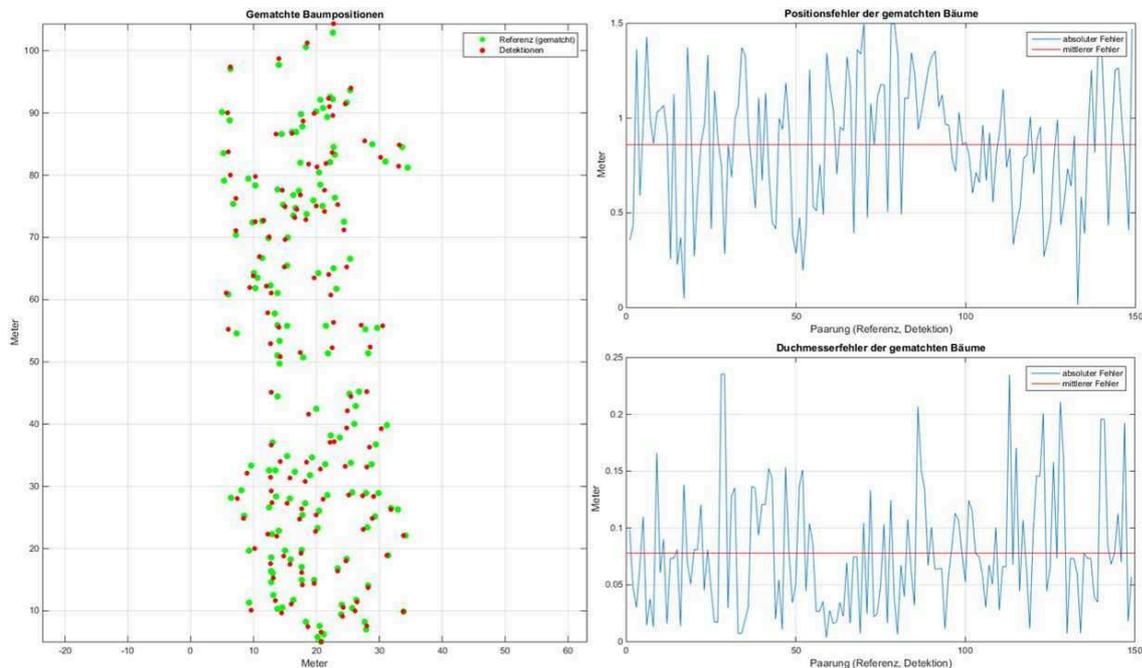


ABBILDUNG 6.5: Evaluation des SLAM-Ergebnisses in Traunstein (Bayern). Links: Ergebnis der automatisierten Kartenerzeugung. Die detektierten Bäume (rot) wurden mit den manuell erfassten Referenzdaten gematcht (grün), rechts-oben: Positionsfehler der detektierten Bäume, rechts-unten: Fehler des Baumdurchmessers in der automatisiert erzeugten Karte.

Art und Alter der vorliegenden Bäume sowie an Schicht, Dichte und Durchforstung des Bestandes selbst, herrschen auf der gesamten Fläche unterschiedliche Bedingungen vor. Auf nur wenigen Teilflächen war es aufgrund vieler Störquellen (Unterholz, Beastungen im unteren Bereich, Jungwuchs) möglich mit der SELMA-Einheit zuverlässig Bäume zu erfassen und zu tracken. Diesen Umstand gibt die Auswertung in Abbildung 6.5 wieder. Sowohl die Genauigkeit der Baumpositionen, als auch die der Durchmesserbestimmung liegt weit hinter dem Genauigkeitsniveau aus Lattenberg.

Somit zeigt der Testbestand in Traunstein die Grenzen der technischen Möglichkeiten für eine terrestrische Erfassung von Waldbeständen auf. Ist der Unterstand zu hoch wird die Sicht im Aufnahmebereich der Sensoren so stark eingeschränkt, dass eine Erfassung und gleichzeitige Lokalisierung nicht mehr zuverlässig möglich ist. In Beständen dieser Ausprägung kann die SELMA-Einheit lediglich ergänzend zur manuellen

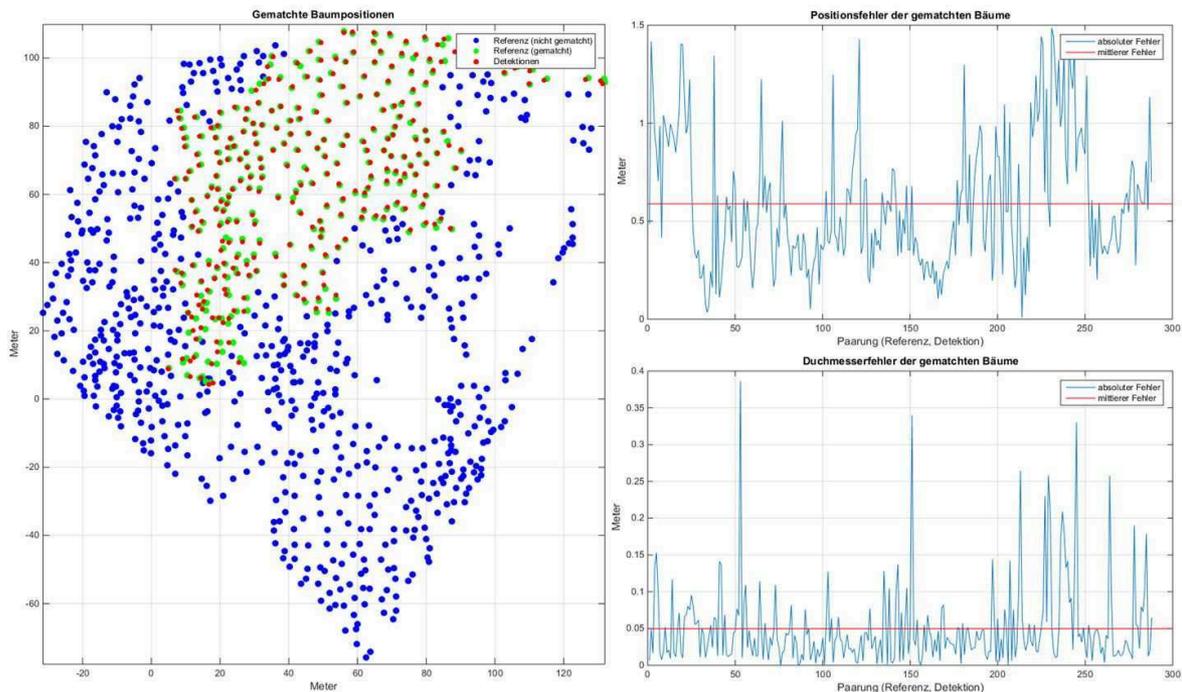


ABBILDUNG 6.6: Evaluation des SLAM-Ergebnisses in Zollikofen (Schweiz). Links: Ergebnis der automatisierten Kartenerzeugung. Die detektierten Bäume (rot) wurden mit den manuell erfassten Referenzdaten gematcht (grün), rechts-oben: Positionsfehler der detektierten Bäume, rechts-unten: Fehler des Baumdurchmessers in der automatisch erzeugten Karte.

Aufnahme eingesetzt werden, d.h. es können zunächst einfache Bereiche automatisch erfasst werden und anschließend durch manuelle Messungen fehlende Bäume ergänzt, und Fehldetektionen verworfen werden. Dabei kann die manuelle Messung von der SELMA-Sensorik profitieren, indem die Sensordaten für die Ermittlung des einzumessenden Baumes genutzt werden. Dies erlaubt, wie bei der automatisierten Kartenerstellung und der Lokalisierung, eine parallele Erfassung mehrerer Bäume von einer Position aus. Im Gegensatz zur herkömmlichen Methode, wo jede Baumposition separat mit einem 1D-Laserentfernungsmessgerät und einem Reflektor eingemessen werden muss, wird mit der SELMA-Sensorik die Anzahl der Einzelmessungen und damit ebenfalls die Anzahl der möglichen Fehlerquellen reduziert.

### Zollikofen (Schweiz)

Die Marteloskopfläche in Zollikofen, direkt am Campus der HAFL-Bern gelegen, wurde ein weiterer Test mit der SELMA-Einheit durchgeführt. Wie in Lattenberg und in Traunstein, wurde auch hier automatisiert eine Karte erzeugt, die anschließend, anhand der Daten der manuellen Erfassung dieser Fläche, evaluiert werden konnte. Anders als in Lattenberg, wurden hier, aufgrund der Größe der Testfläche, mehrere Testläufe, ausgehend von einem zuvor festgelegten Startpunkt, durchgeführt und anschließend eine gemeinsame Karte aus allen Messläufen akkumuliert. Die unverarbeitete Karte, d.h. ohne Filterung und Eliminierung von Ausreißern, wurde mehrfach zur Lokalisierung verwendet und sukzessive durch das Besuchen „neuer Bereiche“ erweitert. Damit wurde das Vorgehen eines Produktiveinsatzes getestet, wo jeder Baum in möglichst vielen Messungen getroffen wird und damit sowohl Messfehler als auch „loop-closing“-Fehler bei der Kartenerzeugung minimieren zu können.

Abbildung 6.6 zeigt die Ergebnisse der automatisierten Kartenerzeugung in Zollikofen. Wie in den Testgebieten zuvor werden hier die Ergebniskarte, die Evaluierung der Baumpositionen anhand manuell erfasster Referenzdaten sowie die Güte der Durchmesserbestimmung ausgewertet. Nach eigenen Angaben der HAFL, wo die Fläche als Lehrfläche eingesetzt wird, liegt die Genauigkeit der manuell eingemessenen Baumpositionen im Größenbereich von 0,1-0,2 Metern und die Bestimmung des Durchmessers bei 1-2 Zentimetern. Die Güte der automatisiert erstellten Karte reicht zwar nicht ganz an diese Genauigkeiten heran<sup>2</sup>, jedoch liegt der Fehler dennoch im vergleichbarem Größenbereich. Außerdem spielt die Zeit, die für die Aufnahme aufgewendet werden muss hier eine wichtige Rolle. Einzelbaumaufnahmen werden in der Praxis gerade deshalb nicht regelmäßig durchgeführt, da der Nutzen in einem schlechten Verhältnis zum Aufwand steht: Der Nutzen ist zwar hoch, jedoch ist der Aufwand um ein Vielfaches höher. Bei der automatisierten Aufnahme kann eine Fläche in ein paar Stunden aufgenommen werden, wofür sonst Tage und Wochen nötig wären.

Die Fläche in Zollikofen ist besonders interessant, da sie, wie in Abbildung 6.3 zu sehen, sowohl aus leicht zugänglichen Bereichen ohne

---

<sup>2</sup>Bei der Auswertung in Abbildung 6.6 muss die Ungenauigkeit der Referenzdaten ebenfalls berücksichtigt werden, da diese als Ground-Truth-Daten zur Evaluierung herangezogen wurden.

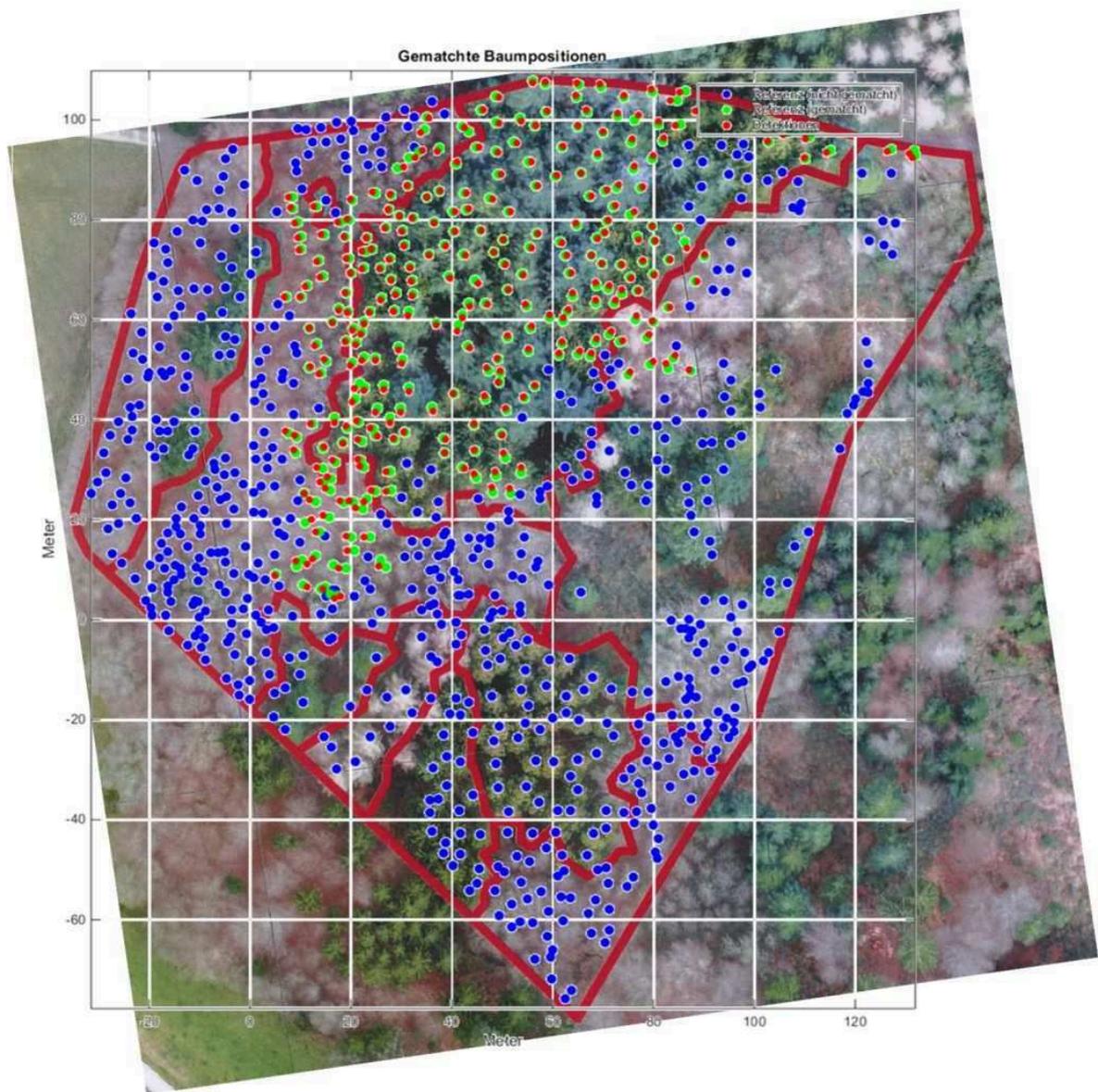


ABBILDUNG 6.7: Ergebnis des SLAM-Verfahrens in Zollikofen mit unterlegtem Orthofoto (Quelle Orthofoto: HAFL-Bern).

Jungwuchs und Unterstand besteht und ebenfalls Bereiche aufweist, die nur mit viel Mühe bzw. überhaupt nicht zugänglich sind. Eine Luftbildaufnahme mit überlagertem Detektionsergebnis ist in Abbildung 6.7 abgebildet. Hier sind deutlich die zuverlässig erfassten Bereiche mit Nadelholz zu sehen, aber auch die dichten und jungen Laubbaumbereiche oder die durch dichte Büsche und Sträucher kaum zugänglichen Bereiche

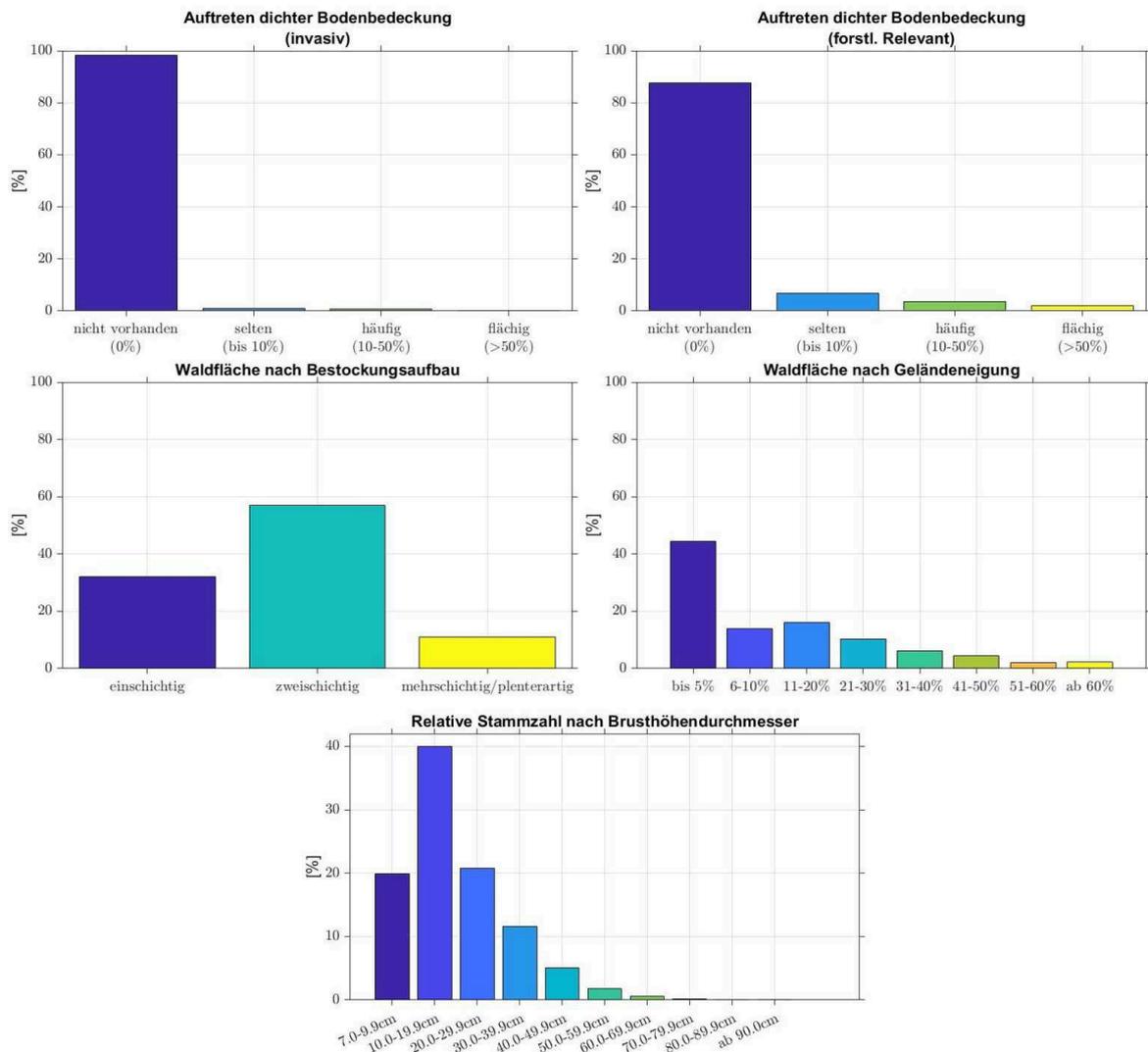


ABBILDUNG 6.8: Umgebungsbedingungen für eine zuverlässige und automatisierte Bestandesaufnahme

im Osten des Bestands.

## Bewertung

Auf Basis dieser Ergebnisse lassen sich einige Umgebungsbedingungen für die zuverlässige automatisierte Bestandesaufnahme festlegen. Diese Kriterien sind im Diagramm aus Abbildung 6.8 aufgeführt und deren jeweiliges, prozentuales Vorkommen in den Wäldern Deutschlands nach der Bundeswaldinventur von 2012<sup>3</sup> angegeben.

<sup>3</sup>Quelle: <https://bwi.info/> - abgerufen am 06.06.2017.

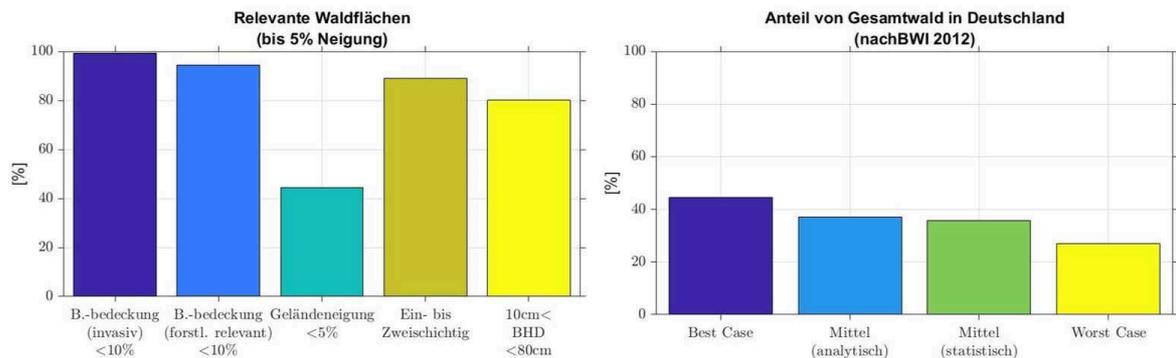


ABBILDUNG 6.9: Statistischer Anteil der Waldflächen in Deutschland für die automatisierte Bestandeserfassung

Die Kombination der angegebenen relativen Auftretswahrscheinlichkeiten ist aufgrund der Unabhängigkeit der jeweiligen Kriterien nicht ganz trivial. Aus diesem Grund werden die Kriterien zunächst für den besten (best case) und schlechtesten (worst case) Fall separat kombiniert und anschließend diese beiden Werte gemittelt. Das Ergebnis dieser Auswertung ist in Abbildung 6.9 dargestellt.

Der beste Fall wäre der, dass alle Kriterien, die eine automatisierte Bestandesaufnahme erlauben grundsätzlich zusammen auftreten und somit die größtmögliche Schnittmenge bilden. In diesem Fall wäre die minimale Auftretswahrscheinlichkeit der Kriterien gleich der maximalen Schnittmenge, sodass im best case 44,46% des Waldes in Deutschland für diese Art der Bestandesaufnahme geeignet sind. Dieser Wert entspricht der Auftretswahrscheinlichkeit für Flächen mit einer Steigung unter 5%.

Im Gegensatz dazu entspricht der worst case dem Inversen der Summe aller Gegenannahmen, d.h. also der Annahme, dass alle Kriterien, die keine automatisierte Aufnahme erlauben alleine auftreten und keine gemeinsame Schnittmenge besitzen. Diese Annahme führt dazu, dass das gemeinsame Auftreten aller positiven Kriterien möglichst gering ist. Nach dieser Beurteilung der Auftretswahrscheinlichkeiten sind lediglich etwa 9% der Waldfläche in Deutschland für diese Art der Bestandesaufnahme geeignet. Im statistischen Mittel aller Auftretswahrscheinlichkeiten wären ca. 30% des deutschen Waldes geeignet und nach dem analytischen Mittel, also dem Mittelwert zwischen best case und worst case, liegt der Wert bei etwa 27% des Waldes.

