



Universität der Bundeswehr München
Fakultät für Luft- und Raumfahrttechnik
Institut für Technik Autonomer Systeme

Konzeption und Aufbau von Autonomen Versuchsfahrzeugen für die Forschung

Dipl.-Ing. Thorsten Lüttel

Vollständiger Abdruck der von der Fakultät für Luft- und Raumfahrttechnik der Universität der Bundeswehr München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

angenommenen Dissertation.

Gutachter: 1. Univ.-Prof. Dr.-Ing. Hans-Joachim Wünsche
2. Univ.-Prof. Mag. Dr. habil. Thomas Pany

Diese Dissertation wurde am 12.06.2023 bei der Universität der Bundeswehr München eingereicht und durch die Fakultät für Luft- und Raumfahrttechnik am 29.09.2023 angenommen. Die mündliche Prüfung fand am 29.09.2023 statt.

Danksagung

Die hier vorliegende Dissertation entstand während meiner Tätigkeit als Wissenschaftlicher Mitarbeiter am Institut für Technik Autonomer Systeme (TAS) an der Fakultät für Luft- und Raumfahrttechnik der Universität der Bundeswehr München. Ich möchte an dieser Stelle die Gelegenheit nutzen, mich bei all den Menschen zu bedanken, die zum Gelingen dieser Arbeit beigetragen haben.

Zuallererst geht der Dank an meinen Doktorvater, Prof. Dr.-Ing. Hans-Joachim Wünsche: Er hat nicht nur für hervorragende Rahmenbedingungen für die Forschung gesorgt, die wissenschaftliche Arbeit des Teams unterstützt und mir die Gelegenheit zur Promotion gegeben, sondern mir auch viele Freiheiten eingeräumt und in vielen Bereichen vertrauensvoll die Verantwortung übertragen. Diese Aufgaben waren oft interessanter und – in eigener Priorisierung – dringender als die Arbeit an der Dissertation, so dass letztere oft hintenanstehen musste... Prof. Mag. Dr. habil. Thomas Pany danke ich für die Erstellung des Zweitgutachtens und seine Anregungen, die zum Gelingen dieser Arbeit beigetragen haben. An Prof. Dr.-Ing. Axel Schulte geht mein Dank für die wiederholte Motivation, die Arbeit zum Abschluss zu bringen, und für die Übernahme des Vorsitzes in meinem Promotionsverfahren.

Ein großer Dank geht an die zahlreichen Kolleginnen und Kollegen für den wissenschaftlichen und nichtwissenschaftlichen Austausch, die gemeinsam verbrachte Zeit, viele bleibende Erinnerungen und Freundschaften.

Mit der ersten TAS-Generation haben wir ab 2006 den Lehrstuhl und das Versuchsfahrzeug MuCAR-3 aufgebaut sowie die ersten ELROB-Teilnahmen gemeistert. Mit Michael Himmelsbach habe ich viele Tage im Auto verbracht, um die Software für die ELROB weiterzuentwickeln, und in den Nächten vor Konferenzdeadlines bei einem Glas Schreibwein an den besten Formulierungen gefeilt. Mit Michael Manz habe ich viele Dienstreisen unternommen, um unsere Algorithmen bei ersten Industriepartnern zu integrieren. Des Weiteren danke ich Falk Hecker und meinem langjährigen Bürokollegen Alois Unterholzner, die im Low-Level-Aufbau des Fahrzeugs involviert waren, sowie den weiteren Kollegen Felix von Hundelshausen, Gerhard Rohe, André Roskopf geb. Müller, Martin Ruß, Sebastian Schneider und Michael Schweitzer. In dieser Zeit sind auch mehrere studentische Arbeiten entstanden, die meine Arbeit unterstützt haben – namentlich geht der Dank an Torsten Bruns und Igor Reichert, deren Arbeiten zur Fahrzeugregelung und zu den Manövern beigetragen haben.

In den Jahren ab 2012 kamen die Kollegen Carsten Fries, Sebastian Bayerl, Hanno Jaspers, Dennis Faßbender, Benjamin Heinrich, Georg Müller, Michael Kusenbach, Jan Kallwies, Felix Ebert, Torsten Engler, Patrick Burger, Benjamin Naujoks, Martin Michaelis, Philipp Berthold, Alexander Frericks, Kai Metzger und Andreas Sticht ins Team. Mit vielen von ihnen habe ich unzählige Wochen in den Versuchsfahrzeugen TULF und StrAsRob auf Standortübungsplätzen sowie auf dem StreetScooter-Testgelände Avantis und auf dem Gillishof verbracht, um dort unseren gesamten Softwarestack weiterzuentwickeln und unsere Autonomie robust demonstrierbar zu machen. In diese Zeit fallen die Erfindung der TAS-Woche sowie weitere erfolgreiche Teilnahmen an den Robotik-Wettbewerben ELROB und euRathlon.

Seit 2019 sind Juan David González González, Bianca Forkel, Andreas Reich, Jakob Dichgans, Peter Mortimer, Lukas Beer, Alexander Bienemann, Anton Backhaus

und Thomas Steinecker im Team hinzugekommen. Zu Beginn der Corona-Pandemie mussten wir gemeinsam ganz TAS schlagartig ins Homeoffice verlagern und haben trotzdem immer Möglichkeiten gefunden, die praktische Erprobung auf den Versuchsfahrzeugen weiterzuführen – und sogar neue Kollegen für das MORE-Projekt ins Team aufzunehmen. Die Tradition der TAS-Wochen und ELROB-Teilnahmen ging hier – mit Corona-Pause – weiter.