Da die Geländeneigung den limitierenden Faktor des best-case-Szenarios

darstellt, und eine höhere Geländeneigung bei höher angebrachter Sensoreinheit durchaus denkbar ist, wurde die Untersuchung für Geländeneigungen bis 10% wiederholt (Abbildung 6.10). Dadurch ergibt sich ein neuer Wert für den best case von knapp 60% und der worst case liegt nach dieser Betrachtung bei über 22%. Bei den Mittelwerten können durch diese neue Einschätzung Werte um 40% erreicht werden. Damit ist die automatisierte Bestandesaufnahme zwar in den meisten Gebieten des Waldes in Deutschland nicht möglich, kann aber ergänzend zur manuellen Aufnahme genutzt werden. Die hohe Zeitersparnis bei der Datenerhebung wirkt sich auch bei einer nicht flächendeckenden Anwendung signifikant aus, da die Bäume lediglich in anspruchsvollen Beständen noch manuell erfasst werden müssten.

### 6.3.2 Indoor-Lokalisierung

Die in Abschnitt 5.3 betrachtete markerbasierte Lokalisierung wurde auch im Hinblick einer zur Lokalisierung gleichzeitigen Kartenerzeugung betrachtet. Dazu wurde, während der Durchfahrt einer mit AR-Markern versehenen Tiefgarage und zu Beginn ausgehend von nur einem einzigen, bekannten Marker, die Karte automatisch aufgebaut, indem alle neu erfassten Marker sukzessive in die Karte aufgenommen wurden. Der Anwendungsfall wird in Kapitel 7.4.1 ausführlich vorgestellt. Wie im Fall der Lokalisierung wurde auch hier auf die gleichzeitige Verwendung ergänzender Lokalisierungsverfahren und Filterungen verzichtet. Zwar weicht die rekonstruierte Lokalisierung (grüne Linie in Abbildung 6.11) aufgrund von Fehlern in der Posenschätzung an

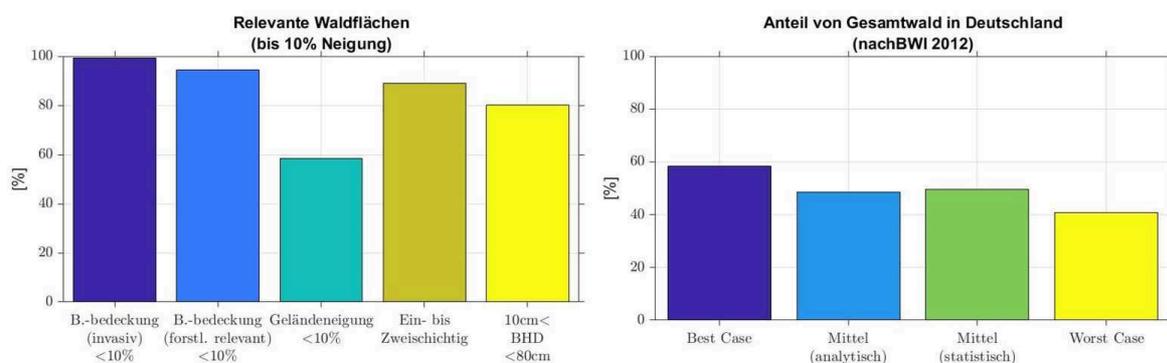


ABBILDUNG 6.10: Statistischer Anteil der Waldflächen in Deutschland für die halbautomatisierte Bestandeserfassung

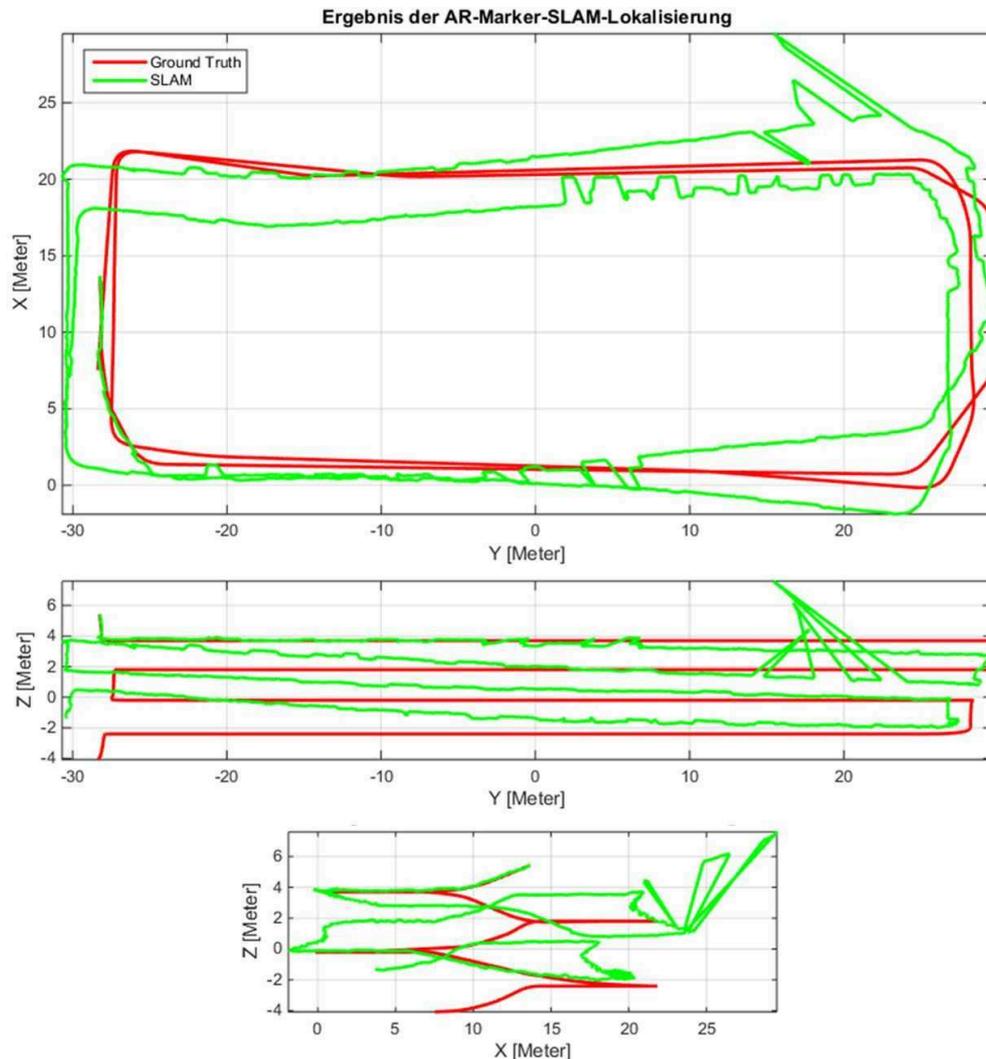


ABBILDUNG 6.11: Ergebnistrajektorie des AR-Marker-basierten SLAM

vereinzelt Stellen mit der Zeit immer mehr von der tatsächlich gefahrenen Bahn (rote Linie in Abbildung 6.11) ab, allerdings lässt sich die grundlegende Form der Bahn doch noch sehr gut in der Rekonstruktion erkennen. Im direkten Vergleich zum Wald-Szenario ist hier zu betonen, dass aufgrund der eingeschränkten Bewegung des Fahrzeugs in der Tiefgarage auf der gesamte Länge der Trajektorie kein Loop-Closing stattfindet. Somit wird auch keine Korrektur der Markerpositionen vorgenommen, sodass sich der Fehler (Drift) mit der Zeit akkumuliert. Die Analyse der Lokalisierungsfehler in Abbildung 6.12 über die Zeit zeigt, dass die erzielte Genauigkeit mit einem translatorischen Fehler von unter

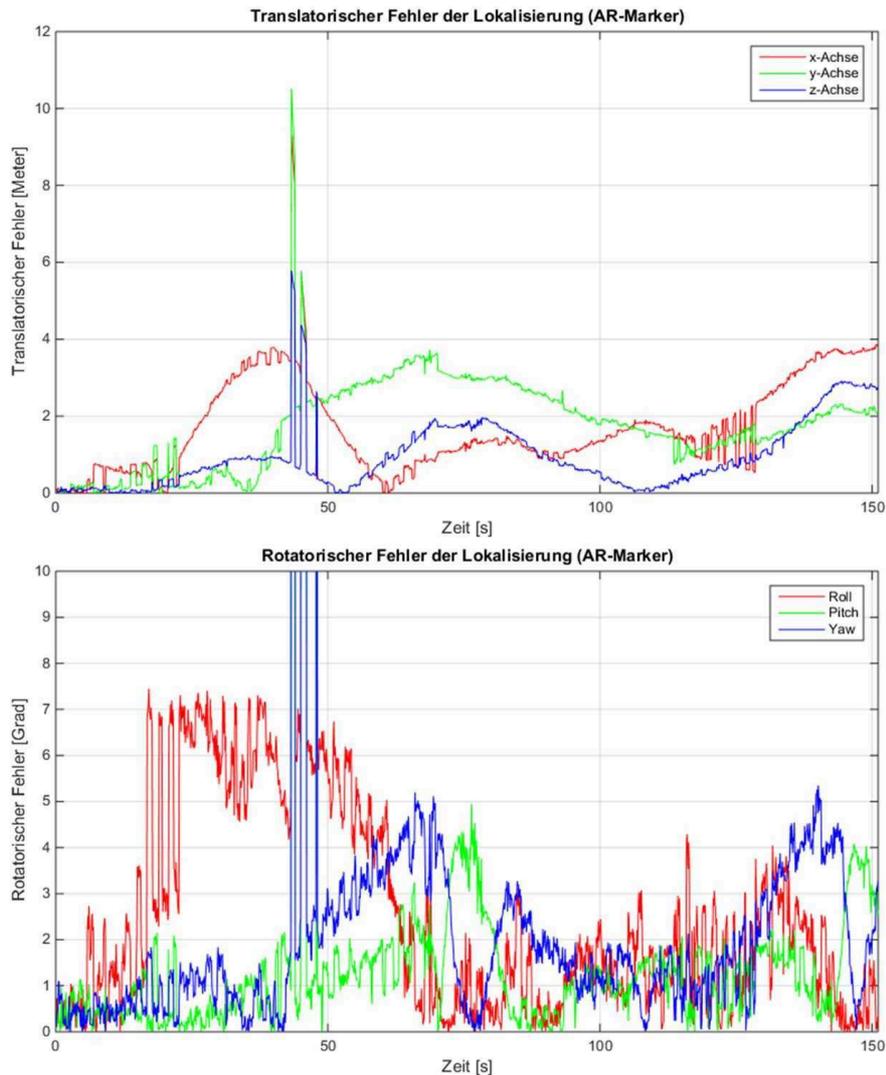


ABBILDUNG 6.12: Auswertung der rotatorischen/translatorischen Abweichung beim AR-Markerbasierten SLAM

4 Metern und einem rotatorischen Fehler von unter  $8^\circ$ , auf der 355 Meter langen Bahn dennoch in Bereichen verbleibt, die eine Anwendbarkeit des Verfahrens nicht ausschließen lassen.

Die Genauigkeit dieses Verfahrens, lässt sich schnell verbessern, indem zusätzlich die Radodometrie des Fahrzeugs verwendet wird, oder die Karte nicht vollständig als unbekannt angenommen wird. So ließe sich eine dünn besetzte Karte erzeugen, die lediglich Marker enthält, die

sich in Auf-/Ab-Fahrten befinden und vom Fahrzeug mit hoher Wahrscheinlichkeit „gesehen“ werden. Durch derartige Stützstellen, könnte man gerade in schwierigen Bereichen die Lokalisierung absichern und den bis dahin aufgetretenen Fehler gewissermaßen zurücksetzen. Die Übertragung einer solchen Karte, mit nur wenigen (<50) Markern ließe sich mit Hilfe eines QR-Codes ziemlich einfach und mit höherer Fehlerkorrektur an das Fahrzeug übermitteln, als eine Karte mit allen Markern. Das reduziert auch den Aufwand des Einmessens dieser Marker und erlaubt damit die Anwendung des Verfahrens auch in temporären Umgebungen, also Gebieten, die nicht über längere Zeit statisch sind, wie in Lagerhäusern, Bergwerken oder auf Baustellen.

Unabhängig von der Verwendung zusätzlicher, externer Messungen und Informationen, ist es ebenfalls möglich die Zuverlässigkeit und Robustheit der Bewegungsrekonstruktion mit Hilfe von Filterung und Prädiktionen, z.B. durch die Nutzung von Kalmanfilter, zu erhöhen. Gerade durch die eingeschränkte, also nicht-holonome, Möglichkeit zur Bewegung von Autos ist die Prädiktion an dieser Stelle besonders gut möglich.

## 6.4 Umsetzung

Besteht eine Karte, wie oben aus semantischen Landmarken mit beschreibenden Attributen, so ist der Informationsgehalt besonders hoch, und wir sprechen von einem semantischen Umweltmodell, das neben der Lokalisierung auch weitere Anwendungsfälle ermöglicht (Die Unterscheidung der Begriffe „Karte“ und „semantisches Umweltmodell“ wurde in Kapitel 3.2 behandelt). Die semantischen Informationen sind nicht nur für die Lokalisierung nutzbar, sondern erlauben ebenfalls eine (semantische) Analyse der Umgebung, denn Attribute, die dabei helfen Objekte voneinander zu unterscheiden, sagen in vielen Fällen auch etwas über den Zustand bzw. die Eigenschaften der Umgebung aus. Beispielsweise erlauben Baumdurchmesser in gewissem Maße die Unterscheidung einzelner Bäume, was die Lokalisierung im Wald vereinfacht und zusätzlich kann aus einem Umgebungsmodell, das aus derart attribuierten Bäumen besteht, der Holzvorrat der Fläche - eine wichtige Größe der Forsteinrichtung - bestimmt werden. Aus diesem Grund sollte bei der Umgebungserfassung nicht ausschließlich der Anwendungsfall

der Lokalisierung entscheidenden Einfluss auf die Entwicklung von Detektionsalgorithmen haben. Stattdessen sollte darauf geachtet werden möglichst viele bestimmbar Attribute zu definieren, die im Laufe der Erfassung gesammelt werden. Daher ist es von Vorteil, eine zu starke Bindung des Umweltmodells an eine spezifische Anwendung zu vermeiden, und stattdessen allgemeine und damit anwendungsübergreifende Umweltmodelle zu erzeugen. Oftmals können diese Attribute bereits während der Detektion implizit mitbestimmt werden. Zum Beispiel werden bei der Baumdetektion mittels Laserscannern ohnehin Cluster von zusammenhängenden Messpunkten gesucht, die in Form eines Kreischnitts angeordnet sind. Die Bestimmung des Kreisdurchmessers ist dann bereits implizit gegeben und kann daher ohne negativen Einfluss auf die Laufzeit des Algorithmus' dokumentiert werden. Selbstverständlich gibt es auch messbare Attribute, die erheblich mehr Aufwand benötigen, als Kreisdurchmesser, aber auch diese müssen bei der Umgebungserfassung nicht zwingend außer Acht gelassen werden, auch wenn sie nicht in jedem Berechnungsschritt bestimmt werden können.

Zur Speicherung von Umgebungskarten werden in der Regel Datenbanken mit räumlichen Datenmodellen (GIS) bzw. spezielle Datenstrukturen zur Haltung räumlicher Daten (z.B. Octrees) verwendet. Diese eignen sich insbesondere zur Verwaltung von Daten, die über räumliche Eigenschaften verfügen. Ein semantisches Umweltmodell besitzt jedoch nicht ausschließlich solche Daten, die über eine räumliche Komponenten verfügen. Stattdessen können auch Beziehungen zwischen einzelnen Objekten der Umgebung Inhalt des Umgebungsmodells sein, die ebenfalls abgespeichert werden sollen. Daher sind sogenannte Graphdatenbanken [186] ideal zur Speicherung von Umgebungsmodellen geeignet. Sie erlauben sowohl die objektbasierte Speicherung von Daten (Knoten) als auch das Abspeichern von Beziehungen der Objekte untereinander ((gerichtete) Kanten). Die in Kapitel 3.3 vorgestellte Simulationsdatenbank VSD der Simulationsumgebung VEROSIM ist eine solche Graphdatenbank und wurde zur Umsetzung der in dieser Arbeit vorgestellten Verfahren genutzt.



## Kapitel 7

# Anwendungen

Das in Kapitel 3 vorgestellte Konzept der semantischen Umwelterfassung, Lokalisierung und Modellierung, sowie dessen Ausführung (Kapitel 4-6) bilden die Grundlage für eine Vielzahl verschiedener Anwendungsmöglichkeiten. Diese verschiedenen Anwendungen sollen in diesem Kapitel im Detail vorgestellt werden, um die Vielfältigkeit des in der vorliegenden Dissertation erarbeiteten Ansatzes zu verdeutlichen.

### 7.1 Forsteinrichtung und Forstbewirtschaftung

Als erstes Anwendungsgebiet soll der Wald und dort insbesondere die Forstwirtschaft dienen. Ausgehend von einer sensorgestützten Forsteinrichtung, Stichprobeninventur und Bestandsaufnahme soll gezeigt werden, wie das durch die semantische Umwelterfassung aufgebaute Modell dazu genutzt werden kann, Fragen und Probleme der Forstwirtschaft anhand einer gemeinsamen Datenbasis zu beantworten. Im Vordergrund stehen hier Themen wie die Waldplanung, Sicherheit bei der Arbeit und ein Controlling, das erstmalig eine Möglichkeit bietet, nach Vorschriften zertifizierter Waldbewirtschaftung (z.B. nach FSC-Zertifizierung<sup>1</sup>) durchgeführte Maßnahmen nachzuweisen.

#### 7.1.1 Automatisierte Waldinventur

Bislang ist eine einzelbaumbasierte Aufnahme von Waldbeständen sowohl finanziell als auch zeitlich in hohem Maße aufwändig. Als Beispiel sei hier die Anlage von Referenzflächen im Rahmen des *Center for Tropical*

---

<sup>1</sup>Forest Stewardship Council: <http://www.fsc-deutschland.de/de-de>

*Forest Science - Forest Global Earth Observatory (CTFS-ForestGEO)*<sup>2</sup>, wozu u.a. auch die Flächen in Traunstein aufgenommen wurde, genannt. Es gibt die Möglichkeit einer Aufnahme von Fernerkundungsdaten durch beauftragte Überflüge oder auf Basis von regelmäßig erhobenen Daten, die zur Einzelbaumidentifikation genutzt werden können. Jedoch sind speziell für diesen Anlass beauftragte Überflüge äußerst Kostenaufwändig und die Informationsdichte der regelmäßig von Diensten wie GeoBasis-Deutschland im Auftrag des Bundesamts für Kartographie und Geodäsie<sup>3</sup> aufgenommenen Daten reicht in der Regel nicht für diese Aufgabe aus.

Die Anwendung des Konzepts der semantischen Umwelterfassung und Kartenerzeugung für terrestrische, mobile Systeme stellt eine, im Vergleich zu den zuvor genannten Möglichkeiten, kosten- und zeitgünstigere Alternative zur Erfassung und Beschreibung von Einzelbäumen dar. Im Rahmen der Forschungsprojekte „Der Virtuelle Wald 3“ und insbesondere im Spin-Off-Projekt „SELOK - Selbstlokalisierung auf planetaren Oberflächen“ wurde die zuverlässige semantische Baumdetektion in erster Linie im Anwendungsfall der Lokalisierung bezüglich einer im Vorfeld gegebenen Karte entwickelt und implementiert.

Hier war man jedoch immer auf eine vorhergehende Befliegung zur Berechnung einer Einzelbaumkarte angewiesen. Zur Bestimmung der erzielten Lokalisierungsgenauigkeit wurden ebenfalls Vermessungsingenieure beauftragt, um eine terrestrische, und damit genauere, Einmessung von Einzelbäumen vorzunehmen. Eine Auswertung der Genauigkeit der Lokalisierung wurde in Kapitel 5.4.3 vorgestellt.

Bei der reinen Lokalisierung werden die Bäume der Umgebung erfasst und mit denen aus der Karte verglichen. Nach dem Lokalisierungsschritt, also dem Aktualisieren der Pose, werden die lokalen Baumbeobachtungen verworfen. Die semantischen Attribute, wie Baumdurchmesser, werden höchstens zur Absicherung des Matchings zwischen erfassten Bäumen und jenen aus der Karte verwendet. Erst durch die Verwendung des in Kapitel 5 der vorliegenden Dissertation vorgestellten Landmarkentrackings ist es möglich, die lokal erfassten Bäume auch ohne zuvor erzeugte Karte für die Lokalisierung zu verwenden.

Durch den simultanen Aufbau einer Karte aus den erkannten Baumlandmarken erwachsen zwei Vorteile. Erstens kann das relative, und

---

<sup>2</sup><http://www.forestgeo.si.edu/>

<sup>3</sup><http://www.geodatenzentrum.de>

somit durch das sogenannte „drifting“ fehlerbehaftete, Landmarken-tracking durch das bereits zuvor validierte und kartenbasierte MCL-Verfahren abgesichert werden. Dies erlaubt ein „Loop-Closing“, wenn zuvor besuchte Bereiche erneut angelaufen werden und korrigiert die mit der Zeit aufgetretenen Fehler. Der zweite Vorteil ist der, dass am Ende der Aufnahme eine Karte aller detektierter und erkannter Bäume, mit samt ihren über die Zeit mehrfach gemessenen und gemittelten Attributen, entstanden ist. Bei einer flächendeckenden Begehung eines Waldbestandes sind somit potenziell alle Bäume des Bestandes erfasst und in die Karte eingetragen worden. Diese Karte, oder besser gesagt diese Bestandsaufnahme kann nun als Grundlage für viele Aufgaben der Forsteinrichtung, Stichprobeninventur und der Waldplanung verwendet werden. Diese Aufgaben werden im Folgenden beschrieben.

### **Anwendung in der Forsteinrichtung**

In der Forsteinrichtung werden Waldflächen in der Regel hierarchisch unterteilt und einzeln beschrieben. Aus diesen Beschreibungen lässt sich feststellen, um welche Baumart es sich jeweilig handelt, wie alt, hoch und dick die Bäume sind und wie viel Holzvorrat es entspricht. In Nordrhein-Westfalen beispielsweise werden diese Daten von Forsteinrichtern der jeweiligen Forstämter erhoben. Dazu werden die einzelnen Forstbetriebsbezirke in sogenannte Abteilungen unterteilt, die anschließend noch einmal in Unterabteilungen untergliedert werden. Die Unterteilung in Abteilungen sind meist noch von verwaltungstechnischer Form. In den einzelnen Unterabteilungen lassen sich schon vegetative oder altersbedingte Unterschiede in den Beständen feststellen. Diese Unterabteilungen werden vom Forsteinrichter, durch eine Begehung und Begutachtung der Flächen in sogenannte Bestandeseinheiten unterteilen. Diese Bestandeseinheiten sind Flächen, die einen weitestgehend homogenen Waldbestand einer Baumart und eines Alters beinhalten. In Abbildung 7.1 ist eine solche hierarchische Einteilung der Flächen in Abteilungen (Zahlen), Unterabteilungen (Großbuchstaben) und Bestandeseinheiten (kleine Buchstaben) zu sehen. Beschrieben werden die einzelnen Bestandeseinheiten durch sogenannte Baumartenzeilen (BAZ). Wichtige Bestandteile dieser BAZ sind jeweils die Baumart, das Alter, und die Höhe der Bäume. Zur Aufnahme dieser Attribute vermisst der Forsteinrichter nicht die einzelnen Bäume sondern begutachtet die Fläche als Gesamtes und

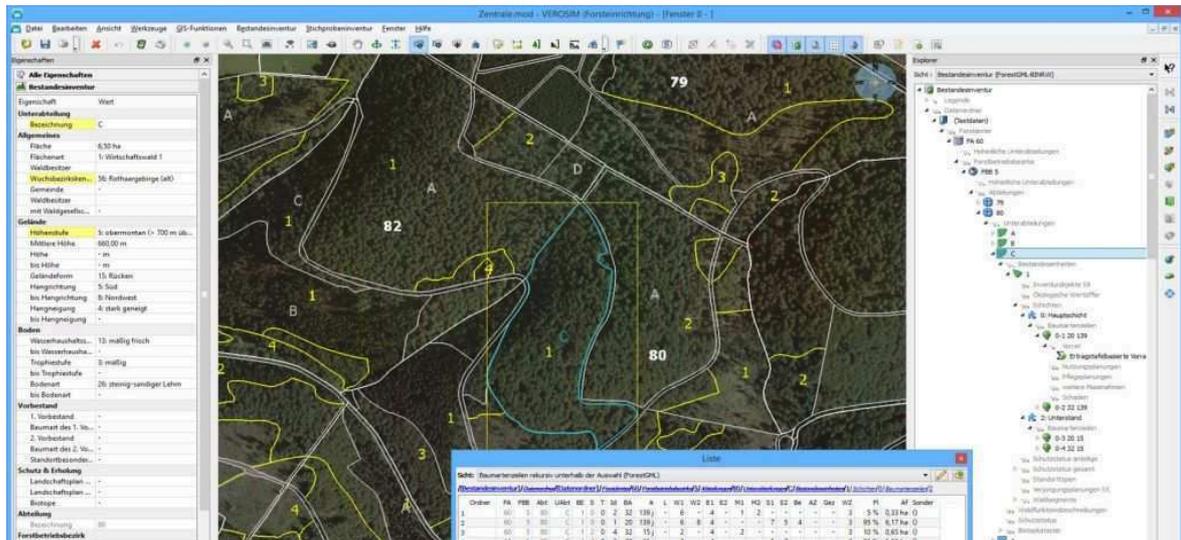


ABBILDUNG 7.1: Waldbestände werden in der Forsteinrichtung in hierarchische Flächen eingeteilt und beschrieben

bedient sich stochastischer Methoden, um die Bestandeseinheit anhand von wenigen Messungen umfassend zu beschreiben. Weitere Attribute können mit der Hilfe verschiedener Verfahren abgeleitet werden<sup>4</sup>.

Ist ein Bestand mit den in dieser Arbeit vorgestellten Methoden einzelbaumbasiert aufgenommen worden, so lassen sich die Attribute der Baumartenzeile größtenteils aus diesen Daten ableiten. Liegen zudem noch Fernerkundungsdaten vor, die eine Bestimmung der Baumart und Höhe für jeden Baum erlauben, so lassen sich die sensorisch erfassten Einzelbäume mit diesen Daten verschneiden und man erhält eine detaillierte und qualitativ der manuellen Aufnahme durchaus ebenbürtige Bestandesbeschreibung. Zur Ableitung fehlender Attribute werden ebenfalls dieselben Tafelwerke verwendet, wie vom Forsteinrichter bei der manuellen Einrichtung. Fehlende Informationen können auch nachträglich noch manuell hinzugefügt werden.

### Anwendung in der Stichprobeninventur

Eine alternative Methode zur Inventurisierung von Waldbeständen stellt die rein stochastische Stichprobeninventur dar. Anstatt den Wald Bestandesweise aufzunehmen, werden, unabhängig von Waldflächen und

<sup>4</sup>In NRW wird dazu in der Regel das Tafelwerk aus dem Buch „Hilfstafeln für die Forsteinrichtung“ [187] verwendet.

Forstbetriebsbezirken, auf einem festgelegten Gebiet Stichprobenpunkte in regelmäßigem Raster angelegt. Die Rasterung wird auf einer Karte, ebenfalls im Vorfeld, festgelegt und richtet sich vielmehr an geographischen Koordinaten als an der tatsächlichen Vegetation. Ziel dieser Einteilung ist eine regelmäßige Abdeckung der Fläche mit Stichprobenpunkte, die einzeln und unabhängig voneinander vermessen und dazu genutzt werden eine Hochrechnung für die gesamte Fläche zu ermitteln. Für das Erheben einer Stichprobe an einem Stichprobenpunkt ist in der Regel viel Zeit und Ausrüstung erforderlich.

Mit dem in dieser Arbeit vorgestellten Ansatz lässt sich die Aufnahme einer Stichprobe folgendermaßen durchführen:

1. Der Stichprobenpunkt selbst wird wie bisher lokalisiert oder es wird ein Einmesspunkt in der Nähe lokalisiert und eingemessen.
2. Auf dem eingemessenen Punkt wird ein künstlicher Marker auf einem Stativ aufgesetzt und mit Hilfe eines Kompass in eine definierte Ausrichtung gebracht. Dieser Marker dient nun als initiale Pose und wird zur markerbasierten Lokalisierung (beschrieben in 5.3) verwendet. Zunächst werden für die Marker-Id die georeferenzierten Koordinaten des eingemessenen Punkts und seine Orientierung abgespeichert, sodass eine eindeutige Lage in der Karte zugeordnet werden kann.
3. Ausgehend von der initialen Markerpose werden nun die Bäume der Umgebung sensoruell erfasst und anschließend georeferenziert in die Karte eingetragen. Dazu wird die relative Lage des Markers zur Sensoreinheit und die relative Lage der Bäume zur Sensoreinheit über die Sensoreinheit und die Initialisierungspose zusammengefasst, sodass die Bäume in globalen (georeferenzierten) Koordinaten angegeben werden können. Abbildung 6.1 veranschaulicht die jeweiligen Transformationen in die Bezugssysteme in Form von Pfeilen. Solange der Marker sichtbar ist und im detektierbaren Bereich der Sensoreinheit liegt, können Bäume direkt über diese Transformationskette in die Karte übernommen werden.
4. Liegt der Marker nicht mehr im Sichtbereich der Sensoreinheit, so werden Bäume über das Landmarkentracking relativ zu den bereits in der Karte verzeichneten Bäumen hinzugefügt. Ein spiralförmiges

Abschreiten des Aufnahmekreises ermöglicht ein stetig wiederkehrendes Betrachten der Bäume aus verschiedenen Blickwinkeln. Dadurch wird der Fehler des relativen Lokalisierungsverfahren mittels Loop-Closing-Strategie möglichst gering gehalten und die erfassten Attribute (z.B. Durchmesser) werden über die verschiedenen Sichtungen aus unterschiedlichen Perspektiven gemittelt.

Auf Basis der erfassten Bäume kann die Stichprobe nun automatisiert erhoben werden. Fehldetektionen, d.h. Bäume mit einer geringen Anzahl an Beobachtungen, werden entfernt und die Erfassungskreise werden in der Karte gezogen. Daraus ergibt sich eine Stichprobe analog zur manuellen Erfassung, wie oben beschrieben.

### 7.1.2 Anwendung in der technischen Produktion

Eine Bestandesaufnahme basierend auf sensorisch aufgenommenen Einzelbäumen kann nicht nur zur Ableitung von Bestandesinformationen genutzt werden, wie in Abschnitt 7.1.1 beschrieben. Die Daten können auch bei der Planung von Maßnahmen im Bestand (z.B. Holzerntemaßnahmen) verwendet werden. Vor einer Hiebmaßnahme beispielsweise wird der Bestand vom Förster begutachtet, und die zu fällenden Bäume aktuell mit Farbe markiert. In Beständen, die vollständig, einzelbaum-basiert aufgenommen wurden, kann diese Auszeichnung bereits am Rechner stattfinden. Die Bäume werden dann digital ausgezeichnet und können vor Ort mittels Sensoren auf die gleiche Weise wie bei der Erfassung lokalisiert werden. In Beständen, wo die Daten in nicht ausreichend hoher Auflösung (Einzelbäume) vorliegen, kann der Förster vor Ort dennoch digital durchführen.

Da er ohnehin die Fläche betreten muss kann er, wie bei der Stichprobeninventur ausgehend von einer definierten Position, die relevanten Bereiche des Bestandes aufnehmen und die zu fällenden Bäume darin markieren. Auf Basis der ausgezeichneten Bäume und des Wegenetzes lassen sich mit Methoden der Holzerntesimulation ganze Durchforstungsszenarien im Vorhinein simulieren (siehe Abbildung 7.2) und somit eine kosten-, zeit- und bodenbelastungseffiziente Durchführung berechnen lassen. Als Fahrer eines Harvesters ist es dann möglich, sich die optimale Halteposition für weit im Bestand befindliche Bäume berechnen zu lassen, und sich anschließend vom System an die richtige Position navigieren zu lassen.

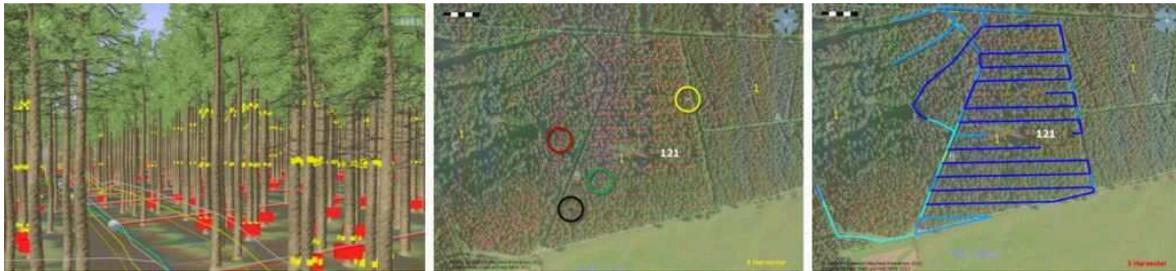


ABBILDUNG 7.2: Holzerntesimulator auf Basis einer Einzelbaumbasierten Bestandesbeschreibung. Links: Simulationsgestützte Auszeichnung zu fällender Bäume inkl. Fixlängen zur Kurzholzsortimentierung; Mitte: Simulation des Erntebereichs bei unterschiedlichen Anzahlen an Holzerntemaschinen; Rechts: Erwartete Bodenbelastung während der Holzernte abhängig von der Anzahl der beteiligten Holzerntemaschinen.

Bilder: D. Losch, MMI

Zur nachhaltigen Forstwirtschaft nach dem Forest Stewardship Council (FSC) gehört u.a. eine bodenschonende Bewirtschaftung der Waldflächen. Das schließt die Verwendung von Forstmaschinen wie Harvester, Forwarder, etc. zwar nicht aus, schreibt ihr aber gewissen Regeln vor. In erster Linie dürfen die schweren Maschinen nur die für sie gekennzeichneten Wege – Forststraßen und sogenannte Rückegassen – befahren und diese auch nicht verlassen. Diese Rückegassen werden in der Planungsphase der Waldbewirtschaftung zunächst in Karten verzeichnet und später vor Ort erschlossen, d.h. abgesteckt und kenntlich gemacht. Der festgelegte Verlauf der Rückegassen darf später nicht mehr verändert werden, sodass diese mit größter Sorgfalt angelegt werden müssen. Bei der Neuplanung müssen vorhandene Fahrspuren in das Feinerschließungsnetz integriert werden, dabei können alte Überfahrlinien oftmals nur schwer zu erkennen sein und müssen dann anhand Vegetation (durch Zeigerpflanzen) oder Bodenveränderungen (z.B. Verfärbungen) neu gesucht werden. Um die Fahrlinien permanent kennzeichnen zu können ist daher eine verortete, digitale Speicherung in Karten notwendig, die unabhängig von der Witterung Bestand hat. Diese Speicherung ist momentan äußerst schwierig, da kein derzeit verfügbares System eine submetergenaue absolute Positionierung im Wald ermöglicht. Daher ist zurzeit davon auszugehen, dass die satellitenbürtigen Daten ergänzt werden müssen mit Messungen, die sich auf Referenzpunkte am Boden



ABBILDUNG 7.3: Harvester im semantisch erfassten Wald. Die Bestandes- und Einzelbauminformationen des semantischen Weltmodells dienen als Navigationsgrundlage für den Fahrer.

beziehen Purfürst et al. schlagen in [188] zwei grundsätzlich unterschiedliche Prinzipien vor:

1. Durch die Ermittlung der Relativbewegung zu bekannten Objekten. Das können z.B. stehenden Bäumen wie in (2), oder für diesen Zweck ausgesetzte, temporäre Marker (Farbmarkierungen, Tonkinstäbe, etc.) sein.
2. Durch den Nachvollzug der räumlichen Ortsveränderung, z.B. durch Odometrie oder Inertialsensorik (IMU).

Besser ist hier eine Kombination beider Verfahren, die es ermöglicht die Nachteile beider Ansätze auszugleichen und die Vorteile effektiv zu fusionieren. Ist ein Verfahren zur Erstellung einer digitalen Wegekarte implementiert, so kann dieses nicht nur dazu verwendet werden, die angelegten Gassen wiederzufinden sondern ebenfalls den Fahrer dabei unterstützen diese Wege einzuhalten und durch Logging-Mechanismen die Feinerschließung nach den Richtlinien des FSC zu kontrollieren und nachzuweisen.

## 7.2 Explorationsmissionen

In diesem Abschnitt werden Explorationsmissionen näher betrachtet, die aufgrund von Gefahrenvermeidung, Sicherheit aber auch aus ökonomischen Aspekten nicht von Menschen durchgeführt werden können, und deshalb mobile Roboter eingesetzt werden. Diese von mobilen Robotern durchgeführten Missionen haben verschiedene Ziele und Herangehensweisen, jedoch teilen sie sich das Problem der sensoruell zu erfassenden Umgebung des Roboters und die anschließende Verarbeitung dieser Daten im Kontext der jeweiligen Mission. In diesen Anwendungsfällen soll näher auf die extraterrestrische Exploration auf fremden Planeten durch mobile Roboter eingegangen werden.

### 7.2.1 Anwendung in extraterrestrischen Explorationsmissionen

Extraterrestrische Planeten und Himmelskörper wurden und werden (bis auf den Mond) nicht von Menschen erkundet, da u.a. sowohl die Reise dorthin als auch der Aufenthalt dort zu gefährlich wäre. Daher werden mobile Roboter für diese Aufgabe verwendet. Sie werden mit zahlreichen Werkzeugen und Sensoren ausgestattet, um auf der Oberfläche Messungen durchzuführen, die von der Erde angestoßen werden. Die Roboter werden meist von der Erde aus gesteuert. Da die Übertragung von Signalen zu den extraterrestrischen Explorationsgebieten mehrere Minuten dauern und die Verbindung aufgrund der Rotationen der Himmelskörper regelmäßig unterbrochen wird, müssen die Roboter manche Aufgaben teilweise autonom durchführen. Was bei der Durchführung von Messungen und der Analyse von Proben in der Regel keine Gefahr für den Roboter bedeutet, kann aber bei der unbeaufsichtigten Bewegung zu Problemen führen. Selbst wenn der Roboter mit einer mehrminütigen Verzögerung ferngesteuert wird, muss er den zurückgelegten Weg selbst erfassen, damit der Operateur auf der Erde weiß, wohin der Roboter sich tatsächlich bewegt hat und wo er sich aktuell befindet. In unwegsamem Gelände kann ein Befehl, der den Roboter zehn Meter nach vorne fahren soll in einer Bewegung resultieren, die nicht geradlinig verläuft und signifikant kürzer als zehn Meter ist. Einige solcher Fälle werden in [121] beschrieben, nachdem das Mars Science Lab (MSL) mit dem Curiosity Rover (Abbildung 7.4) zwei Jahre lang

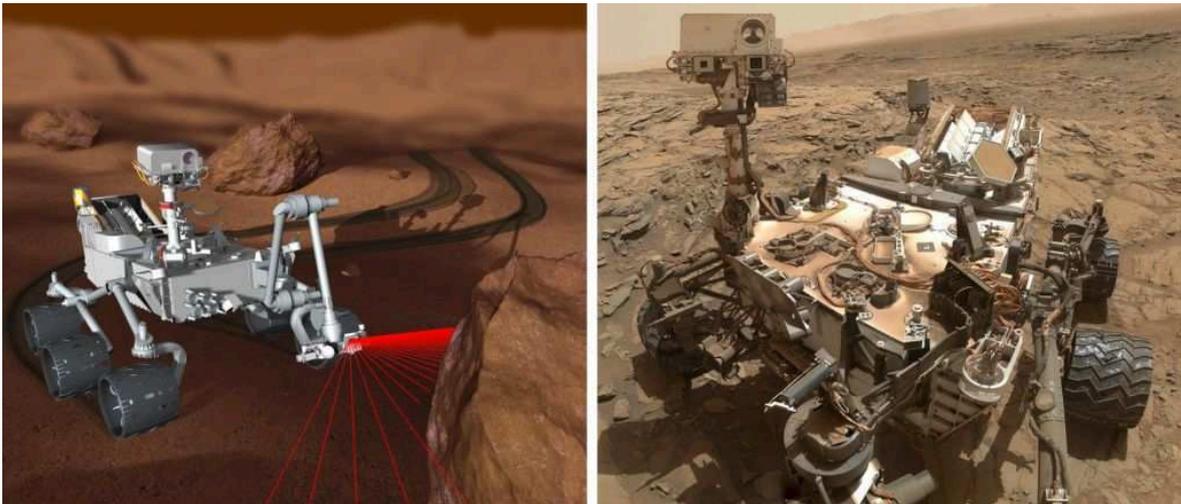


ABBILDUNG 7.4: Curiosity-Rover des Mars Science Laboratory (MSL) mit realer Aufnahme (©NASA) und simuliert in virtueller Explorationsmission

auf dem Mars in Betrieb war. Basierend auf diesen Erkenntnissen wird ebenfalls in [121] die Notwendigkeit von Lokalisierungsmethoden für mobile Roboter auf extraterrestrischen Oberflächen betont. Zunächst war das Visual Odometry-Verfahren lediglich als optionales Verfahren auf dem Curiosity-Rover implementiert worden. Nachdem es erfolgreich gegen die auftretenden Fehler bei der bis dahin primär verwendeten Odometriemessung der Räder eingesetzt wurde, hat es für die Explorationsmission einen hohen Stellenwert erlangt.

Im Rahmen der vorliegenden Dissertation wurde die landmarkenbasierte Lokalisierung aus dem Anwendungsfall „Wald“ um zusätzliche Landmarkendetektoren erweitert und für extraterrestrische Explorationsmissionen erweitert. Die Ergebnisse des hier vorgestellten, anwendungsunabhängigen Lokisierungsframeworks haben wesentlich zum Erfolg des Forschungsprojekts „SELOK - Selbstlokalisierung von mobilen Robotern auf planetaren Oberflächen“<sup>5</sup> beigetragen. An Stelle von Bäumen wurden Steine als wesentliche Landmarken verwendet. Während der Landephase können Steine, Krater und größere Erhebungen bereits erkannt und, ähnlich wie bei der Verwendung von Fernerkundungsdaten

<sup>5</sup>Gefördert von der Raumfahrt-Agentur des Deutschen Zentrums für Luft- und Raumfahrt e.V. mit Mitteln des Bundesministeriums für Wirtschaft und Technologie aufgrund eines Beschlusses des Deutschen Bundestages unter dem Förderkennzeichen 50 RA 0911

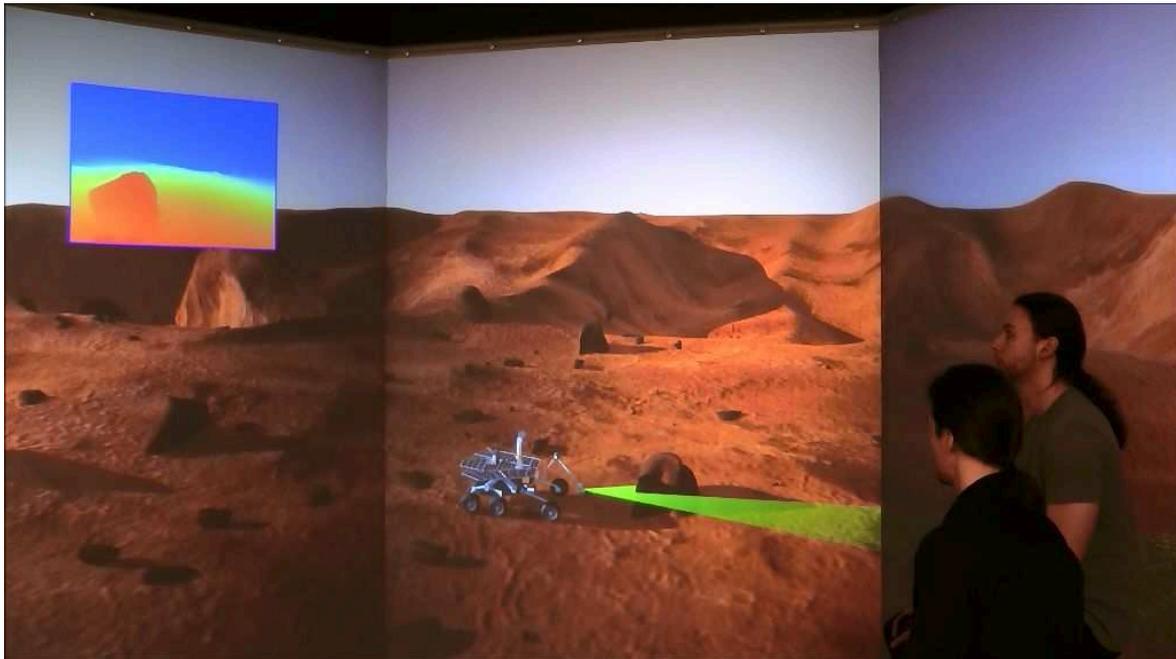


ABBILDUNG 7.5: Interaktives Planetenexplorationsszenario in der 3D-Simulation auf einer Projektionswand.

im Wald, in eine Karte geschrieben werden. Die Erzeugung einer solchen Karte wurde in einem begleitenden Forschungsprojekt („FastMap - Schnelle 3-D Kartengenerierung für planetare Lande- und Explorationsoperationen“) untersucht und ebenfalls erfolgreich umgesetzt.