Danke für die produktive und spannende Zeit, in der neben der reinen (Team-)Arbeit auch immer Raum für gemeinsame private Aktivitäten war!

Unseren technischen Mitarbeitern Georg Thalhammer, Patrick Neufeld und Rudi Waldherr gilt mein Dank für die Unterstützung bei Aufbau, Weiterentwicklung und Betrieb unserer wachsenden Flotte an Versuchsfahrzeugen und der Forschungsinfrastruktur – ebenso wie den beiden Vorgängern Günther Hofbauer und Anton Pröls. Ohne deren Anregungen und konstruktive Diskussionen über mögliche Umsetzungen, aber vor allem ohne deren Hands-On-Arbeit – sei es in der Werkstatt oder auch vor Ort auf Wettbewerben – wäre die wissenschaftliche Arbeit des gesamten Teams nicht in dieser Qualität möglich. Immer wieder beeindruckend, was man mit einem kleinen Team auf die Beine stellen kann! Herzlichen Dank auch an Thomas Rottmann von LRT 1, der die MORE-Versuchsfahrzeuge maßgeblich mit vorantreibt.

Unserer Teamassistentin Derya Durak und ihrer Vorgängerin Martina Falter – jetzt primär die rechte Hand des Dekans – danke ich für die Unterstützung im Uni-Alltag, die Schnittstelle zur Verwaltung und den ein oder anderen kurzen Dienstweg.

Danke TAS, dass ihr mich durch die Finalisierung dieser Dissertation, die in einer Hau-Ruck-Aktion im Frühjahr 2023 kurzzeitig alle Projektthemen verdrängt hat, begleitet und mich temporär aus dem Tagesgeschäft entlassen habt! Ein besonderer Dank geht an Bianca Forkel und Philipp Berthold für die rasche Durchsicht des finalen Manuskripts dieser Dissertation – ich werde mich gern revanchieren.

Neben dem Institut haben auch externe Projektpartner, die mich teils länger begleitet haben als der ein oder andere direkte Kollege, nennenswert zu den Ergebnissen dieser Dissertation beigetragen. Exemplarisch für das Diehl-Defence-Team möchte ich hier Thomas Kopfstedt und Björn Steurer danken, mit denen wir in den Projekten TULF und StrAsRob zahlreiche gemeinsame Integrations- und Erprobungswochen hatten, um die einzelnen Autonomiefunktionen beider Partner zu einem robusten Gesamtsystem zusammenzubringen. Für das StreetScooter-Team geht mein Dank exemplarisch an Felix Orth und Martin Sommer, mit denen wir viele Wochen in Tag- und Nachtschichten auf Avantis verbracht haben, um eine autonome Paketzustellung zu entwickeln. Darüberhinaus geht der Dank auch an das BAAINBw U6.2, namentlich Johannes Pellenz und Christian Winkens, für die vertrauensvolle Zusammenarbeit in den letzten Jahren und Jahrzehnten!

Ein ganz besonderer Dank geht auch an meine Eltern Erika und Bernward, die mein Leben lang hinter mir stehen und alle Entscheidungen und Wege mittragen – auch wenn ich Nordwestdeutschland zu Gunsten der UniBwM verlassen habe. Abschließend danke ich meiner Familie und meinen Freunden für das Verständnis dafür, dass ab und zu die Dissertation Vorrang hatte, und für den notwendigen Ausgleich neben der Arbeit. Insbesondere die Aktivitäten mit der RMD-Truppe sorgen dafür, dass bei sportlichen Rad- und Ruderurlauben der Kopf wieder frei wird.

Neubiberg, im Herbst 2023

Kurzfassung

Die Anfänge des autonomen Fahrens in den 1980er Jahren waren eher im universitären Umfeld angesiedelt. Hieraus gingen erste große Projekte und Demonstrationen zusammen mit der Industrie hervor. In den 2000er Jahren brachten die DARPA Grand Challenges die Forschung einen großen Schritt nach vorne. Anschließend wurde die Entwicklung von Assistenz- und Autonomiefunktionen von Start-Ups und der Automobilindustrie aufgegriffen. Daraus sind als Produkt schon viele Fahrerassistenzsysteme und erste straßenzugelassene (teil)automatisierte Funktionen entstanden, allerdings bei weitem nicht in dem Tempo, wie anfangs euphorisch angekündigt – das autonome Fahren ist immer noch auf bestimmte Bereiche begrenzt. Mit Einsatzszenarien im unstrukturierten Umfeld, weit abseits von Autobahn und Innenstadt, beschäftigt sich dagegen nur ein kleiner Teil der Forschung, unter anderem das Institut für Technik Autonomer Systeme an der UniBw München.