Durch die semantische Erfassung der Umgebung und die anschließende Abstraktion zu Landmarken ist es möglich in diesem Framework dieselben Lokalisierungsalgorithmen zu verwenden, wie bereits bei der automatisierten Waldbestandsaufnahme (siehe 7.1). Lediglich die Landmarkendetektoren wurden ausgetauscht, sodass im Wald Bäume und im Explorationsszenario Steine detektiert werden. Abbildung 5.3 zeigt eine Gegenüberstellung der Konfiguration des Lokalisierungsframeworks für die Szenarien „Wald“ bzw. „Planetenerkundung“.

Da extraterrestrische Explorationsmissionen nur durch viel Aufwand und hohe Kosten auf der Erde in physikalischen Mockups nachgestellt werden können, ist die Simulation ein wesentlicher Teil dieser Anwendung. Wichtig ist hier eine ausreichend hoch detaillierte Umgebung und eine äußerst realistische Sensorsimulation. Nur durch das Zusammenspiel dieser beider Möglichkeiten, können Szenarien dieser Komplexität ausreichend genau simuliert werden.

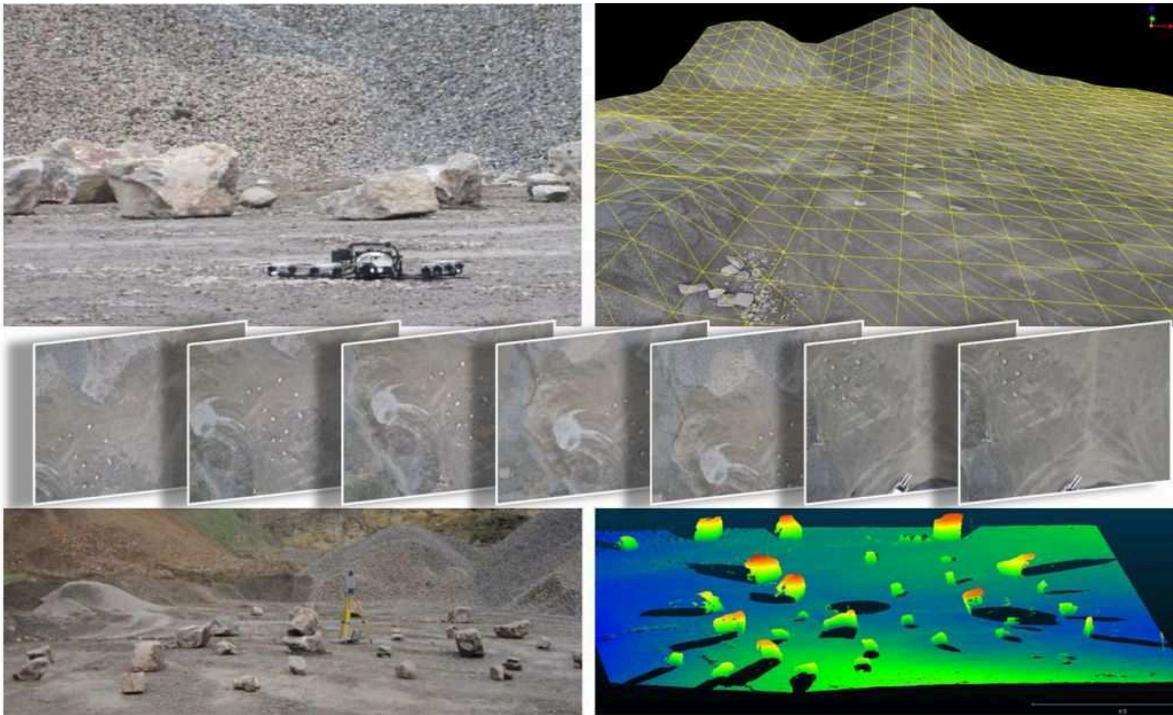


ABBILDUNG 7.6: Ein Reales Testbed für Planetenexplorationen in einem Steinbruch. Der Testbereich wurde vor den Tests mittels Flugdrohne (o.l.) befliegen und aus den Bilddaten (Bilderreihe oben) ein grobes Umgebungsmodell (o.r.) erzeugt. Genauere Messungen (u.r) wurden durch die Verwendung eines 3D-Laserscanners (u.l.) aus verschiedenen Standpunkten erzeugt.

Im Rahmen dieser Anwendung wurden die entwickelten Detektions- und Lokalisierungsmethoden anhand der mobilen Roboterplattform SeekurJr in einem Steinbruch getestet und die simulierten Ergebnisse verifiziert. Dazu wurde ein Bereich des Steinbruchs abgesperrt und vereinzelte Steine als Landmarken verwendet. Für die Erhebung von Referenzdaten wurde der Bereich sowohl mit einer Flugdrohne abgeflogen und überlappend fotografiert, als auch mit einem 3D-Laserscanner von verschiedenen Standorten aus gescannt. Abbildung 7.6 zeigt das Testareal im Steinbruch während der Referenzmessungen und die jeweiligen Ergebnisse in der Simulation.

Dieses Testareal konnte der Roboter durchfahren und mit den Sensoren der SELMA-Einheit (Laserscanner und Stereokamera) die Umgebung erfassen. Abbildung 7.7 zeigt den Roboter mit der SELMA-Einheit im



ABBILDUNG 7.7: Reales Testbed für Planetenexploration in Steinbruch

Steinbruch. Zur Lokalisierung wurden die sensorisch detektierten und erkannten Steine als Landmarken verwendet. Ein Ergebnis der Steindetektion ist in Abbildung 7.7 unten zu sehen: Das Overlay links-oben zeigt hier das linke Bild einer Stereoaufnahme. Die aus der Stereoaufnahme generierte Punktwolke sowie die darin erkannten Steine sind auf der rechten Seite visualisiert. Eine Bildmaske der erkannten Steine, passend zum linken Bild der Stereoaufnahme, befindet sich im Overlay links-unten.

## 7.3 Automatisierte Gebäudevermessung

Laut [189] wird mit der neuen Planungstechnologie Building Information Modeling (BIM) die Digitalisierung als ein großer Schritt hin zum industrialisierten Bauen genutzt, ohne dass es wesentliche Einschränkungen der Gestaltungsfreiheit gibt. Dabei eignet sich BIM nicht nur für Neubauten, sondern im Zusammenspiel mit Laserscannern auch und gerade für das Bauen im Bestand nach der Cradle-to-Cradle-Technologie<sup>6</sup>.

<sup>6</sup>Der Begriff „Cradle-to-Cradle-Technologie“ wurde in [190] vom Chemiker Michael Braungart und vom Architekt William McDonough eingeführt. Sie entwickeln darin eine Definition für eine nachhaltige Bauweise im Sinne einer Ökoeffektivität, d.h. angelehnt an die Begriffe Ökobilanz (chemische Wiederverwertung von Nährstoffen in einem biologischen Kreislauf in Zellen von Lebewesen) und Ökoeffizienz (Quotient aus dem wirtschaftlichen Wert eines Produktes und der während der Herstellung auf die Umwelt auswirkende Belastung).

Dabei unterscheidet sich das Building Information Model von der bislang üblichen CAD-Planung für Gebäude, denn es werden hier nicht nur die Geometriedaten des Gebäudes modelliert und in Form von 2D-Plänen weitergereicht, sondern verknüpft mit den Geometrien werden weiterführende, semantische Informationen abgespeichert, die den einzelnen Elementen des Modells eine Bedeutung geben. Somit hat sich mit dem BIM eine ähnliche Modellierungsmethodik entwickelt, wie in anderen Anwendungsbereichen und insbesondere wie in der semantischen Umweltmodellierung, die grundlegender Bestandteil der vorliegenden Arbeit darstellt (vgl. Abschnitt 3.2).

Im Folgenden sollen zwei konkrete Anwendungsfälle in diesem Bereich betrachtet und dort die Nutzung der in dieser Arbeit entwickelten Methoden vorgestellt werden.

### 7.3.1 Anwendung im Fabrik- und Anlagenbau

Die Planung und der Bau von Fabriken und Industrieanlagen soll durch BIM besser miteinander verknüpft werden, als bisher. Damit das geschehen kann, ist ein regelmäßiger Abgleich der Planungsdaten mit dem tatsächlichen Baufortschritt erforderlich. Um diesen Abgleich zu ermöglichen sind regelmäßige Messungen durch entsprechende Vermessungsingenieure notwendig. Dabei dienen diese Messungen nicht ausschließlich der Kontrolle und Einhaltung von Planungsdaten. Auch zusätzliche Daten, die bei der Planung nicht im Detail berücksichtigt werden, wie der tatsächliche Verlauf von Versorgungsleitungen für Strom, Telekommunikation, Wasser, Abwasser, Gas, etc., können auf diese Weise in das Modell aufgenommen werden, um spätere Wartungs- und Reparaturtätigkeiten zu unterstützen (siehe hierzu auch nächstes Kapitel 7.3.2).

Hierbei ist genau abzuschätzen, in welchen Intervallen diese Messungen durchzuführen sind, denn auf früh erkannte Fehler und Mängel kann direkt reagiert werden, bevor dadurch große Verzögerungen im Bauprozess oder hohe Zusatzkosten entstehen. Auf der anderen Seite ist jede Messungen sowohl mit Kosten als auch mit Totzeiten im Bauprozess

verbunden, denn für die Messung werden in der Regel Fachleute beauftragt und eine Gesamtmessung kann bis hin zu einigen Stunden<sup>7</sup> dauern. Aus diesem Grund bietet sich schon einmal die Messung außerhalb gewöhnlicher Bauzeiten, z.B. nachts, an. Des Weiteren ist ein voll- oder teilautomatisierter Ablauf der Aufnahme eine sinnvolle Entwicklung, um die Zeit für Fachpersonal zu reduzieren.

Eine mögliche Lösung für dieses Problem ist die regelmäßige Vermessung durch einen automatisierten Roboter, der außerhalb der Bauzeiten (z.B. nachts, an Feiertagen oder an Wochenenden) autonom auf die Baustelle fährt und 3D-Laserscans von verschiedenen Aufnahmepunkten vornimmt und diese Daten mit den eigenen Lokalisierungsdaten bereits referenziert und anschließend algorithmisch nachbearbeitet. Als Ergebnis könnte eine erste Auswertung und ein grober Vergleich der Aufnahme mit den Planungsdaten am nächsten Morgen bei der zuständigen Person auf dem Tisch liegen.

Abbildung 7.8 skizziert einen solchen Ablauf. Als Beispiel dient hier die Demofabrik des Werkzeugmaschinenlabors (WZL) der RWTH Aachen University<sup>8</sup>. Dieses Fabrikgebäude Abbildung 7.8 oben am Campus der Universität, in dem Prototypen und Produkten in Vorserie produziert werden, wurde bereits in Form eines BIM-Modells digital modelliert (Abbildung 7.8 zweite Zeile links). Die mobile Plattform SeekurJr wurde für diesen Anwendungsfall, neben den für die Lokalisierung notwendigen Sensoren zusätzlich mit zwei 3D-Laserscannern bestückt (Abbildung 7.8 zweite Zeile rechts). Der Multi-Layer Laserscanner der Firma Velodyne, misst simultan in 16 Ebenen und einem vertikalen Öffnungswinkel von 30° die Entfernung zu allen Objekten im Umfeld. Mit 5-20 360°-Messungen in der Horizontalen ist er für dynamische Scans während der Fahrt konzipiert, die mit den Lokalisierungsdaten des Roboters verschnitten und akkumuliert werden können um eine dichte Punktwolke um den Arbeitsbereich des Roboters zu erzeugen. Abbildung 7.8 dritte Zeile rechts zeigt dagegen das Ergebnis mehrerer Messungen mit einem, ebenfalls auf dem Roboter SeekurJr montierten 3D-Laserscannern vom Typ „Imager 5006i“ der Firma Zoller+Fröhlich. Hier dauert ein vollständiger

---

<sup>7</sup>Während eines 3D-Laserscans sollte die Umgebung im besten Fall vollständig geräumt sein, und darf währenddessen nicht betreten werden, denn jedes dynamische Objekt (bewegte Objekte während eines Scans bzw. zwischen einzelnen Scans) erschwert die anschließende Analyse der Messdaten.

<sup>8</sup><http://demofabrik-aachen.rwth-campus.com>

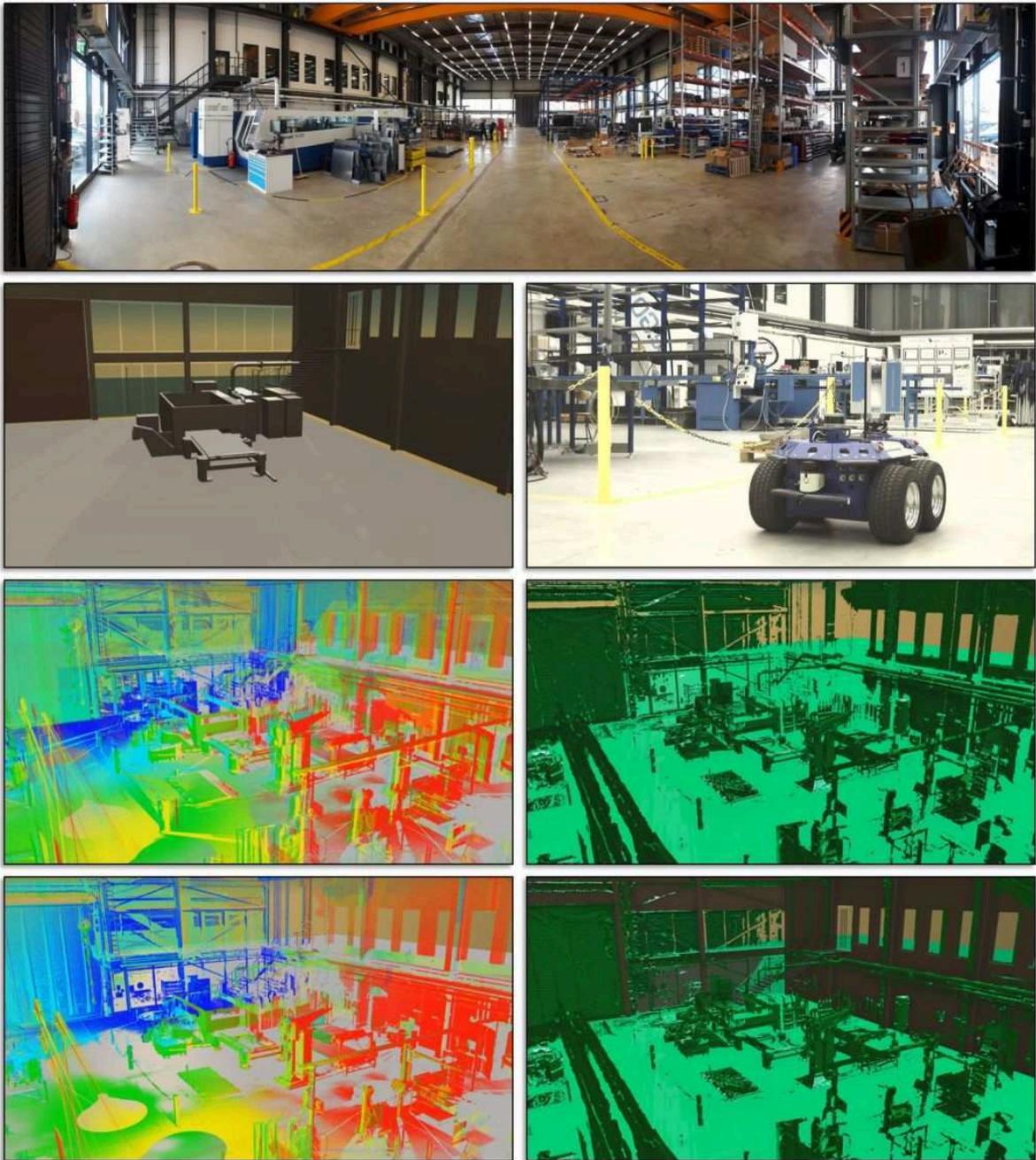


ABBILDUNG 7.8: Anwendung im Fabrik- und Anlagenbau. Um die BIM erzeugten Planungsdaten mit realen Scans abzugleichen, können mobile Roboter mit Messsensorik verwendet werden. Diese nehmen 3D-Laserscans auf, registrieren die Scans untereinander und erlauben einen Vergleich der Mess- und Planungsdaten

Rundumscan ( $310^\circ/360^\circ$  vertikal/horizontal), je nach Auflösung, bis hin zu ungefähr 30 Minuten, weshalb sich die Trägerplattform während dieser Zeit nicht bewegen darf und auch die Szene statisch verbleiben muss. Aus diesem Grund sollten sich während der Aufnahme kein Mensch im Scanbereich aufhalten. Um Verschattungen in der Aufnahme zu reduzieren muss die Szene von verschiedenen Standorten einzeln gescannt werden. Dazu fährt der Roboter einzelne Positionen der Reihe nach und nutzt die eigenen Lokalisierungsdaten, um die Scans geeignet in ein im Vorfeld festgelegtes Koordinatensystem zu registrieren. Ein Ergebnis aus fünf anhand reiner Odometriedaten des Roboters registrierter Einzelscans ist in Abbildung 7.8 dritte Zeile rechts abgebildet. Durch eine automatisierte Bodenerkennung und -filterung (siehe hierzu 4.2.3) und anschließende, ebenfalls automatisierte, Feinregistrierung mittels Iterative Closest Point-Algorithmus (ICP) ergibt sich daraus eine einheitliche Gesamtpunktwolke (Abbildung 7.8 letzte Zeile links), die für den Abgleich mit dem BIM-Modell bestens geeignet ist. Eine Mesherzeugung auf der Punktwolke erlaubt zudem die Detektion und Beschreibung von planaren Flächen in der Punktwolke, die einen Abgleich mit dem BIM-Modell in manchen Fällen erleichtert. Das Ergebnis eines solchen Meshs zeigt Abbildung 7.8 letzte Zeile rechts. Durch die Verschneidung existierender BIM-Daten mit dem Ergebnis des 3D-Laserscan-Prozesses, lassen sich die semantischen Informationen aus den BIM-Daten auf die reale Messung übertragen und so stehen dem Roboter neben den Lokalisierungsergebnissen ebenfalls wichtige Gebäudeinformationen zur Verfügung, die beispielsweise im Anwendungsfall des Gebäudemanagements weitere Verwendung finden.

### 7.3.2 Anwendung im Gebäudemanagement

Der grundlegende Zweck und Unterschied von BIM-Modellen zu gewöhnlichen Bauplänen ist die intensive Verwendung des Modells und die Dokumentation weit über die Bauphase hinaus. Insbesondere gehört u.a. die Verwaltung des Gebäudes im Bereich des Gebäudemanagements (Facility Management) dazu, denn BIM soll während des gesamten Existenzzyklus eines Gebäudes („Cradle-to-Cradle“) Verwendung finden. Dieser Ansatz der Abbildung des Gebäudes in Form eines digitalen Modells, oder, in der Terminologie von Industrie 4.0, als digitaler Zwilling, erlaubt eine direkte Protokollierung und Dokumentation von Wartungs-

und Reparaturarbeiten am Gebäude bzw. an dessen Versorgungsleitungen für Telekommunikation, Strom, Wasser, Abwasser, Gas, etc.. Somit können Defekte und Störungen sofort durch entsprechende Sensorik automatisch detektiert und gemeldet, aber auch manuelle Meldungen im BIM-Modell manuell dokumentiert werden. Auf diese Weise steht dem Techniker, der die Mängel beheben soll auch nicht nur eine losgelöste Fehlerbeschreibung zur Verfügung, sondern er kann bereits im Modell die dadurch betroffenen Bereiche des Gebäudes identifizieren bzw. vorübergehende Maßnahmen einleiten.

Auch zur Lokalisation der Fehlerursache, kann das BIM-Modell herangezogen werden. Für das Auffinden entsprechender Leitungen im Gebäude kann ein mobiler Roboter verwendet werden, der das BIM-Modell als Karte zur Indoorlokalisierung (siehe nächster Abschnitt 7.3.3, insbesondere 7.3.3) verwenden, dem Techniker aber darüber hinaus auch weiterführende Informationen mit Hilfe von Augmented Reality anzeigen kann, wie in Abbildung 7.9 gezeigt.

Wurde die fehlerhafte Leitungen lokalisiert, so steht der mobile Roboter ebenfalls als Plattform für Messensorik, Stromquelle oder zum Transport schwerer Instrumente oder Ersatzteile zur Verfügung. Die weiteren Anwendungsgebiete zur Lokalisierung und Navigation von Robotern in Gebäuden sollen nun im anschließenden Abschnitt näher betrachtet werden.

### 7.3.3 Indoorlokalisierung und Navigation

Durch die Entwicklungen der letzten 10 Jahre im Bereich der mobilen Robotik, der Miniaturisierung und Leistungssteigerung von Prozessoren und der vermehrten Erhältlichkeit von Sensoren, sind mobile Roboter nicht mehr ausschließlich im Bereich der Forschung anzutreffen. Haushaltsroboter für Aufgaben, wie Rasenmähen, Staubsaugen und Putzen gehören mittlerweile schon zum Alltag und sind in den meisten Elektromärkten bereits erhältlich. Im Industriebereich werden fahrerlose Transportsysteme für die Logistik innerhalb großer Lagerhäuser eingesetzt, um einfache Transportaufgaben zu automatisieren und damit Personalkosten zu senken und Lagerfläche zu maximieren.



ABBILDUNG 7.9: Beispiel für eine Augmented Reality Anwendung im Gebäudemanagement. Oben v.l.n.r.: SeekurJr auf einem mit AR-Markern ausgestatteten Flur; Kameraposition am Roboter; Bildaufnahme der Roboterkamera. Unten v.l.n.r.: Digitaler Zwilling des SeekurJr an der anhand der Marker geschätzten Position im Virtuellen Testbed; Bild der simulierten Kamera am SeekurJr; Transparentes Overlay der im Modell gespeicherten Strom- und Heizungsleitungen.

Diese mobilen Systeme werden in Bereichen eingesetzt, wo Satellitennavigations-Empfänger in der Regel keinen Empfang haben, d.h. innerhalb von Gebäuden oder unter der Erde, und demnach keine satellitengestützte Lokalisierung möglich ist.

In diesem Abschnitt sollen Anwendungen, die in den Rahmen des relativ weitläufigen Begriffs der professionellen Servicerobotik fallen, näher betrachtet werden. Die Servicerobotik stellt an dieser Stelle ein Sammelbegriff für Roboter dar, die bestimmte Aufgaben ausüben. Nach Standard 8373 aus dem Jahr 2012 der International Organization for Standardization (ISO) [191] umfasst der Begriff „Serviceroboter“ alle Roboter, die nützliche Aufgaben für Menschen oder Apparaturen ausüben, die aber nicht unter industrielle Automatisierungsanwendungen fallen. Dabei beinhalten industrielle Automatisierungsanwendungen hier: Produktion, Inspektion, Verpacken und Montage, ohne darauf beschränkt

zu sein. Der ISO Standard unterscheidet nun weiter zwischen Servicerobotern für den persönlichen Gebrauch, wie automatisierte Rollstühle oder Staubsaugerroboter, die gewöhnlich von Laien bedient werden und professionellen Servicerobotern die im kommerziellen Umfeld eingesetzt, und normalerweise von entsprechend ausgebildetem Bedienpersonal betrieben werden. Zu den professionellen Servicerobotern gehören beispielsweise Reinigungsroboter für öffentliche Gebäude, Auslieferungsroboter in Büros oder Krankenhäusern oder fahrerlose Transportsysteme in Lagerhäusern.

### **Anwendung in der Servicerobotik**

Im Folgenden werden Auslieferungsroboter in öffentlichen Gebäuden, die gleichzeitig dieselben Wege zusammen mit Menschen verwenden und daher über ein hohes Maß an Sicherheit verfügen müssen, und fahrerlose Transportsysteme in Lagerhäusern als konkrete Anwendung der Servicerobotik im Detail betrachtet.

In großen, öffentlichen Gebäuden, wo eine Vielzahl von verschiedenen Abteilungen zusammenarbeiten, werden oftmals Güter wie Verbrauchsmaterial, Dokumente oder Arbeitsutensilien zwischen diesen Abteilungen transportiert. Gerade bei großen Einrichtungen wie etwa Krankenhäusern, sind die Wege zwischen den einzelnen Abteilungen unter Umständen sehr lang und Botengänge werden von Mitarbeitern der jeweiligen Abteilungen zusätzlich zur eigentlichen Arbeit erledigt. In diesen Gebäuden lohnt sich oftmals der Einsatz spezieller Auslieferungsroboter, die diese Aufgaben automatisiert übernehmen können.

Hier ist es wichtig ein System für die Lokalisierung und Navigation zu verwenden, das eine parallele Nutzung der Gänge für Menschen und Roboter erlaubt und dabei die Sicherheit der Menschen sicherstellt und keine baulichen Veränderungen des Gebäudes voraussetzt. Für diesen Fall bietet sich an, ähnlich wie bei der Anwendung im Facility Management, eine markerbasierte Lokalisierung zu nutzen, da die Ausrüstung von bestehenden Gebäuden mit diesen einfach ausdrückbaren 2D-Markern keinen Einfluss auf die eigentliche Infrastruktur hat, und dennoch zu zuverlässigen Ergebnissen bei der Lokalisierung führt (Abbildung 7.10 rechts). Sind bereits andere, semantisch nutzbare Landmarken vorhanden, ist eine nachträgliche Anbringungen künstlicher Marker unter Umständen nicht notwendig. Beispiele für semantische Objekte in



ABBILDUNG 7.10: Beispiele zur Servicerobotik in öffentlichen Gebäuden. Links: Semantische Objekte, die zur Lokalisierung geeignet sind. Rechts: Mobiler Roboter, der sich anhand künstlicher Marker orientiert.

öffentlichen Gebäuden, die zur Lokalisierung genutzt werden können, sind Schilder, Feuerlöscher, Beschriftungen und so weiter. Abbildung 7.10 links zeigt ein paar dieser Beispiele im Inneren eines Krankenhauses.

### Anwendung als Besucherleitsystem

Öffentliche Gebäude, die viele verschiedene Abteilungen mit unterschiedlichen Anforderungen beherbergen können oftmals unübersichtlich sein. Das gilt nicht nur für Roboter wie im vorhergehenden Beispiel der Servicerobotik, sondern auch für Menschen, die nicht alltäglich in diesen Gebäuden arbeiten. Als Beispiel soll hier konkret das Uniklinikum Aachen genannt werden, das durch seine Größe 34 Fachkliniken, 27 Instituten und fünf fachübergreifenden Einheiten beherbergt. Zusätzlich befinden sich u.a. noch Labore, Hörsäle, Wäscherei, Mensa, etc. im Haus. Insgesamt arbeiten rund 7000 Mitarbeiter im Haus<sup>9</sup>. Das Krankenhaus besitzt eine Nettotonutzfläche von 130.000  $m^2$ , verteilt auf acht Stockwerken, die über 24 Treppenhäuser und Fahrstühle erreichbar sind. Das Haus erstreckt sich vom Haupteingang aus gesehen auf ca. 130 Meter Länge, und ist in 4 Flure (bezeichnet mit A-D) unterteilt, die sich über die gesamte Breite des Gebäudes (ca. 240 Meter) mit Treppenhäusern im

<sup>9</sup>Die Daten wurden dem „Fact Sheet“ der Uniklinik RWTH Aachen mit Stand vom 25.07.2016, entnommen ([https://www.ukaachen.de/fileadmin/files/global/Fact\\_Sheet.pdf](https://www.ukaachen.de/fileadmin/files/global/Fact_Sheet.pdf)).



ABBILDUNG 7.11: Das Uniklinikum der RWTH Aachen besitzt vier Hauptflure, die mit den Buchstaben A-D bezeichnet werden und jeweils 6 Treppenhäuser im Abstand von etwa 40 Metern beseiten. Die Treppenhäuser werden von links nach rechts je Flur von 1-6 bezeichnet.

Abstand von ca. 40 Metern aufgeteilt. Abbildung 7.11 zeigt das Gebäude in einer Luftaufnahme und einen Lageplan der Flure und Treppenhäuser.

Da sich die einzelnen Flure im Gebäude optisch nicht voneinander unterscheiden, ist es für Besucher in der Regel zunächst nicht einfach die Struktur des Gebäudes zu erfassen und sich darin zurechtzufinden. Deshalb werden an der Information des Gebäudes meist Pläne mitgegeben, die eine Wegbeschreibung anhand der Flur-Buchstabierung und Treppenhaus-Nummerierung zur gewünschten Abteilung geben.

Werden die Flure nun, wie im Beispiel zuvor, mit Markern ausgestattet, so lässt sich ein Mobiltelefon mit Kamera dazu verwenden, diese Marker zu detektieren und dem Besucher den exakten Standort im Gebäude zu visualisieren. Da der Nutzer hier nicht im Vorbeigehen permanent eine Lokalisierung benötigt, reicht hier eine Verwendung von gewöhnlichen QR-Markern für diese Anwendung aus. Darüber kann z.B. dann eine Gebäudekarte verlinkt sein, die die Position des aktuell gescannten Markers hervorhebt. Durch eine Zieleingabe ist dann eine Routenberechnung möglich, die den Besucher nicht von einem fest definierten Startpunkt, sondern dynamisch von der aktuellen Position zum Ziel führt.

Bei der Verwendung einer Augmented Reality (AR) Anwendung oder

einer entsprechenden AR-Brille, lassen sich die Informationen aus der Wegbeschreibung direkt im Sichtfeld des Nutzers einblenden. Dafür sind die QR-Marker allerdings nicht geeignet, da die Erfassung zu lange dauert. Alternativ können hier wieder AR-Marker verwendet werden, die allerdings aufgrund ihrer stark begrenzten Informationsdichte lediglich eine Identifikationsnummer (ID) und keine vollständige URL speichern können. Hier ist die Verbindung beider Verfahren von Nutzen. Eine Lösung kann dann wie folgt aussehen: Beim Betreten des Gebäudes kann ein QR-Code gescannt werden, der einen Gebäudeplan mit allen IDs der dort verteilten AR-Marker enthält bzw. auf eine URL, von wo diese Informationen heruntergeladen werden können. Durch eine entsprechende Anwendung im Hintergrund können nun die IDs der AR-Marker gelesen und gegen die Position im Gebäude ausgetauscht werden. Im Bild der AR-Brille kann an der Position der AR-Marker entweder die aktuelle Position im Gebäude oder die Richtung zum Ziel eingeblendet werden.

Durch eine einheitliche Verwendung wäre die Anwendung nicht auf ein bestimmtes Gebäude beschränkt, sondern könnte universell für alle Gebäude verwendet werden, die dieses System nutzen. Damit wäre eine allgemeine Anwendung in der Lage die Weginformationen für öffentliche Gebäude zu generieren und anzuzeigen, sodass nicht für jedes Gebäude eine spezielle Anwendung entwickelt und vom Benutzer geladen werden müsste.

## 7.4 Automotive-Anwendungen

Nach den Anwendungen in der mobilen Robotik soll nun der Einsatz von Sensoren und Umwelterfassungssystemen im Bereich der Fahrerassistenz bis hin zu autonomen Fahrzeugen näher betrachtet werden. Viele Aspekte der mobilen Robotik lassen sich zwar in den Bereich „Automotive“ übernehmen, jedoch liegt hier aufgrund des äußerst wichtigen Sicherheitsaspekts, der Fokus oftmals näher auf der zuverlässigen Erfassung und Beurteilung von Szenarien, als es beispielsweise bei der Exploration von anderen Planeten der Fall ist. Dennoch, oder vielleicht auch gerade deswegen lässt sich das in dieser Arbeit entwickelte Konzept zur Objekterkennung und semantischen Umwelterfassung gleichermaßen auf diese Anwendungsfälle anwenden.

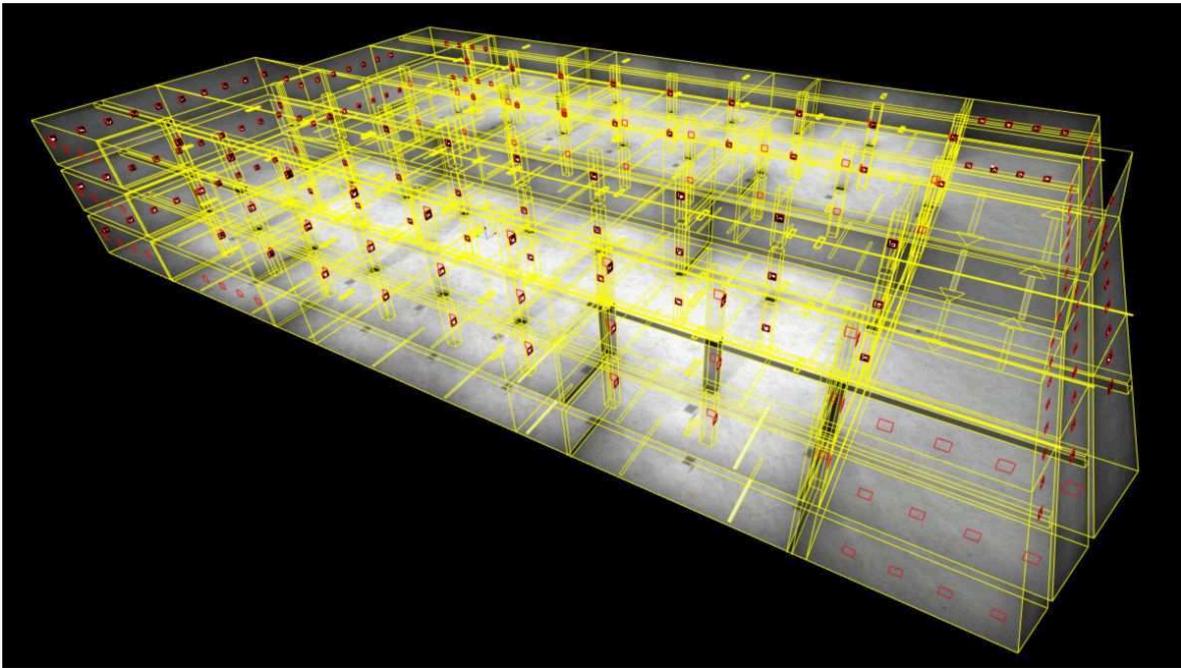


ABBILDUNG 7.12: Modell einer Tiefgarage mit AR-Markern zur Lokalisierung

### 7.4.1 Lokalisierung in Tiefgaragen und Parkhäusern

In Tiefgaragen und Parkhäusern ist eine satellitengestützte Lokalisierung nicht möglich. Insbesondere für die Entwicklung von Fahrzeugfunktionen, wie einem automatisierten Parkassistenten, wo der Fahrer an der Einfahrt des Parkhauses das Fahrzeug verlässt und dieses automatisch einen Parkplatz anfährt und einparkt, ist eine genaue Lokalisierung erforderlich. In Kapitel 5.3 wurde dazu bereits die Verwendung künstlicher Marker untersucht. Diese ermöglichen die Nutzung bereits existierender Parkhäuser und Tiefgaragen für das automatisierte Parken von Fahrzeugen, im Gegensatz zu vollautomatischen Parkhäusern, worin das Fahrzeug als passives Objekt über Aufzüge und Fließbänder an einen Parkplatz transportiert wird. Das Fahrzeug kann einen freien Parkplatz gezielt anfahren und dort einparken. Wird es wieder gebraucht, kann es automatisch zur Ausfahrt fahren wo der Fahrer einsteigen kann.

Für diesen Anwendungsfall wurde die AR-Marker-Detektion und Posenschätzung aus Abschnitt 5.3.2 verwendet, um ein Fahrzeug in einer mehrstöckigen Tiefgarage zu lokalisieren. Dazu wurde in der Simulation ein Fahrzeug mit drei Kameras ausgestattet, die gemeinsam

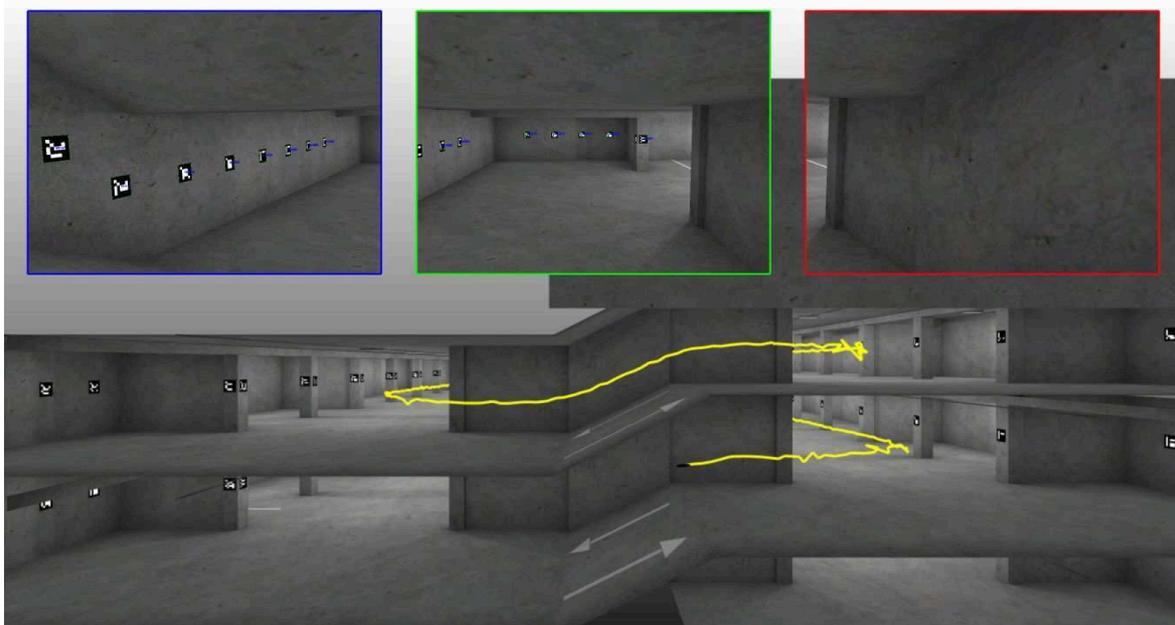


ABBILDUNG 7.13: Zustand zum Zeitpunkt während der Lokalisierung mittels AR-Marker im Tiefgaragenmodell. Die Overlays zeigen die Bilddaten der drei Kameras sowie die daran erkannten Marker (grüne Umrandung). Das Ergebnis der Posenschätzung ist durch die gelbe Linie im Bild gekennzeichnet.

einen horizontalen Sichtbereich von  $150^\circ$  vor dem Fahrzeug abdecken. Verwendet wurden jeweils Kameras mit  $60^\circ$  horizontalem Sichtbereich, wovon die linke und rechte Kamera jeweils um  $45^\circ$  zur Seite gedreht angebracht wurden und die mittlere Kamera in Fahrtrichtung zeigt. Für die folgenden Auswertungen wurden jeweils nur die linke und mittlere Kamera verwendet, die gemeinsam einen horizontalen Bereich von  $105^\circ$  abdecken.

Im Tiefgaragenmodell wurden, verteilt auf fünf Parkebenen, 269 AR-Marker mit individuellen Ids angebracht um eine möglichst lückenlose Beobachtung von Markern zu gewährleisten. Das Fahrzeug wurde entlang eines Pfades (355 Meter Länge) von der Einfahrt bis hinunter auf die unterste Parkebene in einen Parkplatz gesteuert (Abbildung 7.13 zeigt den Zustand während der Testfahrt). Währenddessen wurden sowohl die geschätzten Posen, als auch die, aus der Simulation bekannten, tatsächlichen Posen (ground truth) aufgezeichnet, um die Abweichungen der Lokalisierung über die Zeit auswerten zu können. Die aus der Lokalisierung resultierenden Trajektorien (grün und blau) sind in Abbildung

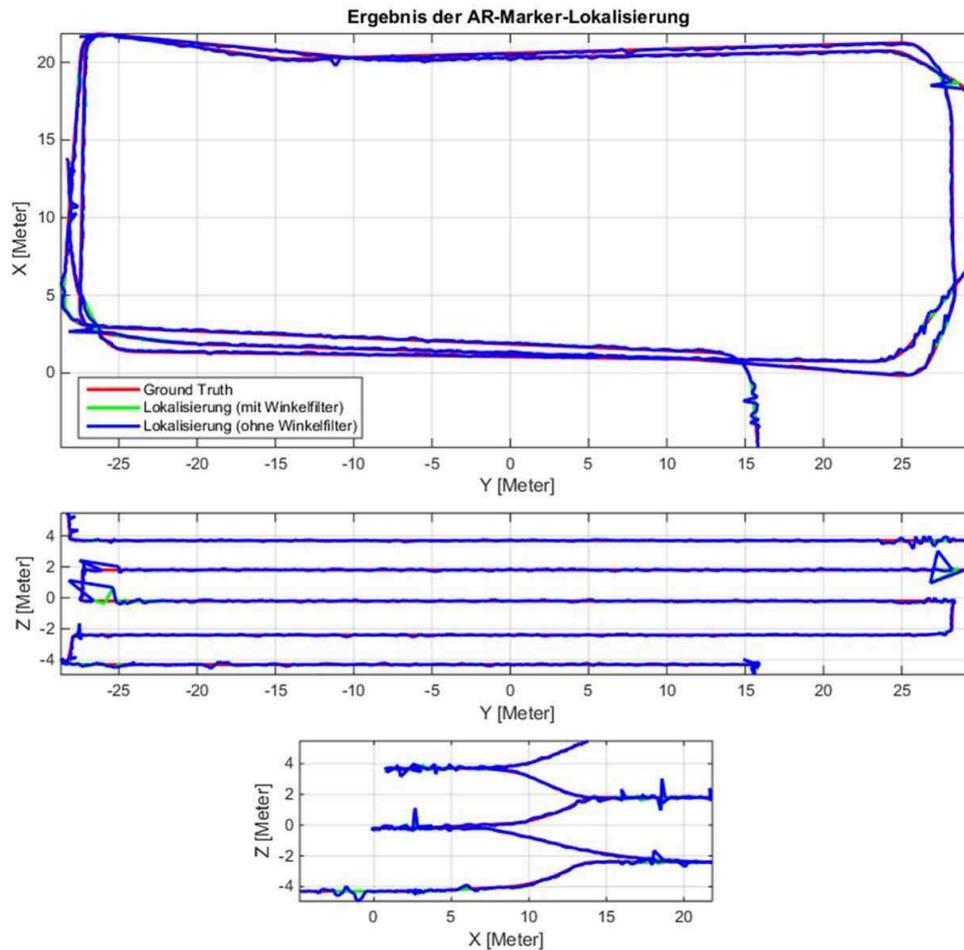


ABBILDUNG 7.14: Ergebnistrajektorie der AR-Marker basierten Lokalisierung

7.14 der Ground-Truth-Trajektorie (rot) gegenübergestellt. Es wurde kein zusätzlicher Algorithmus implementiert, der die Posenschätzung aller sichtbaren Marker fusioniert, d.h. die Gesamtpose des Fahrzeugs wurde jeweils aus der Posenschätzung des, gemäß euklidischem Abstand, nächsten Markers zum Kamerasystem übernommen. Für die Posenschätzung aus den einzelnen Markern wurden bereits die Ergebnisse aus Abschnitt 5.3.2 herangezogen, um die Singularitäten zu eliminieren, die bei der Posenschätzung auftreten wenn der Marker nahezu frontal betrachtet wird. Dafür wurden zunächst alle Marker herausgefiltert, deren Betrachtungswinkel eine kleinere Abweichung als  $16^\circ$  zur Normalenrichtung des jeweiligen Markers beträgt<sup>10</sup>. Die Trajektorie der auf diese

<sup>10</sup>Vergleiche hierzu die Auswertung in den Abbildungen 5.9 und 5.10.

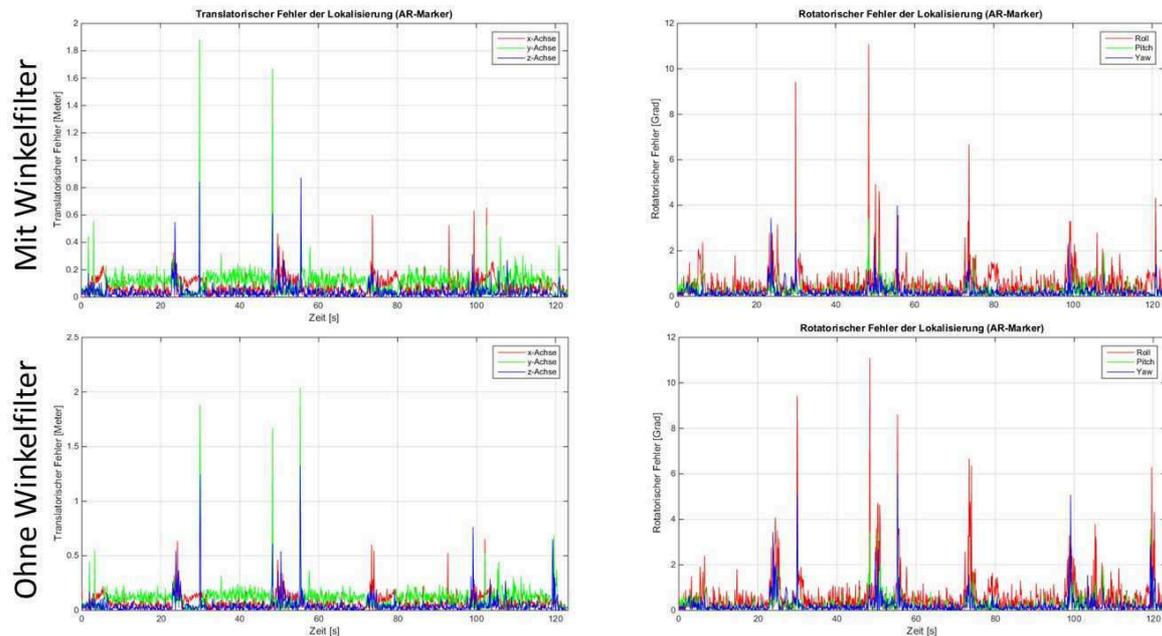


ABBILDUNG 7.15: Auswertung der translatorischen und rotatorischen Abweichungen bei der AR-Marker-basierten Lokalisierung im Parkhausszenario. Gegenüberstellung der ungefilterten Posenschätzung (linke Spalte) und einer Filterung der Marker nach Betrachtungswinkel (rechte Spalte).