Ein thematischer Schwerpunkt dieser Arbeit liegt in der Konzeption und dem Aufbau von autonomen Versuchsfahrzeugen, um damit die Grundlagen für die Forschung zu legen. Hierbei werden Fragen nach Basisfahrzeugen, geeigneter Sensorik zur Eigen- und Umweltwahrnehmung genauso behandelt wie flexible Rechnerarchitekturen, Energieversorgung, die zur Fahrzeugansteuerung notwendige Aktorik und schlussendlich ein für den Forschungsbetrieb geeignetes Sicherheitskonzept. Am Institut wurden nach diesen Konzepten Forschungs-Versuchsfahrzeuge aufgebaut, die MuCARs (Munich Cognitive Autonomous Robot Cars). Zusammen mit Partnern aus der Industrie wurden außerdem zwei Bundeswehr-LKW sowie drei Fahrzeuge für den automatisierten Gütertransport umgerüstet.

Neben den Hardware-Komponenten werden zudem grundlegende Software-Module benötigt, um Algorithmen zur maschinellen Wahrnehmung, Verhaltensgenerierung und Bahnplanung entwickeln und testen zu können. Im Rahmen dieser Arbeit wurden dazu die Grundlagen zur Inter-Prozess-Kommunikation gelegt, über die alle Informationen von rohen Kamera-, LiDAR- und Radar-Daten bis hin zu den finalen Stellgrößen für die Fahrzeugansteuerung ausgetauscht werden, und viele der Schnittstellen zur Fahrzeug- und Umweltrepräsentation definiert.

Eine essentielle Grundfunktionalität für ein autonomes Versuchsfahrzeug ist eine präzise Eigenbewegungsschätzung und Lokalisierung, ohne die die Wahrnehmungs- und Navigationsalgorithmen nicht arbeiten können. Für diesen Zweck wird ein Algorithmus vorgestellt, der flexibel alle an den Fahrzeugen vorhandenen Sensoren wie GNSS-Empfänger, IMUs oder Raddrehzahlsensoren, in einem modularen Framework integriert, um auch mit günstigen Sensoren eine robuste und präzise Schätzung erzielen zu können. Eine Redundanz wird hier sowohl durch die gegenseitige Stützung komplementärer Sensoren als auch mehrerer gleichartiger Sensoren ermöglicht.

Als dritter Schwerpunkt wird eine aus Missionen, Aufgaben und Verhaltensweisen bestehende Architektur vorgestellt, die die Ergebnisse der unterschiedlichen, von KollegInnen entworfenen Wahrnehmungs- und Navigationsmodule miteinander verknüpft, um damit auch komplexe Szenarienabfolgen abbilden zu können.

Die vorgestellten Fahrzeuge und Konzepte bewährten sich auch bei internationalen Wettbewerben, wie die Erfolge bei den regelmäßigen Teilnahmen an der ELROB (European Land Robot Trial) zeigen.

Abstract

Design and construction of autonomous test vehicles for research

The beginnings of autonomous driving in the 1980s took place mostly in the university environment. From these emerged the first major projects and demonstrations together with industry. In the 2000s, the DARPA Grand Challenges marked a major step forward, and the development of assistance and autonomy functions was subsequently taken up by start-ups and the automotive industry. As a result, many driver assistance systems and the first road-legal (partially) automated functions have already emerged. Although nowhere near at the pace initially euphorically announced - autonomous driving is still limited to certain areas. In contrast, only a minor part of researchers, including the Institute for Autonomous Systems Technology at UniBw Munich, is dealing with operational scenarios in unstructured environments, far-off from the highway and city traffic.

A major focus of this thesis is the conception and construction of autonomous test vehicles in order to lay the foundations for research. Questions concerning base vehicles, suitable sensors for self- and environmental perception are dealt with as well as flexible computer architectures, energy supply, the actuators necessary for vehicle control and, finally, a safety concept suitable for research work. Based on these concepts, a series of research testing vehicles, the MuCARs (Munich Cognitive Autonomous Robot Cars), were built at the institute. Additionally, together with partners from the industry, two Bundeswehr trucks and three vehicles were also converted for automated freight transport.

In addition to the hardware components, basic software modules are also needed to develop and test algorithms for machine perception, behavior generation and path planning. Within this work, the foundations for inter-process communication were laid, through which all information from raw camera, LiDAR and radar data to the final control variables for vehicle guidance are exchanged, and many of the interfaces for vehicle and environmental representation defined.

A basic functionality for an autonomous test vehicle is accurate ego-motion estimation and localization, which is essential for the perception and navigation algorithms to work properly. For this purpose, an algorithm is presented that flexibly integrates all sensors present on the vehicles, such as GNSS receivers, IMUs, or wheel speed sensors, into a modular framework to be able to achieve robust and precise estimation even with low-cost sensors. Redundancy is enabled here by both mutual support of complementary sensors and multiple sensors of the same type.

As a third focus, an architecture consisting of missions, tasks and behaviors is presented, which links the results of the different perception and navigation modules designed by colleagues in order to be able to represent complex scenario sequences.

The presented vehicles and concepts also proved themselves in international competitions, as the successes in the regular participations in the ELROB (European Land Robot Trial) show.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Definition der Autonomie und Automatisierung	4
1.3	Anforderungen an autonome Systeme	7
1.4	Forschungsfrage	8
1.5	Forschungsbeitrag	9
1.6	Gliederung der weiteren Arbeit	10
2	Stand der Technik	11
2.1	Anfänge des autonomen Fahrens	11
2.2	Große Schritte im autonomen Fahren	15
2.2.1	DARPA Grand Challenges	15
2.2.2	ELROB	20
2.2.3	VisLab Intercontinental Autonomous Challenge	21
2.3	Aktuelle Entwicklung	21
2.3.1	Rechtlicher Rahmen	22
2.3.2	Platooning im Straßengütertransport – zivil und militärisch	23
2.3.3	Innerbetriebliche Logistik	27
2.3.4	Arbeitsmaschinen auf Flughäfen	28
2.3.5	Landwirtschaftliche Nutzung	29
2.3.6	Anwendungen in der Bauwirtschaft und im Bergbau	29
2.3.7	Militärische Nutzung	30
2.3.8	Start-Ups und Tech-Konzerne in den USA	31
2.3.9	Autohersteller und große Zulieferer	34
2.3.10	Urbane Mobilität	37
3	Versuchsfahrzeuge	41
3.1	Gemeinsamkeiten in der Ausstattung	42
3.2	Versuchsfahrzeug VW Touareg – MuCAR-3	44
3.2.1	Energieversorgung und Kühlung	45
3.2.2	Aktorik	47
3.2.3	Sensorik	49
3.2.3.1	Fahrdynamiksensorik	49
3.2.3.2	Lokalisierungssensorik	51
3.2.3.3	Umfeldsensorik	54
3.2.4	Rechnerarchitektur, Kommunikation	57
3.2.5	Sicherheitseinrichtungen	59
3.3	Versuchsfahrzeug VW Tiguan – MuCAR-4	61
3.3.1	Energieversorgung und Kühlung	61
3.3.2	Aktorik	61
3.3.3	Sensorik	63
3.3.3.1	Fahrdynamiksensorik	63
3.3.3.2	Lokalisierungssensorik	64
3.3.3.3	Umfeldsensorik	64

3.3.4	Rechnerarchitektur, Kommunikation	65
3.3.5	Sicherheitseinrichtungen	66
3.4	Versuchsfahrzeug Audi Q8 – MuCAR-5	67
3.4.1	Energieversorgung und Kühlung	68
3.4.2	Aktorik	68
3.4.3	Sensorik	69
3.4.3.1	Fahrdynamiksensorik	69
3.4.3.2	Lokalisierungssensorik	69
3.4.3.3	Umfeldsensorik	69
3.4.4	Rechnerarchitektur, Kommunikation	70
3.4.5	Sicherheitseinrichtungen	70
3.5	Technologieträger Unbemanntes Landfahrzeug	72
3.5.1	Energieversorgung	73
3.5.2	Aktorik	73
3.5.3	Sensorik	73
3.5.3.1	Fahrdynamiksensorik	73
3.5.3.2	Lokalisierungssensorik	76
3.5.3.3	Umfeldsensorik	76
3.5.4	Rechnerarchitektur, Kommunikation	77
3.5.5	Sicherheitseinrichtungen	78
3.6	StrAsRob	79
3.6.1	Sensorik	80
3.6.1.1	Fahrdynamik- und Lokalisierungssensorik	80
3.6.1.2	Umfeldsensorik	80
3.6.2	Rechnerarchitektur, Kommunikation	81
3.6.3	Sicherheitseinrichtungen	81
3.7	StreetScooter B14 Work	82
3.7.1	Energieversorgung	82
3.7.2	Aktorik	83
3.7.3	Sensorik	83
3.7.4	Rechnerarchitektur, Kommunikation	84
3.7.5	Sicherheitseinrichtungen	85
3.8	StreetScooter B16 – MORE Gütertransport	86
3.8.1	Energieversorgung und Kühlung	86
3.8.2	Aktorik	87
3.8.3	Sensorik	88
3.8.3.1	Fahrdynamiksensorik	88
3.8.3.2	Lokalisierungssensorik	88
3.8.3.3	Umfeldsensorik	88
3.8.4	Rechnerarchitektur, Kommunikation	90
3.8.5	Sicherheitseinrichtungen	90
3.9	Fazit	91
4	Softwarearchitektur und Synchronisierung	95
4.1	Softwarearchitektur und Middleware	95
4.1.1	Anforderungen an die Softwarearchitektur	95