Weise gefilterten Messungen ist in Abbildung 7.13 grün dargestellt, die resultierende Trajektorie der ungefilterten Messungen in blau. Durch diese äußerst einfache Vorfilterung konnte die Anzahl der starken Ausreißer während der Lokalisierung an einigen Stellen sichtbar reduziert werden.

Eine Gegenüberstellung der gesamten Testläufe mit und ohne Filterung ist in Abbildung 7.15 abgebildet. Hier ist deutlich zu erkennen, dass die Filterung zur meisten Zeit keinen großen Einfluss auf die Zuverlässigkeit der Lokalisierung hat. Lediglich in Randfällen, also genau dann wenn der maßgeblich an der Lokalisierung beteiligte Marker frontal betrachtet wird, kommt es hier zu vermehrten Ausreißern.

Da hier kein gesonderter Aufwand betrieben wurde, die Posenschätzung aus den einzelnen Markern zu verbessern, sind hier noch erhebliche Verbesserungen in der Genauigkeit und Zuverlässigkeit möglich. Insbesondere sollte hier der Einsatz eines Kalman-Filters zur Reduktion der noch eindeutig identifizierbaren Ausreißer beisteuern und somit den Gesamtfehler der Lokalisierung signifikant verbessern können, da das mittlere Rauschen der Posenschätzung weit unter der Größenordnung

der Ausreißer liegt.

Somit ist dieses Verfahren heute schon real umsetzbar. Viele Fahrzeuge sind bereits serienmäßig mit einer oder mehreren Kameras ausgestattet bzw. lassen sich leicht und kostengünstig nachträglich mit einem Kamerasystem nachrüsten. Für die Übermittlung der Karte, wo zu allen Markern mittels ihrer Id eine Pose zugeordnet wird, ist keine zusätzliche Hardware am Fahrzeug erforderlich, wenn an der Einfahrt zur Tiefgarage ein QR-Marker angebracht wird, der die erforderlichen Informationen enthält und diese optisch an das Fahrzeug übermittelt. Durch die Einschränkung der Markerposen in der Karte auf eine im Vorfeld festgelegte Rasterung mit diskreten Rotationsmöglichkeiten, ließe sich eine Karte von 512 Markern in nur einem QR-Code<sup>11</sup> samt Informations-Header abspeichern. Es wäre auch möglich nur die Marker jeweils eines Parkdecks zu kodieren und bei den Zufahrten zu positionieren, sodass nur die Marker des aktuellen Parkdecks übermittelt werden müssen.

Diese Überlegungen lassen sich ohne weiteres direkt auf weitere Anwendungen überführen, die einer Indoor-Lokalisierung bedürfen<sup>12</sup>.

## 7.4.2 Verbesserte Fahrzeuglokalisierung

Die Aufgabe von (teil-)autonomen Fahrerassistenzsystemen (FAS) ist zum einen die Erhöhung der Sicherheit für die Insassen des Fahrzeugs (z.B. Notbremsen bei akut drohender Kollision, oder das automatische Halten der Fahrspur) und zum anderen, die Unterstützung des Fahrers, indem komplexe Manöver (z.B. Rückwärts einparken parallel zur Fahrbahn) oder monotone Aufgaben (z.B. monotones Anfahren und Abbremsen im sogenannten „zähfließenden Verkehr“ auf der Autobahn) bis zu einem gewissen Punkt übernommen werden. Für all diese Beispiele muss die Umgebung des Fahrzeugs sensorisch erfasst und bis zu einem gewissen Maß semantisch verstanden werden. Das heißt, um etwa eine drohende Kollision zu vermeiden, muss das FAS erkennen, dass entgegenkommender Verkehr auf einer schmalen Landstraße kein Grund für eine Notbremsung darstellt, eine Person, die vor dem Fahrzeug auf die Straße läuft, hingegen sehr wohl. Diese Unterscheidung ist durch eine

---

<sup>11</sup>Der maximale Informationsgehalt eines maximal großen (177×177 Elemente) QR-Codes und geringer Fehlerkorrektur (Fehlerkorrektur-Level „L“ entspricht einer Redundanz von 7%) beträgt 23.648 Bit oder 2.956 Byte.

<sup>12</sup>Vgl. Kapitel 7.3.3.

kontextfreie Auswertung der Messungen einzelner Sensoren nicht möglich. Stattdessen muss eine hoch entwickelte Umgebungserfassung und ggf. Sensorfusion für die Auswertung der Sensordaten und Bewertung der Situation herangezogen werden.

Im Vordergrund dessen steht eine semantische Objekterkennung, die die wesentliche Grundlage für die Erfassung und Einschätzung von Verkehrssituationen darstellt. Nicht nur aus semantischer Sicht besteht ein großer Unterschied zwischen einem Fußgänger und einem Fahrradfahrer im Straßenverkehr. Ein Fußgänger bewegt sich im Vergleich zum Fahrrad meist deutlich langsamer und kann zu jeder Zeit abrupt stehen bleiben oder die Richtung ändern. Diese spontanen Richtungsänderungen treten beim Fahrrad zwar nicht auf, dafür muss hier aber mit, bezogen auf die Fahrtrichtung, lateralen Auslenkungen gerechnet werden. Beide Verkehrsteilnehmer sind aus Sicht von Sicherheitsaspekten mit höchster Aufmerksamkeit zu betrachten, da im Falle eines Unfalls mit einem deutlich größeren Schaden zu rechnen wäre, als bei geschlossenen Kraftfahrzeugen oder rein statischen Objekten wie etwa Bäumen. Aus diesem Grund stellt die Objekterfassung, Klassifizierung und das Tracking dieser Objekte einen Forschungsbereich dar, der zur Zeit einen äußerst starken Fokus erhält. Der aktuelle Stand der Technik auf diesem Gebiet wurde in Kapitel 2.3.1 tiefergehend untersucht. Neue Fahrerassistenzsysteme sind aktuell bereits in der Lage einzelne Aufgaben des Fahrers zu übernehmen und Fahrmanöver selbstständig durchzuführen oder den Fahrer über mögliche Gefahren aufmerksam zu machen. Neben dem semantischen Verständnis von Verkehrssituationen benötigt das Fahrzeug zusätzliche Lokalisierungsinformationen, die zu größten Teilen über GNSS-Empfänger zur Verfügung stehen. Die Spezialfälle, wo diese Informationen nicht zur Verfügung stehen wurden bereits im letzten Abschnitt behandelt. Aber auch wenn satellitengestützte Positionsdaten vorliegen, reicht deren Genauigkeit für manche Funktionen nicht aus, sodass diese Informationen mit der Umgebungserfassung kombiniert werden muss. So kann sich ein Spurhalteassistent nicht allein auf Basis von GNSS-Daten auf der Fahrspur halten und benötigt zusätzlich noch die optische Fahrspurerkennung einer Kamera. Auf vielen Straßen im städtischen Bereich fehlen allerdings auch Straßenmarkierungen, sodass das Assistenzsystem hier nicht zur Verfügung steht. In diesen Fällen ist es hilfreich, wenn das System semantische Objekte aus der Umgebung zur Verbesserung der Lokalisierung heranziehen kann. In diesem Abschnitt



ABBILDUNG 7.16: Baumdetektion zur Verbesserung der Fahrzeuglokalisierung. Links: Verwendung erkannter Baumlandmarken zur Lokalisierung bei fehlender Fahrbahnmarkierung. Rechts: Landmarkendetektion auf Basis von Stereobilddaten.

Bildquelle: [1]

werden zwei Beispiele zur Nutzung von semantischen Umgebungsinformationen zur Verbesserung der Lokalisierung betrachtet. Zunächst werden, wie bereits im Anwendungsfall Wald erfolgreich geschehen, Bäume und baumähnliche Strukturen entlang der Straße zur Ergänzung der satellitengestützten Lokalisierung verwendet. Anschließend werden Straßenschilder als bereits zahlreich vorhandene Landmarken im Straßenverkehr herangezogen.

### Verwendung von Bäumen und säulenförmigen Strukturen

In Städten säumen meist Bäume, Straßenlaternen und weitere säulenförmige Objekte den Straßenrand. Abbildung 7.16 zeigt auf der linken Seite ein Beispiel für eine derartige Situation. Wie bereits im Wald gezeigt wurde, eignen sich diese Art von Landmarken hervorragend zur Lokalisierung, da sie zuverlässig zu erkennen sind und selbst eine eindeutig bestimmbare Position aufweisen (vgl. Kapitel 4.1). Abbildung 7.16 rechts zeigt die Landmarkendetektion auf Basis einer fest im Fahrzeug verbauten Stereokamera und dem in Kapitel 4.2.2 der vorliegenden Arbeit vorgestellten Verfahren. Im Automotive-Bereich, insbesondere bei der

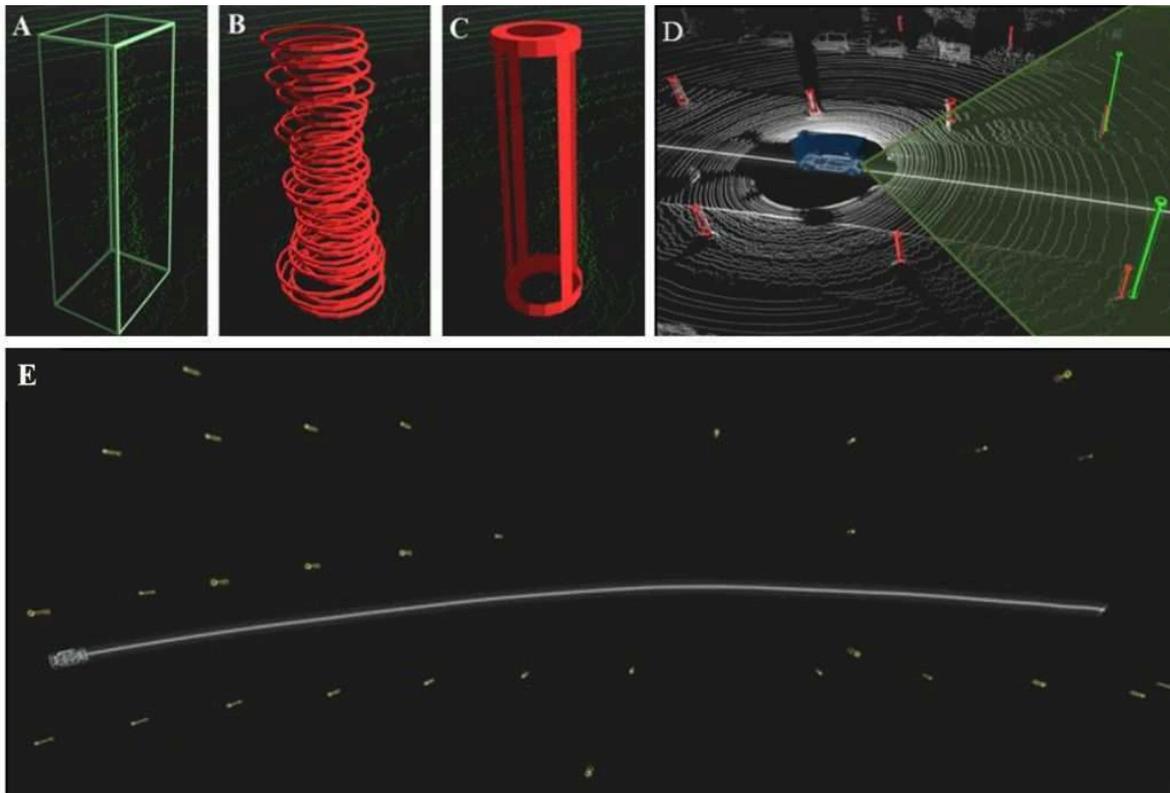


ABBILDUNG 7.17: Detektion von Bäumen und Straßenlaternen zur Verbesserung der Lokalisierung. A: Detektiertes Cluster eines 3D-Laserscans. B: Detektierte Kreise in den einzelnen Schichten des 3D-Laserscanners. C: Kombination der Kreise zu einem gemeinsamen Zylinder. D: Die erfassten Landmarken aus den LiDAR-Daten in rot und aus den Stereodaten in grün. E: Alle beobachteten Landmarken entlang einer Beispielroute.

Bildquelle: [1]

Entwicklung (teil)autonomer Fahrzeuge werden häufig 3D-Multilayer-Laserscanner verwendet, die durch die simultane Aufnahme mehrschichtiger Scanprofile eine zuverlässigere Erkennung von teilweise verdeckten Objekten ermöglichen, als 2D-Laserscanner. In [1] wird eine Erweiterung des kreisbasierten Detektionsalgorithmus aus Kapitel 4.2.2 vorgestellt. Hierzu werden zunächst ebenfalls Cluster in der 3D-Punktwolke eines Scans detektiert und Kreise der einzelnen Scanschichten erkannt. Übereinander liegende Kreise werden anschließend zu einem Zylinder zusammengefasst, der den Baumstamm bzw. die Säulenstruktur des erfassten Objekts wiedergibt. Dieser Ablauf wird in Abbildung 7.17 A-C dargestellt. Erkannte Bäume aus Stereo- und 3D-Laserscannerdaten können



ABBILDUNG 7.18: Bildbasierte Detektion und Entfernungsschätzung von Verkehrsschildern. Links: Ein Verkehrsschild wird Detektiert und über die Zeit verfolgt. Rechts: Eine Auswertung der Verkehrszeichendetektion im Virtuellen Testbed. Bilder: A. Atanasyan, MMI

anschließend getrackt und entsprechend zur Verbesserung der Lokalisierung auf der Straße verwendet werden (Abbildung 7.17 D). Abbildung 7.17 E zeigt die erfassten Landmarken entlang einer beispielhaften Route aus der Top-down-Ansicht.

In [1] wird gezeigt, dass die Verwendung von Bäumen und baumähnlichen Strukturen zur Lokalisierung im Straßenverkehr eine Positionierungsgenauigkeit von unter  $0,5m$  und eine Rotationsgenauigkeit von unter  $1^\circ$  erlaubt. Als Lokisierungsalgorithmus wurde hier wie in Kapitel 5.4 ein MCL-Verfahren mit Partikelfilter verwendet.

### Verwendung von Straßenschildern

Die Verkehrszeichenerkennung ist eine spezielle Ausprägung der semantischen Umwelterfassung und ist vergleichbar mit der Markerdetektion aus Abschnitt 4.2.4 in dieser Arbeit. Das Prinzip ist in beiden Fällen gleich: In einer Sequenz von Bildern soll permanent nach speziellen Objekten gesucht werden, die, nach erfolgter Detektion, klassifiziert und somit semantisch ausgewertet werden. Das jeweilige FAS kann dann entsprechend darauf reagieren, und im Fall einer Geschwindigkeitsbegrenzung einen „Speedlimiter“ setzen, wenn ein Vorschriftszeichen erkannt wurde.

Nach erfolgter Detektion und Klassifizierung endet die Aufgabe der Fahrerassistenzsysteme zur Verkehrsschilderkennung und die erfassten

Objekte werden nicht weiter verwendet und daher verworfen. Im Kontext der in dieser Arbeit behandelten semantischen Umweltmodellierung und unter der Berücksichtigung der zur Zeit akut erforschten Themen des „Internet-of-Things“ und der „Vehicle-to-vehicle Communication“ bzw. der „Vehicular communication systems“ sollten die erfassten Informationen gespeichert und für weitere Anwendungen genutzt und auch an andere Verkehrsteilnehmer weiter gegeben werden.

Für die Lokalisierung in dicht besiedelten Städten mit hohen Gebäuden und dadurch hohen Abweichungen bei der Lokalisierung durch GNSS, können die erkannten Verkehrsschilder auch zur fahrspurgenauen Lokalisierung verwendet werden. Dazu müssen sie nach erfolgreicher Detektion in eine Karte abgespeichert werden. Hierzu eignet sich z.B. eine bereits bestehende digitale Straßenkarte, wie etwa „Open Street Maps“ (OSM), die bereits in vielen Systemen zur Navigation verwendet wird. Durch die offenen Schnittstellen lassen sich zusätzliche, georeferenzierte Informationen in der Karte ablegen und stehen anschließend online für weitere Anwender zur Verfügung.

Ähnlich zur markerbasierten Lokalisierung (behandelt in Abschnitt 5.3) kann die Entfernung der Schilder, anhand der Größe der Abbildung im Bildraum einer kalibrierten, monooptischen Kamera, geschätzt werden, sofern die tatsächliche Größe des Schildes bekannt ist. In Deutschland legt die allgemeine Verwaltungsvorschrift zur Straßenverkehrs-Ordnung (VwV-StVO) im Abschnitt zu den Paragraphen 39-43 der StVO die Größe der zu verwendenden Verkehrszeichen in drei geschwindigkeitsabhängigen Klassen genau fest (siehe 7.1). Daher kann hier von einer einheitlichen, wenn auch nicht vollständig von Ausnahmen ausgenommenen, Regelung ausgegangen werden.

Da die Ausrichtung der Schilder zur Fahrbahn nicht geregelt wird,

Verkehrszeichen	Größe 1 (70%)	Größe 2 (100%)	Größe 3 (125 bzw. 140%)
Ronde (∅)	420	600	750 (125%)
Dreieck (Seitenlänge)	630	900	1260 (140%)
Quadrat (Seitenlänge)	420	600	840 (140%)
Rechteck (HxB)	630 x 420	900 x 600	1260 x 840 (140%)

TABELLE 7.1: Vorgegebene Größen von Verkehrsschildern in Deutschland nach VwV-StVO (Maße in mm)

und auch nicht grundsätzlich eingehalten werden kann, ist eine Schätzung der eigenen Orientierung anhand der Schilder, im Gegensatz zur markerbasierten Lokalisierung, nicht möglich. Stattdessen kann der Straßenverlauf anhand der Odometriedaten des Fahrzeugs während Detektion und Tracking der jeweiligen Schilder verwendet, oder die Detektionen mehrerer Schilder kombiniert werden. Abbildung 7.18 zeigt die bildbasierte Verkehrszeichendetektion mit einer größenabhängigen Entfernungsschätzung. Die grüne Linie zeigt den getrackten Verlauf der jeweiligen Schildpositionen (grüne Kreuze) über die Zeit.

Durch die Bereitstellung der erkannten Objekte und semantischen Informationen in einer öffentlich zugänglichen Karte oder Datenbank, ist der Weg von einer lokalen hin zu einer gemeinschaftlich nutzbaren Lösung geschaffen. Es verbleibt hier, wie bei allen Verbundlösungen, die Frage der universalen Schnittstelle, die im Verbund der Anwender (hier: der gesamten Automobilindustrie) getroffen werden muss.

## Kapitel 8

# Zusammenfassung und Ausblick

Mobile Roboter sollen den Menschen in seiner Arbeit unterstützen oder, bei besonders riskanten Tätigkeiten, sogar ersetzen. Dazu müssen sie in der Lage sein, ihre Umwelt zu erfassen und sich darin zu lokalisieren, um anschließend auch in Interaktion mit ihr zu treten. Zur Umgebungserfassung existieren mittlerweile zahlreiche, unterschiedliche Sensoren und ebenfalls zahlreiche Verfahren zur Verarbeitung dieser sensorspezifischen Daten. In der mobilen Robotik werden aktuell noch verstärkt Umweltmodelle mit geringem Abstraktionsgrad verwendet, wie Punktwolken oder Occupancy Grids. Durch Klassifikation werden diese Daten als Grundlage zur Entscheidungsfindung und Handlungsplanung aufbereitet. Mit zunehmender Komplexität der Aufgaben reicht diese Art der Umgebungsrepräsentation jedoch nicht aus.

In der vorliegenden Dissertation wird zu diesem Zweck ein anwendungsübergreifendes Konzept zur modellbasierten Umgebungserfassung erarbeitet und umgesetzt. In diesem Konzept wird die Abhängigkeit zu einzelnen Sensoren mittels Abstraktion durch Sensordatenverarbeitung aufgelöst. Darauf aufbauend erlaubt die Erzeugung anwendungsübergreifender Modellbeschreibungen einen kognitiven Abstraktionsprozess, der die Detektion und Attribuierung von Objekten als Grundlage für eine gegenständliche Abbildung der Umwelt ermöglicht. Das modulare Lokalisierungsframework, das in der vorliegenden Arbeit vorgestellt wird, erlaubt die einheitliche Nutzung der Detektoren in verschiedensten Anwendungsszenarien über einen abstrakten Landmarkenbegriff. Dabei bleibt das Lokalisierungsverfahren selbst unabhängig vom mobilen System, der Anwendungsdomäne und den konkreten Sensoren.

Eingebettet in ein 3D-Simulationssystem, werden die Ergebnisse der Umgebungserfassung mit der Lokalisierung in einem Simulationsmodell verknüpft und erlauben auf diese Weise eine simulationsgestützte Modellgenerierung. In der vorliegenden Arbeit wurde gezeigt, dass das generierte, semantische Modell im Simulationssystem durch weitere Daten ergänzt, erweitert und somit auf vielfältige Weise, in unterschiedlichen, anspruchsvollen Anwendungen, verwendet werden kann, die weit über den Einsatz des mobilen Systems selbst hinausgehen (z.B. automatisierte Waldbestandsinventur).

Konkrete Erweiterungen dieser Arbeit sind über den modularen Aufbau des vorgestellten Frameworks zur Umgebungserfassung und Lokalisierung sehr einfach zu realisieren. So lassen sich neue Landmarkentypen und -detektoren oder weitere Lokalisierungsverfahren entwickeln und über den abstrakten Landmarkenbegriff als universelle Schnittstelle mit wenig Aufwand integrieren. Neue Sensoren können durch die sensordatenverarbeitende Abstraktionsschicht ebenfalls leicht in das Framework aufgenommen werden.