4.1.2	KogMo-RTDB	97
4.1.3	ROS	98
4.1.4	Hybrider Betrieb mit mehreren Middlewares	100
4.2	Zeitstempel	101
4.2.1	Rechner-Uhren	102
4.2.2	Zeitstempelvergabe	103
4.2.3	Fazit zur Zeitstempelung	106
4.2.4	Synchronisierung	107
4.3	Kalibrierung	108
5	Eigenbewegungsschätzung und Lokalisierung	111
5.1	Motivation	111
5.2	Koordinatensysteme	117
5.2.1	Erdmodelle	117
5.2.1.1	Beschreibung der Erdfigur durch Referenzellipsoide	117
5.2.1.2	Beschreibung des Schwerefeldes durch das Geoid .	118
5.2.1.3	Verwendete Erdmodelle	119
5.2.2	Raum- und Erd feste Koordinatensysteme	120
5.2.2.1	Inertialkoordinatensystem	120
5.2.2.2	Erdfestes Koordinatensystem WGS 84	120
5.2.2.3	Ebene Koordinatensysteme	122
5.2.3	Mitbewegte Koordinatensysteme	124
5.2.3.1	Fahrzeugkoordinatensysteme	125
5.2.3.2	Sensorkoordinatensysteme	128
5.2.4	Navigationskoordinatensystem	129
5.2.5	Anmerkung	130
5.3	Stand der Technik	130
5.4	Schätzverfahren	133
5.4.1	Kalman-Filter	135
5.4.2	Erweitertes Kalman-Filter	136
5.4.3	Unscented Kalman-Filter	137
5.4.4	Verzögerte Messungen	137
5.5	Modellierung der Fahrzeugbewegung	139
5.5.1	Grundlagen	140
5.5.1.1	Kreisbewegung	140
5.5.1.2	Sekante als Näherung der Kreisbewegung	140
5.5.1.3	Näherung für die Bewegung im dreidimensionalen Raum	142
5.5.1.4	Starrkörper	143
5.5.2	Punktmassenmodell	144
5.5.3	Einspurmodelle	145
5.5.3.1	Kinematisches Einspurmodell	146
5.5.3.2	Dynamisches Einspurmodell	149
5.5.3.3	Überblendung	152
5.5.4	Sonstige Fahrzeugbewegungsmodelle	152
5.5.5	Modelle für weitere Teilbereiche der Fahrzeugdynamik . . .	152

5.6	Filterarchitektur	153
5.6.1	Monolithischer Eigenbewegungsschätzer	153
5.6.2	Dynamisch erzeugter modularer Eigenbewegungsschätzer . .	156
5.7	Modellierung der Sensoren zur Messung der Fahrzeugbewegung . .	158
5.7.1	Allgemeines Messmodell	159
5.7.2	Messung von Einzelradgeschwindigkeiten	159
5.7.3	Weitere Geschwindigkeitsmessungen	160
5.7.4	Messung des Lenkwinkels	161
5.7.5	Messung mit Inertialsensoren	162
5.7.6	Messung mittels GNSS	165
5.8	Bereitgestellte Ergebnisse	167
5.8.1	Global korrigierte Pose	168
5.8.2	Dead-Reckoning-Pose	168
5.9	Umgebungsabhängige Besonderheiten	169
5.10	Ergebnisse Eigenbewegung	170
5.10.1	Monolithischer Eigenbewegungsschätzer	170
5.10.2	Dynamisch erzeugter modularer Eigenbewegungsschätzer . .	171
5.11	Fazit	177
5.11.1	Mögliche Erweiterungen	179
6	Fahrzeugregelung und Manöver	181
6.1	Längsregelung	181
6.1.1	Beschleunigungsvorsteuerung und -regelung	182
6.1.2	Geschwindigkeitsregler	183
6.1.3	Abstandsregler / ACC	183
6.1.4	Anhalteregler	184
6.2	Querregelung	184
6.2.1	Spurregelung	184
6.2.1.1	Kinematisches Einspurmodell	186
6.2.1.2	Dynamisches Einspurmodell	186
6.2.1.3	Überblendung	186
6.2.2	Lenkwinkelregler	186
6.2.3	Lenkratenregler	187
6.3	Verteilung der Regler auf die Ebenen	187
6.3.1	Low-Level	187
6.3.2	High-Level	188
6.4	Manöver	189
7	Missionen, Aufgaben und Verhaltensweisen	193
7.1	Aufgabentypen	195
7.2	Verhaltensweisen	199
7.2.1	Objektfolgen	199
7.2.2	Spurfolgen	200
7.2.3	Abbiegen an Kreuzungen	201
7.2.4	Tentakel-Navigation	203
7.2.5	Wenden	204

7.2.6	Zurücksetzen	204
7.2.7	Lenkwinkelanpassung im Stand	205
7.2.8	Gridplanning-Navigation	205
7.2.9	Anhalten und Not-Halt	205
7.2.10	Fahrstufenwechsel	206
7.3	Wahrnehmungs- und Navigationsmodule	206
7.3.1	Blickrichtungssteuerung	206
7.3.2	Rasterkarten	208
7.3.2.1	Einfache Rasterkarten	209
7.3.2.2	Akkumulierte Rasterkarten	209
7.3.3	Tracking dynamischer Objekte	211
7.3.3.1	LiDAR-basiertes Objekttracking	212
7.3.3.2	Visuelles modellbasiertes Fahrzeugtracking	215
7.3.3.3	Fusioniertes modellbasiertes Fahrzeugtracking	215
7.3.3.4	Objektbasierte Datenfusion im Fahrzeugtracking	216
7.3.3.5	INS-basierte Objekt-Ground-Truth	216
7.3.4	Fahrspuren	217
7.3.4.1	Erkennung und Tracking von Wegen	217
7.3.4.2	Fahrspurgenerierung anhand von Objekten	220
7.3.4.3	GNSS- & kartenbasierte Spurgenerierung	220
7.3.5	Kreuzungen	220
7.3.5.1	Erkennung und Tracking von Kreuzungen	221
7.3.5.2	GNSS- & kartenbasierte Kreuzungsgenerierung	222
7.3.6	Tentakel	223
7.3.7	Gridplanning	226
7.4	Koordination der Einzelverhalten durch hierarchischen Zustandsautomat	226
7.5	Missionen	231
7.5.1	Erstellung von Missionen	231
7.5.2	Überwachung und Kontrolle der Mission	233
7.5.2.1	Übergangsbedingungen für den Wechsel zwischen Aufgaben	233
7.6	Zusammenfassung	235
8	Fazit	239
8.1	Beantwortung der Forschungsfragen	239
A	Posen und Homogene Koordinaten	243
A.1	6-DOF-Pose	243
A.2	Homogene Transformationsmatrizen in 3D	244
A.3	Zusammenhang Pose vs. HTM	245
A.3.1	Berechnung der Euler-Winkel aus einer HTM	246
B	Karten	249
B.1	Definition	249
B.1.1	Menschenlesbare Darstellungsarten	251
B.1.2	Technisch geprägte Darstellungsarten	252

B.2	Analyse Kartenmaterial	255
B.2.1	Vektorbasiskarte 25	255
B.2.2	OpenStreetMap	258
B.2.3	Selbst erstellte Karten	259
B.3	Nutzung Kartenmaterial	260
B.3.1	Kartenimport	260
B.3.2	Globale Routenplanung	261
B.3.2.1	Berücksichtigung von Flächen-Layern	261
B.3.2.2	Lösungsmöglichkeiten für das Shortest Path Problem	262
B.3.2.3	Ausgangsdaten der Routenplanung	263
B.3.3	Hypothesen	263
B.3.3.1	Wegverlauf und Kreuzungen	263
B.3.3.2	Hypothesengenerierung	263
B.3.3.3	Hypothesenverwendung	264
B.3.4	GNSS- & kartenbasierte Spur- und Kreuzungsgenerierung als Rückfallebene	265
B.3.5	Visualisierung	265
C	Auswertung von Missionen	267
C.1	M-ELROB 2010 – Transport Mule-Szenario	267
C.2	Projektabschlussdemonstration	270
	Symbolverzeichnis	275
	Abkürzungen	275
	Konventionen	279
	Indizes	281
	Definitionen	282
	Glossar	283
	Publikationen als Erst- oder Co-Autor	287
	Literaturverzeichnis	293

1 Einleitung

Diese Arbeit beschäftigt sich mit der Automatisierung von Landfahrzeugen für unterschiedliche Einsatzbereiche. Dabei legt sie den Fokus auf Konzeption und Aufbau von Versuchsfahrzeugen für die Forschung im Bereich des autonomen Fahrens und die notwendigen Grundfunktionalitäten, welche zur Entwicklung der Autonomiefunktionen notwendig sind. Ein weiterer Aspekt ist die Entwicklung eines Konzepts, um unterschiedliche Wahrnehmungs- und Navigationsmodule für verschiedene Einsatzszenarien zu kombinieren. Die Gesamteignung der Versuchsfahrzeuge wird anhand realer Wettbewerbsszenarien betrachtet.

Zunächst werden nun Gründe und Ursachen für die Automatisierung von Landfahrzeugen eingeführt, bevor Definitionen für die Autonomie und Automatisierungslevel eingeführt werden. Im Anschluss daran werden Anforderungen an autonome Versuchsfahrzeuge definiert, aus denen sich die Forschungsfragen dieser Arbeit ergeben.

1.1 Motivation

Treiber für die notwendige Automatisierung von Vorgängen lassen sich oft in den „Drei D der Robotik“ finden – *Dull, Dirty, Dangerous*.

Ziviler Straßenverkehr *Dull* bezeichnet hier langweilige, öde Tätigkeiten, wie sie im Straßenverkehr regelmäßig auftreten.

Im Bereich privater PKW wünscht sich der Fahrer gerade in Stausituationen oder im Stop-And-Go-Verkehr durch die Automatisierung eine Entlastung von der eigentlichen Fahraufgabe, um sich risikolos mit Nebentätigkeiten wie der Beschäftigung mit dem Smartphone oder einem Buch widmen zu können. Eine komplette Entfernung des Fahrers aus dem PKW wird derzeit maximal für Parkszenarien oder Hol- und Bringdienste von Mietfahrzeugen angestrebt.

Im zivilen Transportgewerbe wachsen aktuell die Personalprobleme, was sich im Mangel an LKW-Fahrern [Wissenschaftlicher Beirat beim Bundesministerium Verkehr und digitale Infrastruktur, 2020] und Busfahrern [Zeit Online et al., 2023] manifestiert. Aufgrund des hohen Durchschnittsalters der Fahrer steht eine Renteneintrittswelle bevor, und eine vergleichsweise schlechte Bezahlung gekoppelt mit geringer Attraktivität dieser Tätigkeiten (*Dull*) hindert junge Leute daran, diesen Beruf zu ergreifen – dies sind auch Auswirkungen des demographischen Wandels. Der derzeit schon hohe Anteil an Fahrern aus Niedriglohnländern auf europäischen Straßen kann diesen Mangel nur teilweise kompensieren, noch dazu in Kombination mit einem ansteigenden Bedarf im Straßengütertransport. Dies führt bereits jetzt zu volkswirtschaftlichen Auswirkungen wie steigenden Logistikkosten, und diese werden weiter wachsen.