Durch die simulationsgestützte Umsetzung des in dieser Arbeit vorgestellten Konzepts werden alle erkannten Objekte der Umgebungserfassung nach erfolgter Lokalisierung des mobilen Systems als Digitale Zwillinge im anwendungsübergreifenden, semantischen Modell aufgenommen und integrieren sich, zusammen mit dem Digitalen Zwilling des mobilen Systems selbst, zu einem Virtuellen Testbed (VT). VTs erlauben eine simulationsgestützte Prädiktion und ermöglichen, in Erweiterung der vorliegenden Arbeit, die Realisierung autonomer, mobiler Systeme mit einem Situationsbewusstsein das als Grundlage zur simulationsgestützten Handlungsplanung dient. Die in dieser Arbeit betrachteten Anwendungen liefern die ausschlaggebenden Impulse für die anwendungsübergreifende Umsetzung. Mit steigender Komplexität des Anwendungsgebiets steigen auch die Anforderungen an die Umgebungserfassung und vor allem auch deren Interpretation und Modellierung. Die vorliegende Arbeit erweitert somit die momentan vorherrschende Vorstellung einer Umgebungserfassung und -modellierung um die wesentlichen Aspekte der semantischen Erkennung, Beschreibung und Verwaltung in einem anwendungsübergreifenden Weltmodell. Dazu wird der Begriff des Anwendungsübergreifenden Semantischen Weltmodells (ASW) definiert, und im Anschluss die zu dessen Erzeugung notwendigen Teildisziplinen, also die semantische Umgebungserfassung,

die Lokalisierung und die Umweltmodellierung, identifiziert und im Gesamtzusammenhang näher betrachtet. Das Ergebnis ist mehr als die bloße Summe seiner Teile und bildet durch die Bündelung der Informationen aus verschiedenen Quellen, verknüpft mit den vielseitigen Möglichkeiten der Simulation die Grundlage und Voraussetzung für die Realisierung komplexer Anwendungen in den unterschiedlichsten Domänen. Zukünftige Anwendungen werden, unter der Betrachtung derzeitiger Entwicklungen in der Umgebungserfassung, Vernetzung von Komponenten (Internet of Things) und des maschinellen Lernens, weiter in die Richtung einer gesamtheitlichen Abbildung der Umwelt hinauslaufen. Nur auf diese Weise können Systeme entwickelt werden, die ihre Umwelt tatsächlich verstehen, und auf diese Weise volle Autonomiefunktionen besitzen können.

Aktuell liegt der Fokus noch auf der Entwicklung einzelner, autonomer Systeme, die für sich allein genommen in ihrer Umwelt zurechtkommen sollen (z.B. einzelne autonome Fahrzeuge im Straßenverkehr). Zukünftig wird allerdings auch die Interaktion der Systeme untereinander interessant werden, da sich auf diese Weise mehrere Systeme ein semantisches Weltmodell als Datenbasis teilen und schließlich auch sich selbst in diesem Modell abbilden können. Somit ist die Reaktionsmöglichkeit nicht mehr auf Objekte und Situationen begrenzt, die ein System selbst sensorisch erfasst hat, sondern es kann auch Informationen zugreifen, die aktuell noch nicht sensoruell erfassbar sind, jedoch zum jetzigen Zeitpunkt bereits relevant sein kann (z.B. ein Hindernis nach der nächsten Kurve).

Die vorliegende Arbeit legt den Grundstein für weitere Entwicklungen auf dem Gebiet der kollektiv nutzbaren, anwendungsübergreifenden Weltmodelle und wagt einen Ausblick in die vielfältigen Möglichkeiten, die daraus erwachsen können.



# Literaturverzeichnis

- [1] Mohsen Sefati et al. "Improving Vehicle Localization Using Semantic and Pole-Like Landmarks". In: *IEEE Intelligent Vehicles Symposium (IV2017)*. Redondo Beach, 2017.
- [2] Bjoern Sondermann and Juergen Rossmann. "Semantic environment perception, localization and mapping". In: *Proceedings - AIMS 2015, 3rd International Conference on Artificial Intelligence, Modelling and Simulation*. 2016, pp. 285–290.
- [3] Bjoern Sondermann and Juergen Rossmann. "Semantic Environment Perception and Modeling for Automated SLAM". In: *International Journal of Simulation Systems, Science & Technology (IJSSST)* 17.34 (2016), pp. 5.1–5.8.
- [4] Bjoern Sondermann, Markus Emde, and Juergen Rossmann. "Application-Independent Localization Based on 3D Simulation Technology". In: *International Journal of Materials, Mechanics and Manufacturing* 4.3 (2015), pp. 167–170.
- [5] Bjoern Sondermann, Markus Emde, and Juergen Rossmann. "Improved Mobile Robot Localization Using Semantic World Models". In: *Proceedings of the 13th Symposium on Advanced Space Technologies in Robotics and Automation (ASTRA 2015)*. 2015.
- [6] Bjoern Sondermann and Juergen Rossmann. "Simultaneous Localization and Mapping based on Semantic World Modelling". In: *EMS 2014 - UKSim-AMSS 8th European Modelling Symposium on Computer Modelling and Simulation*, ed. by M Vannucci D. Al-Dabass V. Colla and A Pantelous. IEEE, Oct. 2014, pp. 163–168.
- [7] Markus Emde, Bjoern Sondermann, and Juergen Rossmann. "A Self-Contained Localization Unit for Terrestrial Applications and Its Use in Space Environments". In: *The 12th International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS)*. 2014.

- [8] Nico Hempe, Juergen Rossmann, and Bjoern Sondermann. "Generation and Rendering of Interactive Ground Vegetation for Real-Time Testing and Validation of Computer Vision Algorithms". en. In: *Electronic Letters on Computer Vision and Image Analysis* 12.2 (Dec. 2013), pp. 40–53.
- [9] Bjoern Sondermann et al. "Simulation-based Engineering with Hybrid Testbeds". In: *David Al-Dabass, Alessandra Orsoni, Zheng Xie (Eds.): Proceedings of the UKSim-AMSS 7th European Modelling Symposium on Computer Modelling and Simulation (EMS 2013)*. 2013, pp. 442–447.
- [10] Juergen Rossmann et al. "Advanced Mobile Robot Engineering with Virtual Testbeds". In: *Proceedings for the Conference of ROBOTIK 2012, 7th German Conference on Robotics, May 21-22, Munich*. VDE Verlag GmbH Berlin, 2012, pp. 331–336.
- [11] Petra Krahwinkler, Juergen Rossmann, and Bjoern Sondermann. "Support Vector Machine Based Decision Tree for Very High Resolution Multispectral Forest Mapping". In: *IGARSS 2011 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2011.
- [12] Juergen Rossmann et al. "Advanced Self-Localization and Navigation for Mobile Robots in Extraterrestrial Environments". In: *Computer Technology and Application* 2.5 (2011), pp. 344–353.
- [13] Juergen Rossmann, Bjoern Sondermann, and Markus Emde. "Virtual Testbeds for Planetary Exploration: The Self-Localization Aspect". In: *11th Symposium on Advanced Space Technologies in Robotics and Automation (ASTRA 2011)*. 2011.
- [14] Juergen Rossmann, Magdalena Springer, and Bjoern Sondermann. "Ein Virtuelles Testbed für Planeten-Landemissionen - Hintergründe und Konzept". In: *10. Paderborner Workshop Augmented & Virtual Reality in der Produktentstehung*. Gausemeier, J. / Grafe, M. / Meyer auf der Heide, F., 2011, pp. 107–118.
- [15] Markus Emde et al. "Advanced Sensor Simulation In Virtual Testbeds: A Cost-Efficient Way to Develop and Verify Space Applications". In: *AIAA SPACE 2011 Conference & Exposition*. SPACE Conferences & Exposition. American Institute of Aeronautics and Astronautics, Sept. 2011, pp. 1–11.

- [16] Juergen Rossmann et al. "Discussion of a Self-Localization and Navigation Unit for Mobile Robots in Extraterrestrial Environments". In: *The 10th International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS 2010)*. 2010, pp. 46–53.
- [17] Donald A. Davenport. *The Industrial Robot*. Rochester: Institute for Applied FORTH Research, 1983.
- [18] Michael Schluse, Linus Atorf, and Juergen Rossmann. "Experimentable digital twins for model-based systems engineering and simulation-based development". In: *11th Annual IEEE International Systems Conference, SysCon 2017 - Proceedings* (2017).
- [19] Oussama Khatib. "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots". In: *The International Journal of Robotics Research* 5.1 (1986), pp. 90–98. arXiv: arXiv:1011.1669v3.
- [20] J. Borenstein and Y. Koren. "The vector field histogram-fast obstacle avoidance for mobile robots". In: *IEEE Transactions on Robotics and Automation* 7.3 (June 1991), pp. 278–288.
- [21] K. Sabe et al. "Obstacle avoidance and path planning for humanoid robots using stereo vision". In: *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04*. 2004 1 (2004), pp. 592–597.
- [22] Marcus Tönnis et al. "Visualization of spatial sensor data in the context of automotive environment perception systems". In: *2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, ISMAR* (2007).
- [23] Mathias Haberjahn and Karsten Kozempel. "Multi level fusion of competitive sensors for automotive environment perception". In: *Information Fusion (FUSION), 2013* (2013), pp. 397–403.
- [24] Karl-Heinz Ludwig and Volker Schmidtchen. *Metalle und Macht: 1000 bis 1600*. Vol. 2. Propylaeen Technikgeschichte, 1992.
- [25] GeoSLAM Ltd. *GeoSLAM ZEB-Revo*. <http://www.geoslam.com/hardware-products/zeb-revo/>. 2017.

- [26] Daniel Scharstein, Richard Szeliski, and Ramin Zabih. "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms". In: *Proceedings IEEE Workshop on Stereo and Multi-Baseline Vision (SMBV 2001)* 47.1 (2002), pp. 131–140.
- [27] Ekbert Hering. *Sensoren in Wissenschaft und Technik*. Vieweg+Teubner Verlag, 2012, p. 706. arXiv: arXiv:1011.1669v3.
- [28] Konrad Reif. *Sensoren im Kraftfahrzeug*. Wiesbaden: Vieweg+Teubner Verlag, 2012.
- [29] Yi Ma et al. *An Invitation to 3-D Vision: From Images to Geometric Models*. SpringerVerlag, Dec. 2003.
- [30] Bjoern Sondermann. "Automatische Ermittlung von Kameraparametern und 3D-Struktur aus mehreren 2D-Bildern". Diplomarbeit. Rheinische Friedrich-Wilhelms-Universität Bonn, 2009.
- [31] Bill Triggs, Hans Burkhardt, and Bernd Neumann. "Autocalibration from Planar Scenes". In: *Lecture Notes in Computer Science (LNCS)* (1998), pp. 89–105.
- [32] Zhengyou Zhang. "A Flexible New Technique for Camera Calibration (Technical Report)". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.11 (2002), pp. 1330–1334. arXiv: arXiv:1011.1669v3.
- [33] Marc Pollefeys et al. "Visual modeling with a hand-held camera". In: *International Journal of Computer Vision (IJCV)* (2004), pp. 207–232.
- [34] M Pollefeys, F Viest, and L Van Gool. "Surviving dominant planes in uncalibrated structure and motion recovery". In: *7th European Conference on Computer Vision (ECCV)* (2002).
- [35] Gary Bradski and Adrian Kaehler. "OpenCV". In: *Dr. Dobb's Journal of Software Tools* (2000).
- [36] Charles C Chang and Kai-Tai Song. "Ultrasonic Sensor Data Integration and its Application to Environment Perception". In: *Journal of Robotic Systems* 13.10 (1996), pp. 663–677.
- [37] Kai-Tai Song and Wen-Hui Tang. "Environment perception for a mobile robot using double ultrasonic sensors and a CCD camera". In: *Industrial Electronics, IEEE Transactions on* 43.3 (1996), pp. 372–379.

- [38] D S Hall. *High definition lidar system*. <https://www.google.ch/patents/US7969558>. 2011.
- [39] Continental AG. *Hi-Res 3D Flash LIDAR-Technologie*. <https://tinyurl.com/yb9c8ga5>. 2017.
- [40] Quanergy Systems Inc. *Quanergy S3*. <http://quanergy.com/s3/>. 2017.
- [41] Sony Europe Limited. *Playstation4 Stereo Camera*. <https://www.sony.de/electronics/playstation-accessories/ps4camera/specifications>. 2017.
- [42] Zhengyou Zhang. "Microsoft kinect sensor and its effect". In: *IEEE Multimedia* 19.2 (2012), pp. 4–10.
- [43] Clemens Amon and Ferdinand Fuhrmann. "Evaluation of the Spatial Resolution Accuracy of the Face Tracking System for Kinect for Windows V1 and V2". In: *6th Congress of Alps-Adria Acoustics Assosiation* October (2014), pp. 9–12.
- [44] Jakob Engel, Vladyslav Usenko, and Daniel Cremers. "A photometrically calibrated benchmark for monocular visual odometry". In: *arXiv preprint arXiv:1607.02555* (2016). arXiv: 1607.02555.
- [45] Lawson L S Wong, Leslie Pack Kaelbling, and Tomas Lozano-Perez. "Data association for semantic world modeling from partial views". In: *Springer Tracts in Advanced Robotics* 114 (2016), pp. 431–448.
- [46] Evan Kaufman, Taeyoung Lee, and Zhuming Ai. "Autonomous exploration by expected information gain from probabilistic occupancy grid mapping". In: *Proceedings of the IEEE International Conference on Simulation, Modeling, and Programming for Autonomous Robots* (2016), pp. 246–251.
- [47] Joseph K Berry. *Beyond mapping: concepts, algorithms, and issues in GIS*. GIS World, Incorporated, 1993.
- [48] Google Inc. *Street View: We can trike wherever you like*. <https://googleblog.blogspot.de/2009/10/street-view-we-can-trike-whenever-you.html>. 2009.
- [49] Trimble Inc. *Trimble MX2*. <http://www.trimble.com/imaging/Trimble-MX2.aspx>. 2017.

- [50] Leica Geosystems. *Leica Pegasus:Backpack*. [http://www.leica-geosystems.de/de/Leica-PegasusBackpack\\_106730.htm](http://www.leica-geosystems.de/de/Leica-PegasusBackpack_106730.htm). 2017.
- [51] Pix4D. *Pix4D*. <https://pix4d.com/>. 2017.
- [52] F Lu and E Milios. "Globally consistent range scan alignment for environment mapping". In: *Autonomous Robots* 4.4 (1997), pp. 333–349. arXiv: arXiv:1011.1669v3.
- [53] E. Olson. "M3RSM: Many-to-many multi-resolution scan matching". In: *Proceedings - IEEE International Conference on Robotics and Automation* 2015-June. June (2015).
- [54] Kurt Konolige et al. "Efficient Sparse Pose Adjustment for 2D Mapping". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*. Taipei, 2010, pp. 22–29.
- [55] Fernando Martín et al. "Two different tools for three-dimensional mapping DE-based scan matching and feature-based loop detection". In: *Robotica* 32.2014 (2015), pp. 19–41.
- [56] Armin Hornung et al. "OctoMap: An efficient probabilistic 3D mapping framework based on octrees". In: *Autonomous Robots* 34.3 (2013).
- [57] Wolfgang Hess et al. "Real-time loop closure in 2D LIDAR SLAM". In: *Proceedings - IEEE International Conference on Robotics and Automation* 2016-June (2016), pp. 1271–1278.
- [58] Microsoft. *Microsoft HoloLens*. <https://www.microsoft.com/microsoft-hololens/de-de>. 2017.
- [59] Richard I Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2003.
- [60] Junichi Nakamura. *Image sensors and signal processing for digital still cameras*. CRC press, 2016.
- [61] Meng Yang et al. "Gabor feature based robust representation and classification for face recognition with Gabor occlusion dictionary". In: *Pattern Recognition* 46.7 (2013), pp. 1865–1878.
- [62] Ming Liang et al. "Traffic sign detection by ROI extraction and histogram features-based recognition". In: *Proceedings of the International Joint Conference on Neural Networks* (2013).

- [63] Ali Ismail Awad and Mahmoud Hassaballah, eds. *Image Feature Detectors and Descriptors*. Vol. 630. Springer International Publishing, 2016, pp. 203–226.
- [64] J Canny. “A computational approach to edge detection.” In: *IEEE transactions on pattern analysis and machine intelligence* 8.6 (1986), pp. 679–698.
- [65] C Harris and M Stephens. “A combined corner and edge detector”. In: *Fourth Alvey Vision Conference* (1988).
- [66] Edward Rosten and Tom Drummond. “Machine learning for high-speed corner detection”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 3951 LNCS (2006), pp. 430–443.
- [67] Bauckhage C Schmid C. Mohr R. “Evaluation of Interest Point Detectors”. In: *International Journal of Computer Vision* 37.2 (2000), pp. 151–172.
- [68] David G. Lowe. “Distinctive Image Features from Scale-Invariant Keypoints”. In: *International Journal of Computer Vision* 60.2 (Nov. 2004), pp. 91–110.
- [69] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. “SURF: Speeded up robust features”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 3951 LNCS (2006), pp. 404–417.
- [70] Tony Lindeberg. “Scale-space theory: a basic tool for analyzing structures at different scales”. In: *Journal of Applied Statistics* 21.1 (1994), pp. 225–270.
- [71] Sn Sinha et al. “GPU-based video feature tracking and matching”. In: *EDGE, Workshop on Edge ...* 012.May (2006), pp. 1–15.
- [72] David Nistér, Oleg Naroditsky, and James Bergen. “Visual odometry for ground vehicle applications”. In: *Journal of Field Robotics* 23.1 (Jan. 2006), pp. 3–20.
- [73] Marius Muja and David G. Lowe. “Fast matching of binary features”. In: *Proceedings of the 2012 9th Conference on Computer and Robot Vision, CRV 2012* (2012), pp. 404–410.

- [74] David Nister and Henrik Stewenius. "Scalable recognition with a vocabulary tree". In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2* (2006), pp. 2161–2168.
- [75] Raul Mur-Artal and Juan D. Tardos. "Probabilistic Semi-Dense Mapping from Highly Accurate Feature-Based Monocular SLAM". In: *Robotics: Science and Systems* November (2015), p. 6.
- [76] Bernd Helge Sch, Martin Proetzsch, and Karsten Berns. "Stereo-Vision-Based Obstacle Avoidance in Rough Outdoor Terrain Challenges in Rough Terrain". In: *Most* (2005), pp. 1–9.
- [77] Arturo Rankin, Andres Huertas, and Larry Matthies. "Evaluation of stereo vision obstacle detection algorithms for off-road autonomous navigation". In: *AUVSI Symposium on Unmanned Systems* <http://hdl.818> (2005).
- [78] Pedro Santana et al. "Cross-country obstacle detection: Space-variant resolution and outliers removal". In: *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Sept. 2008), pp. 1836–1841.
- [79] Pedro Santana et al. "Stereo-based all-terrain obstacle detection using visual saliency". In: *J. Field Rob.* 28.2 (2011), pp. 241–263.
- [80] Martin Buehler, Karl Iagnemma, and Sanjiv Singh. *The 2005 DARPA grand challenge: the great robot race*. Vol. 36. Springer Science & Business Media, 2007.
- [81] R. Manduchi et al. "Obstacle Detection and Terrain Classification for Autonomous Off-Road Navigation". In: *Autonomous Robots* 18.1 (Jan. 2005), pp. 81–102.
- [82] Alberto Broggi et al. "The Single Frame Stereo Vision System for Reliable Obstacle Detection Used during the 2005 DARPA Grand Challenge on TerraMax". In: *2006 IEEE Intelligent Transportation Systems Conference* (2006), pp. 745–752.
- [83] P.H. Batavia and S. Singh. "Obstacle detection using adaptive color segmentation and color stereo homography". In: *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation* (Cat. No.01CH37164) 1 (2001), pp. 705–710.

- [84] Radu Danescu, Florin Oniga, and Sergiu Nedevschi. "Particle grid tracking system for stereovision based environment perception". In: *2010 IEEE Intelligent Vehicles Symposium* (June 2010), pp. 987–992.
- [85] Hernan Badino et al. "Free Space Computation Using Stochastic Occupancy Grids and Dynamic". In: *Programming, Proc. International Conf. Computer Vision, Workshop Dynamical Vision*. 2007.
- [86] Yu Xin et al. "A new occupancy grid of the dynamic environment for autonomous vehicles". In: *IEEE Intelligent Vehicles Symposium, Proceedings Iv* (2014), pp. 787–792.
- [87] Saurav Kumar, Daya Gupta, and Sakshi Yadav. "Sensor Fusion of Laser & Stereo Vision Camera for Depth Estimation and Obstacle Avoidance". In: *International Journal of Computer Applications* 1.26 (Feb. 2010), pp. 22–27.
- [88] A Elfes. "Occupancy Grids: A Stochastic Spatial Representation for Active Robot Perception". In: *ArXiv e-prints* (2013). arXiv: 1304.1098 [cs.R0].
- [89] Martin C Martin and Hans P Moravec. "Robot Evidence Grids". In: *H. P., Carnegie Mellon University Tech Report CMU-RI-TR-96-06, 1996, 48 pages*. March (1996), p. 48.
- [90] Y Kakiuchi et al. "Working with movable obstacles using on-line environment perception reconstruction using active sensing and color range sensor". In: *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Oct. 2010), pp. 1696–1701.
- [91] Henry Carrillo et al. "Autonomous Robotic Exploration Using Occupancy Grid Maps and Graph SLAM Based on Shannon and Renyi Entropy". In: *2015 International Conference on Robotics and Automation (ICRA)* (2015), pp. 1–8.
- [92] Patrik Schmuck, Sebastian A. Scherer, and Andreas Zell. "Hybrid Metric-Topological 3D Occupancy Grid Maps for Large-scale Mapping". In: *IFAC-PapersOnLine* 49.15 (2016), pp. 230–235.
- [93] Anh Nguyen and Bac Le. "3D point cloud segmentation: A survey". In: *IEEE Conference on Robotics, Automation and Mechatronics, RAM - Proceedings* (2013), pp. 225–230.
- [94] B Bhanu et al. *Range Data Processing: Representation of Surfaces By Edges*. 1986.

- [95] A. D. Sappa and M. Devy. "Fast range image segmentation by an edge detection strategy". In: *Proceedings of International Conference on 3-D Digital Imaging and Modeling, 3DIM 2001-Janua* (2001), pp. 292–299.
- [96] Jie Chen and Baoquan Chen. "Architectural modeling from sparsely scanned range data". In: *International Journal of Computer Vision* 78.2-3 (2008), pp. 223–236.
- [97] Paul J. Besl and Ramesh C. Jain. "Segmentation Through Variable-Order Surface Fitting." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 10.2 (1988), pp. 167–192.
- [98] G Forlani and C Nardinocchi. "Adaptive filtering of aerial laser scanning data". In: *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* 36.3/W52 (2007), pp. 130–35.
- [99] Sagi Filin. "Surface clustering from airborne laser scanning data". In: *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences* 34 (2001), pp. 119–124.
- [100] George Sithole and George Vosselman. "Filtering of laser altimetry data using a slope adaptive filter". In: *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences* 34.3/W4 (2001), pp. 203–210.
- [101] R. Schnabel, R. Wahl, and R. Klein. "Efficient RANSAC for point-cloud shape detection". In: *Computer Graphics Forum* 26.2 (2007), pp. 214–226. arXiv: arXiv:1507.03351v1.
- [102] T Landes and P Grussenmeyer. "Hough-Transform and Extended Ransac Algorithms for Automatic Detection of 3D Building Roof Planes From Lidar Data". In: *ISPRS Workshop on Laser Scanning 2007 and SilviLaser 2007*. Vol. XXXVI. 2007, pp. 407–412.
- [103] Yangyan Li et al. "GlobFit: Consistently Fitting Primitives by Discovering Global Relations". In: *ACM Trans. Graph.* 30.4 (2011), 52:1–52:12.
- [104] Tongtong Chen et al. "Likelihood-Field-Model-Based Vehicle Pose Estimation with Velodyne". In: *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC 2015-Octob* (2015), pp. 296–302.