Ein hoher Automatisierungsgrad kann v. a. in der Logistik diesem Problem entgegenwirken. Durch immer leistungsfähigere Fahrerassistenzsysteme, engl. *Advanced*

Driver Assistance Systems (ADAS) wird bereits eine kognitive Entlastung des Fahrers von der eigentlich Fahraufgabe geboten, sodass sich der Fahrer verstärkt Nebenaufgaben widmen kann oder weniger Ruhezeiten benötigt. Bei weiter wachsendem Automatisierungsgrad kann auch der Fahrer zumindest für Routinestrecken komplett aus dem Fahrzeug genommen werden, oder einem bemannten Fahrzeug werden unbemannte, hochautomatisierte Fahrzeuge mittels einer sog. elektronischen Deichsel angehängt.

Unter *Dirty* ist im zivilen Straßenverkehr weniger das dreckige Umfeld zu verstehen, sondern die Emissionen der Fahrzeuge. Automatisierung kann hier zu sparsamerer Fahrweise und damit einer Reduzierung der Emissionen führen. Ein Abstandsregeltempomat, engl. *Adaptive Cruise Control* (ACC) sorgt bspw. für eine gleichmäßigere Fahrweise, eine Berücksichtigung von Karten-, Steigungs- und Verkehrsinformationen ermöglicht eine Optimierung des Geschwindigkeits- und Beschleunigungsprofils – beides führt im Allgemeinen zu geringerem Spritverbrauch und weniger Bremsen- und Reifenabrieb. Der Einsatz von sog. Robo-Taxis, also automatisierten, fahrerlosen PKW oder Kleinbussen, soll des Weiteren die Gesamtanzahl der Fahrzeuge im Straßenverkehr – insbesondere im ruhenden Verkehr – reduzieren, da jedes Robo-Taxi im Vergleich zu privaten PKW eine wesentlich höhere tägliche Auslastung besitzt. Dadurch müssen weniger Fahrzeuge produziert werden, was die Emissionen reduziert. Eine daran angepasste Verkehrsinfrastruktur benötigt weniger versiegelte Flächen, dies käme der Umwelt zugute.

Betrachtet man *Dangerous* im zivilen Straßenverkehr, so geht es hier weniger um den Schutz des Fahrers durch Entfernen desselben aus dem Fahrzeug, sondern insgesamt kann die Gefährdung der Insassen sowie anderer Verkehrsteilnehmer durch steigende Automatisierung und leistungsfähigere ADAS reduziert werden: Der Notbremsassistent verhindert die Kollision mit vorausfahrenden Fahrzeugen idealerweise komplett oder reduziert zumindest die Aufprallgeschwindigkeit und damit die Unfallschwere, Spurhaltesysteme halten das Fahrzeug auch bei abgelenktem Fahrer sicher in der Spur, und eine Müdigkeitserkennung motiviert oder zwingt den Fahrer zu mehr Pausen.

Militärischer Bereich Bei der Bundeswehr besteht derzeit ein generelles Problem bei der Personalgewinnung [Ismar, 2023]. Die Bundeswehr agiert seit vielen Jahren, insbesondere seit der Aussetzung der Wehrpflicht, unter ihrer Soll-Personalstärke, und auch der geplante Personalaufwuchs ist unter den Rahmenbedingungen des demographischen Wandels schwierig umzusetzen [Egleder, 2023].

So besteht – analog zum zivilen Bereich – die Notwendigkeit, eine komplexer werdende Logistik mit weniger Personal sicherzustellen [Gerlach et al., 2020, Kunkel, 2019]. Hier gilt es ebenso, den MKF (Militärkraftfahrer) in langweiligen Situationen (*Dull*) zu entlasten und ihn für Nebenaufgaben verfügbar zu machen. Im Gegensatz zum zivilen Bereich ist das Ziel nicht die gefahrlose Beschäftigung mit dem Smartphone oder Buch, sondern bspw. eine verbesserte Wahrnehmung der Lage oder die Durchführung von Funkverkehr. Möglicherweise kann so in einigen Situationen auf den bisher notwendigen Beifahrer verzichtet werden.

Automatisierte Nachschubtransporte im eigenen Land oder allgemein im friedlichen Umfeld sind durch Übernahme von Technologie aus dem zivilen Bereich in die militärischen Fahrzeuge leichter zu realisieren als Transportaufgaben im Einsatz, wo eine erhöhte Bedrohungslage durch Beschuss oder Minen (*Dangerous*) besteht und eine weniger gut ausgebaute Infrastruktur, z. B. Sandpisten statt Autobahnen, zur Verfügung steht (*Dirty*).

Hier kommt die Gefahr für Mensch und Sache als Motivation für steigende Automatisierung hinzu, um den Schutz des Menschen allein durch die Reduzierung der Anzahl der Soldaten in den Fahrzeugen zu reduzieren. Jedoch wird man in militärischen Konvois nie komplett auf den Soldaten verzichten können, da ein komplett unbemannter Konvoi ansonsten ein zu einfaches Angriffsziel böte – insbesondere wenn Hochwertziele mitgeführt werden. Der Schutz der verbleibenden Soldaten kann jedoch erhöht werden, wenn sich diese bspw. in gepanzerten (Kampf-)Fahrzeugen, also in den Hochwertzielen, befinden. Bei den unbemannten LKW kann auf den passiven Schutz durch schwere gepanzerte Fahrerkabinen verzichtet werden, damit erhöht sich die Nutzlast des Fahrzeugs merklich. Ergänzend zum automatisierten Betrieb ist hier immer auch eine Möglichkeit zur Fernsteuerung, die sog. Teleoperation, zu integrieren. Darüber kann ein menschlicher Operator aus einer zentralen Leitstelle oder aus einem der geschützten Fahrzeuge in Situationen, in denen die Automatisierung keine sichere Aktion ermitteln kann, die Kontrolle übernehmen und die Problemstelle bewältigen.