- [105] Jonathan R. Schoenberg, Aaron Nathan, and Mark Campbell. "Segmentation of dense range information in complex urban scenes". In: *IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems, IROS 2010 - Conference Proceedings* (2010), pp. 2033–2038.
- [106] Johannes Strom and Edwin Olson. "Graph-based Segmentation for Colored 3D Laser Point Clouds". In: *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2010, pp. 2131–2136.
- [107] Abdul Nurunnabi, David Belton, and Geoff West. "Robust Segmentation in Laser Scanning 3D Point Cloud Data". In: *2012 International Conference on Digital Image Computing Techniques and Applications (DICTA)* (2012), pp. 1–8.
- [108] Matthias Schreier, Volker Willert, and Jürgen Adamy. "Compact Representation of Dynamic Driving Environments for ADAS by Parametric Free Space and Dynamic Object Maps". In: *IEEE Transactions on Intelligent Transportation Systems* 17.2 (2016), pp. 367–384.
- [109] R.Osada et al. "Matching 3D models with Shape distribution". In: *Shape Modeling and Applications, SMI 2001 International Conference*. 2001, pp. 154–166.
- [110] Ahmad Kamal Aijazi, Paul Checchin, and Laurent Trassoudaine. "Segmentation based classification of 3D urban point clouds: A super-voxel based approach with evaluation". In: *Remote Sensing* 5.4 (2013), pp. 1624–1650.
- [111] M. Himmelsbach and T. Luettel. "Real-time object classification in 3D point clouds using point feature histograms". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2009), pp. 994–1000.
- [112] Jiejun Xu et al. "2D/3D sensor exploitation and fusion for enhanced object detection". In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops* (2014), pp. 778–784.
- [113] A Golovinskiy, V G Kim, and T Funkhouser. "Shape-based recognition of 3d point clouds in urban environments". In: *IEEE 12th International Conference on Computer Vision, 2009 Iccv* (2009), pp. 2154–2161.

- [114] A. M. Ramiya, Rama Rao Nidamanuri, and R. Krishnan. "Semantic labelling of Urban point cloud data". In: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives* 40.8 (2014), pp. 907–911.
- [115] Andre Dionisio Rocha et al. "Semantic Model to Perform Plugability of Heterogeneous Smart Devices into Smart City Environment". In: *Service Orientation in Holonic and Multi-Agent Manufacturing : Proceedings of SOHOMA 2016*. Cham: Springer International Publishing, 2017, pp. 327–335.
- [116] Johannes Meyer and Paul Schnitzspan. "A semantic world model for urban search and rescue based on heterogeneous sensors". In: *RoboCup 2010: robot soccer world cup XIV* (2011), pp. 180–193.
- [117] Islem Jebari et al. "Multi-sensor semantic mapping and exploration of indoor environments". In: *2011 IEEE Conference on Technologies for Practical Robot Applications*. IEEE, Apr. 2011, pp. 151–156.
- [118] Silvia Coradeschi and Alessandro Saffiotti. "An introduction to the anchoring problem". In: *Robotics and Autonomous Systems* 43.2-3 (2003), pp. 85–96. arXiv: arXiv:1011.1669v3.
- [119] J. Elfring et al. "Semantic world modeling using probabilistic multiple hypothesis anchoring". In: *Robotics and Autonomous Systems* 61.2 (2013), pp. 95–105.
- [120] D Nistér, O Naroditsky, and James Bergen. "Visual odometry". In: *Computer Vision and Pattern ...* (2004).
- [121] Mark Maimone, Yang Cheng, and Larry Matthies. "Two years of Visual Odometry on the Mars Exploration Rovers". In: *Journal of Field Robotics* 24.3 (Mar. 2007), pp. 169–186.
- [122] Jacek Zienkiewicz, Robert Lukierski, and Andrew Davison. "Dense, Auto-Calibrating Visual Odometry from a Downward-Looking Camera". In: *Proceedings of the British Machine Vision Conference 2013* (2013), pp. 94.1–94.11.
- [123] Jakob Engel, Jurgen Sturm, and Daniel Cremers. "Semi-dense visual odometry for a monocular camera". In: *Proceedings of the IEEE International Conference on Computer Vision* (Dec. 2013), pp. 1449–1456.

- [124] Thomas Schops, Jakob Enge, and Daniel Cremers. "Semi-dense visual odometry for AR on a smartphone". In: *ISMAR 2014 - IEEE International Symposium on Mixed and Augmented Reality - Science and Technology 2014, Proceedings* (2014), pp. 145–150.
- [125] Clark F. Olson et al. "Rover navigation using stereo ego-motion". In: *Robotics and Autonomous Systems* 43 (Feb. 2003), pp. 215–229.
- [126] Andrew Howard. "Real-time stereo visual odometry for autonomous ground vehicles". In: *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Sept. 2008), pp. 3946–3952.
- [127] A Huang et al. "Visual Odometry and Mapping for Autonomous Flight Using an RGB-D Camera". In: *Springer Tracts in Advanced Robotics* 100 (2011), pp. 235–252.
- [128] Zhiwei Zhu et al. "Stereo-based visual odometry method and system". In: *US Patent 7,925,049*. 2011.
- [129] Kurt Konolige, Motilal Agrawal, and J Sola. "Large-scale visual odometry for rough terrain". In: *Robotics Research* (2011).
- [130] Christian Kerl, Jurgen Sturm, and Daniel Cremers. "Dense visual SLAM for RGB-D cameras". In: *IEEE International Conference on Intelligent Robots and Systems* (2013), pp. 2100–2106.
- [131] Jakob Engel, Thomas Schöps, and Daniel Cremers. "LSD-SLAM: Large-Scale Direct monocular SLAM". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 8690 LNCS.PART 2 (2014), pp. 834–849.
- [132] Munir Zaman. "High Precision Relative Localization Using a Single Camera". In: *Proceedings 2007 IEEE International Conference on Robotics and Automation* April (Apr. 2007), pp. 3908–3914.
- [133] Munir Zaman. "High resolution relative localisation using two cameras". In: *Robotics and Autonomous Systems* 55.9 (Sept. 2007), pp. 685–692.
- [134] Paul Besl and Neil McKay. "A Method for Registration of 3-D Shapes". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 14. 2. 1992, pp. 239–256.
- [135] A Segal, D Haehnel, and S Thrun. "Generalized-ICP". In: *Robotics: Science and Systems* 5 (2009), pp. 168–176.

- [136] L.-P. Morency and T. Darrell. "Stereo tracking using ICP and normal flow constraint". In: *Object recognition supported by user interaction for service robots 4* (2002), pp. 367–372.
- [137] Peter Henry et al. "RGB-D Mapping: Using Depth Cameras for Dense 3D Modeling of Indoor Environments". In: *Experimental Robotics SE - 33*. Ed. by Oussama Khatib, Vijay Kumar, and Gaurav Sukhatme. Vol. 79. Springer Tracts in Advanced Robotics. Springer Berlin Heidelberg, 2014, pp. 477–491.
- [138] Yang Chen and Gérard Medioni. "Object modelling by registration of multiple range images". In: *Image and Vision Computing* 10.3 (Apr. 1992), pp. 145–155.
- [139] Roy Anati et al. "Robot Localization Using Soft Object Detection". In: *IEEE International conference Robotics and Automation (2012)*. 2012, pp. 4992–4999.
- [140] M. Tomono and Shin'ichi Yuta. "Object-based localization and mapping using loop constraints and geometric prior knowledge". In: *Robotics and Automation, IEEE International Conference on 1* (2003), pp. 862–867.
- [141] S. Garrido-Jurado et al. "Automatic generation and detection of highly reliable fiducial markers under occlusion". In: *Pattern Recognition* 47.6 (June 2014), pp. 2280–2292.
- [142] Martin Hirzer. "Marker detection for augmented reality applications". In: *Inst. For Computer Graphics and Vision, Graz ...* (2008).
- [143] Michel Bondy et al. "Space Vision Marker System (SVMS)". In: *AIAA SPACE 2007 Conference & Exposition*. Reston, Virginia: American Institute of Aeronautics and Astronautics, Sept. 2007.
- [144] A Ueta et al. "A High-Accuracy 2D Visual Marker for Dexterous Manipulation Robot in Space". In: *Robotics.Estec.Esa.Int* 3.1 (2014), pp. 1–15.
- [145] Shigeru Saito et al. "Indoor marker-based localization using coded seamless pattern for interior decoration". In: *Proceedings - IEEE Virtual Reality* (2007), pp. 67–74.
- [146] Yusuke Nakazato, Masayuki Kanbara, and Naokazu Yokoya. "Localization system for large indoor environments using invisible markers". In: *... of the 2008 ACM symposium on ...* (2008), pp. 295–296.

- [147] Matías Nitsche et al. “WhyCon: An Efficient, Marker-based Localization System”. In: *International Conference on Intelligent Robots and Systems, Workshop Aerial Open Source Robotics Workshop* (2015).
- [148] Feng-yu Zhou et al. “A high precision visual localization sensor and its working methodology for an indoor mobile robot”. In: *Frontiers of Information Technology & Electronic Engineering* 17.4 (2016), pp. 365–374.
- [149] JM O’Kane. “Global localization using odometry”. In: *IEEE International Conference on Robotics and Automation, 2006. (ICRA 2006)*. 2006, pp. 1–9.
- [150] M. Rao, G. Dudek, and S. Whitesides. “Randomized Algorithms for Minimum Distance Localization”. In: *The International Journal of Robotics Research* 26.9 (2007), pp. 917–933.
- [151] M. M. Ullah et al. “Towards robust place recognition for robot localization”. In: *Proceedings - IEEE International Conference on Robotics and Automation* (2008), pp. 530–537.
- [152] Young Ho Choi, Tae Kyeong Lee, and Se Young Oh. “A line feature based SLAM with low grade range sensors using geometric constraints and active exploration for mobile robot”. In: *Autonomous Robots* 24.1 (2008), pp. 13–27. arXiv: thereisnot.
- [153] F. Ferreira et al. “T-SLAM: Registering topological and geometric maps for robot localization”. In: *Lecture Notes in Electrical Engineering* 35 LNEE (2009), pp. 423–438.
- [154] Mohsen Mirkhani et al. “A novel efficient algorithm for mobile robot localization”. In: *Robotics and Autonomous Systems* 61.9 (2013), pp. 920–931.
- [155] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. MIT Press, 2005, p. 647. arXiv: arXiv:1011.1669v3.
- [156] H. Andreasson, A. Treptow, and T. Duckett. “Localization for Mobile Robots using Panoramic Vision, Local Features and Particle Filter”. In: *IEEE International Conference on Robotics and Automation April* (2005), pp. 3348–3353.
- [157] Hashem Tamimi et al. “Localization of mobile robots with omnidirectional vision using Particle Filter and iterative SIFT”. In: *Robotics and Autonomous Systems* 54.9 (2006), pp. 758–765.

- [158] Luca Marchetti, Giorgio Grisetti, and Luca Iocchi. "A comparative analysis of particle filter based localization methods". In: *RoboCup 2006: Robot Soccer World Cup ... Mcl* (2007), pp. 442–449.
- [159] Hyungjin Kim et al. "Robust Vehicle Localization Using Entropy-Weighted Particle Filter-based Data Fusion of Vertical and Road Intensity Information for a Large Scale Urban Area". In: *IEEE Robotics and Automation Letters* 2.3 (2017), pp. 1518–1524.
- [160] Hamzah Ahmad and Toru Namerikawa. "Extended Kalman filter-based mobile robot localization with intermittent measurements". In: *Systems Science & Control Engineering* 1.February 2015 (2013), pp. 113–126.
- [161] Ryan W. Wolcott and Ryan M. Eustice. "Fast LIDAR localization using multiresolution Gaussian mixture maps". In: *2015 IEEE International Conference on Robotics and Automation (ICRA)* (2015), pp. 2814–2821.
- [162] Qiwen Zhang and Gregory Dudek. "Uncertainty Reduction via Heuristic Search Planning on Hybrid Metric / Topological Map". In: *12th Conference on Computer Robot Vision*. 2015.
- [163] Marcus A. Brubaker, Andreas Geiger, and Raquel Urtasun. "Map-based probabilistic visual self-localization". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.4 (2016), pp. 652–665. arXiv: 1511.03240.
- [164] S. Thrun. "Finding landmarks for mobile robot navigation". In: *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*. Vol. 2. IEEE, 1998, pp. 958–963.
- [165] Juergen Rossmann, Petra Krahwinkler, and Christian Schlette. *Navigation of Mobile Robots in Natural Environments: Using Sensor Fusion in Forestry*. 2010.
- [166] Verein Deutscher Ingenieure. *Simulation von Logistik-, Materialfluss- und Produktionssystemen (VDI 3633)*. 2014.
- [167] Herbert Stachowiak. *Allgemeine Modelltheorie*. Vol. formatik-S. Springer Verlag Wien, 1973, p. 494.
- [168] Peter Pin-Shan Chen. "The Entity-Relationship Unified View of Data Model-Toward a". In: *ACM Transactions on Database Systems* 1.1 (1976), pp. 9–36.

- [169] Andreas Nüchter. *3D Robotic Mapping*. Vol. 52. Springer, 2009, p. 201. arXiv: arXiv:1011.1669v3.
- [170] Kichun Jo et al. “Development of Autonomous Car–Part I: distributed system architecture and development process”. In: *IEEE Transactions on Industrial Electronics*. Vol. 61. 12. 2014, pp. 7131–7140.
- [171] Kichun Jo et al. “Development of Autonomous Car Part II: A Case Study on the Implementation of an Autonomous Driving System Based on Distributed Architecture”. In: *IEEE Transactions on Industrial Electronics*. Vol. 62. 8. 2015, pp. 5119–5132.
- [172] Ryan W. Wolcott and Ryan M. Eustice. “Fast LIDAR localization using multiresolution Gaussian mixture maps for autonomous driving”. In: *International Journal of Robotics Research* (2017), pp. 1–28.
- [173] Andreas Nüchter. *3D Robotic Mapping*. Vol. 52. Springer Tracts in Advanced Robotics. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, p. 201. arXiv: arXiv:1011.1669v3.
- [174] J Rossmann et al. “A New Approach to 3D Simulation Technology as Enabling Technology for eROBOTICS”. In: *1st International Simulation Tools Conference & EXPO 2013, SIMEX’2013, September 18-19, Brussels*. Ed. by Jan F M Van Impe and Filip Logist. 2013, pp. 39–46.
- [175] L Atorf, M Schluse, and J Rossmann. “Simulation-based Optimization, Reasoning and Control: The eRobotics Approach Towards Intelligent Robots”. In: *The 12th International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS)*. Canadian Space Agency. 2014, Session 6b: Planning 1, pp.1–8.
- [176] Edward Glaessgen and David Stargel. “The Digital Twin Paradigm for Future NASA and U.S. Air Force Vehicles”. In: *53rd AIAA/ ASME/ ASCE/ AHS/ ASC Structures, Structural Dynamics and Materials Conference/20th AIAA/ASME/AHS Adaptive Structures Conference/14th AIAA*. Reston, Virginia: American Institute of Aeronautics and Astronautics, Apr. 2012.
- [177] Daniel R. Montello. “Spatial Cognition”. In: *International Encyclopedia of Social and Behavioral Sciences* (2001), pp. 14771–14775.

- [178] Xiaozhi Chen et al. "Multi-View 3D Object Detection Network for Autonomous Driving". In: *Cvpr2017* (2016), pp. 1907–1915. arXiv: 1611.07759.
- [179] Arnold Bücken. *Automatische Modellierung von Waldlandschaften für virtuelle Welten und mobile Roboter*. Springer Vieweg, 2013.
- [180] A. Huertas, L. Matthies, and A. Rankin. "Stereo-Based Tree Traversability Analysis for Autonomous Off-Road Navigation". In: *2005 Seventh IEEE Workshops on Applications of Computer Vision (WACV/MOTION'05) - Volume 1*. Vol. 1. IEEE, Jan. 2005, pp. 210–217.
- [181] D.C. Asmar, J.S. Zelek, and S.M. Abdallah. "Tree Trunks as Landmarks for Outdoor Vision SLAM". In: *2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06)*. IEEE, 2006, pp. 196–196.
- [182] Heiko Hirschmüller. "Stereo processing by semiglobal matching and mutual information." In: *IEEE transactions on pattern analysis and machine intelligence* 30.2 (Feb. 2008), pp. 328–41.
- [183] V Gor, R Castano, and R Manduchi. "Autonomous rock detection for Mars terrain". In: *Space* (2001), pp. 1–14.
- [184] Samuel Zuercher, Raphael Schwitter, Pascal Junod, et al. "Moeglichkeiten und Grenzen praxisorientierter Beobachtungsnetze im Forschungsprogramm Wald und Klimawandel". In: (2015).
- [185] Brian Williams et al. "A comparison of loop closing techniques in monocular SLAM". In: *Robotics and Autonomous Systems* 57.12 (2009), pp. 1188–1197.
- [186] Nico Hempe. "Realizing the eRobotics Approach in Semantics-Based Multi-Domain VR Simulation Systems". In: *International Journal of Simulation Systems, Science and Technology (IJSSST)* 15 (2014), pp. 98–108.
- [187] Landaschaftsentwicklung und Forstplanung Nordrhein-Westfalen Landesanstalt für Ökologie. *Hilfstafeln für die Forsteinrichtung*. Recklinghausen: Landesanst. für Ökologie, Landschaftsentwicklung u. Forstplanung Nordrhein-Westfalen, 1989.
- [188] F.T. Purfürst et al. "Hochgenaue Positionsbestimmung von Forstmaschinen". In: *Forst&Technik* (2007).

- 
- [189] Hans Sommer. “Planen und vorbereiten mit BIM und Lean Design Management”. In: *Projektmanagement im Hochbau*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 117–158.
- [190] William McDonough and Michael Braungart. “Cradle to Cradle: Remaking the way we make things”. In: *Chemical and Engineering News* (2002), p. 193.
- [191] SO/TC 184/SC 2. *Robots and robotic devices — Vocabulary*. Tech. rep. International Organization for Standardization (ISO), 2012.



# Eigene Veröffentlichungen

- [1] Mohsen Sefati et al. "Improving Vehicle Localization Using Semantic and Pole-Like Landmarks". In: *IEEE Intelligent Vehicles Symposium (IV2017)*. Redondo Beach, 2017.
- [2] Bjoern Sondermann and Juergen Rossmann. "Semantic environment perception, localization and mapping". In: *Proceedings - AIMS 2015, 3rd International Conference on Artificial Intelligence, Modelling and Simulation*. 2016, pp. 285–290.
- [3] Bjoern Sondermann and Juergen Rossmann. "Semantic Environment Perception and Modeling for Automated SLAM". In: *International Journal of Simulation Systems, Science & Technology (IJSSST)* 17.34 (2016), pp. 5.1–5.8.
- [4] Bjoern Sondermann, Markus Emde, and Juergen Rossmann. "Application-Independent Localization Based on 3D Simulation Technology". In: *International Journal of Materials, Mechanics and Manufacturing* 4.3 (2015), pp. 167–170.
- [5] Bjoern Sondermann, Markus Emde, and Juergen Rossmann. "Improved Mobile Robot Localization Using Semantic World Models". In: *Proceedings of the 13th Symposium on Advanced Space Technologies in Robotics and Automation (ASTRA 2015)*. 2015.
- [6] Bjoern Sondermann and Juergen Rossmann. "Simultaneous Localization and Mapping based on Semantic World Modelling". In: *EMS 2014 - UKSim-AMSS 8th European Modelling Symposium on Computer Modelling and Simulation*, ed. by M Vannucci D. Al-Dabass V. Colla and A Pantelous. IEEE, Oct. 2014, pp. 163–168.
- [7] Markus Emde, Bjoern Sondermann, and Juergen Rossmann. "A Self-Contained Localization Unit for Terrestrial Applications and Its Use in Space Environments". In: *The 12th International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS)*. 2014.

- [8] Nico Hempe, Juergen Rossmann, and Bjoern Sondermann. "Generation and Rendering of Interactive Ground Vegetation for Real-Time Testing and Validation of Computer Vision Algorithms". en. In: *Electronic Letters on Computer Vision and Image Analysis* 12.2 (Dec. 2013), pp. 40–53.
- [9] Bjoern Sondermann et al. "Simulation-based Engineering with Hybrid Testbeds". In: *David Al-Dabass, Alessandra Orsoni, Zheng Xie (Eds.): Proceedings of the UKSim-AMSS 7th European Modelling Symposium on Computer Modelling and Simulation (EMS 2013)*. 2013, pp. 442–447.
- [10] Juergen Rossmann et al. "Advanced Mobile Robot Engineering with Virtual Testbeds". In: *Proceedings for the Conference of ROBOTIK 2012, 7th German Conference on Robotics, May 21-22, Munich*. VDE Verlag GmbH Berlin, 2012, pp. 331–336.
- [11] Petra Krahwinkler, Juergen Rossmann, and Bjoern Sondermann. "Support Vector Machine Based Decision Tree for Very High Resolution Multispectral Forest Mapping". In: *IGARSS 2011 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2011.
- [12] Juergen Rossmann et al. "Advanced Self-Localization and Navigation for Mobile Robots in Extraterrestrial Environments". In: *Computer Technology and Application* 2.5 (2011), pp. 344–353.
- [13] Juergen Rossmann, Bjoern Sondermann, and Markus Emde. "Virtual Testbeds for Planetary Exploration: The Self-Localization Aspect". In: *11th Symposium on Advanced Space Technologies in Robotics and Automation (ASTRA 2011)*. 2011.
- [14] Juergen Rossmann, Magdalena Springer, and Bjoern Sondermann. "Ein Virtuelles Testbed für Planeten-Landemissionen - Hintergründe und Konzept". In: *10. Paderborner Workshop Augmented & Virtual Reality in der Produktentstehung*. Gausemeier, J. / Grafe, M. / Meyer auf der Heide, F., 2011, pp. 107–118.
- [15] Markus Emde et al. "Advanced Sensor Simulation In Virtual Testbeds: A Cost-Efficient Way to Develop and Verify Space Applications". In: *AIAA SPACE 2011 Conference & Exposition*. SPACE Conferences & Exposition. American Institute of Aeronautics and Astronautics, Sept. 2011, pp. 1–11.

- 
- [16] Juergen Rossmann et al. "Discussion of a Self-Localization and Navigation Unit for Mobile Robots in Extraterrestrial Environments". In: *The 10th International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS 2010)*. 2010, pp. 46–53.