Für die Transportaufgaben kommen nicht nur LKW im Konvoi zum Einsatz [Pellenz et al., 2021], sondern auch kleinere unbemannte Systeme zur Unterstützung der Soldaten, bspw. Mule-Systeme für die Infanterie, die neben dem Transport von Marschgepäck auch den Verwundetentransport übernehmen können. Diese Fahrzeuge müssen auch offroad, abseits von Straßen und Wegen, operieren können und mit engem Gelände und schlechten Untergründen zurechtkommen.

Ein ähnliches Anforderungsprofil besteht auch für Aufklärungsaufgaben in unbekanntem und gegnerischem Gelände. *Dirty* bezieht sich hier neben dem Untergrund zusätzlich auf eine mögliche Belastung mit aufzuklärenden Kampfstoffen, wofür spezielle CBRNE-Sensorik (Chemisch, Biologisch, Radioaktiv, Nuklear und Explosiv) existiert.

Weitere Bereiche Als weitere Bereiche können Szenarien aus dem zivilen Katastrophenschutz, wie z. B. nach einem Erdbeben, Bergrutsch, Chemie- oder Nuklearunfall, betrachtet werden. Hier ist das Umfeld für Erkundungs- und Bergungsoperationen im Außenbereich sowie im Gebäudeinneren sowohl *Dirty* im Sinne von unstrukturiert als auch *Dangerous* für Fahrzeug und mögliche Bediener und Fahrer. Um das Rettungspersonal keinen unnötigen Risiken auszusetzen, kommen hier oftmals teleoperierte oder in Teilbereichen bereits hochautomatisierte Fahrzeuge zum Einsatz.

In der Landwirtschaft trifft *Dirty* in beiden Ausprägungen zu: Die Fahrzeuge und Maschinen operieren primär in einer Offroad-Umgebung. Durch die Automatisierung kann

die Effizienz in Hinsicht auf Bearbeitungszeit, Dünge- und Pflanzenschutzmitteleinsatz erhöht werden. Daneben wird die physikalische und chemische Bodenbelastung reduziert und damit auch die Emissionen.

Die Motivation aus den unterschiedlichen Domänen und Szenarien zielt also insgesamt auf eine höhere Automatisierung der Fahrzeuge und Systeme ab.

Im Folgenden sollen nun zuerst die Begriffe im Bereich Automatisierung und Autonomie geklärt werden. Anschließend werden die Anforderungen an die automatisierten Fahrzeuge im Allgemeinen sowie für diese Arbeit im Speziellen betrachtet.

1.2 Definition der Autonomie und Automatisierung

Als Autonomie (altgriechisch *autonomía* ‚Eigengesetzlichkeit‘, ‚Selbstständigkeit‘) – dieser Begriff setzt sich aus *autós* ‚selbst‘ und *nómos* ‚Gesetz‘ zusammen – bezeichnet man den Zustand der Selbstbestimmung, Unabhängigkeit bzw. Souveränität, Selbstverwaltung oder Handlungs- bzw. Entscheidungsfreiheit.

Für ein autonomes Fahrzeug würde diese wortwörtliche Auslegung bedeuten, dass es selbst – ohne menschliche Beeinflussung – sein Ziel bestimmt und entsprechend handelt, um dieses Ziel zu erreichen. Von daher ist der Begriff „autonom“ eigentlich falsch gewählt, es sollte hier von Automatisierung gesprochen werden, da das Fahrzeug primär eine vom Menschen definierte Aufgabe erfüllen soll, wobei es die vom Menschen gemachte Vorgaben einhalten muss und die Art der Durchführung für den Menschen verständlich sein muss.

Automotive-Umfeld Im Automotive-Umfeld nutzen unterschiedliche Organisationen verschiedene Begriffe für Stufen der Automatisierung, um zwischen manuellem Fahren und komplett unbemanntem Fahren zu differenzieren.

International gebräuchlich ist die Unterteilung in sechs Stufen, die von der SAE International (vormals als Verband der Automobilingenieure, engl. *Society of Automotive Engineers* bezeichnet) im Standard J3016 [SAE, 2021] definiert wurden und häufig als *L0* bis *L5* abgekürzt werden. Der Automobilzulieferer Visteon [2019] liefert hierzu eine gute Zusammenfassung.

In den unteren drei Stufen überwacht der Fahrer das Fahrzeugumfeld:

Level 0: No Driving Automation / Keine Automation Der Fahrer übernimmt die komplette Fahraufgabe. Dabei können unterstützende und warnende Systeme vorhanden sein. Beispiel: ABS (Anti-Blockier-System, engl. *Anti-lock Braking System*), ESP (Elektronisches Stabilitätsprogramm, engl. *Electronic Stability Control*), LDWS (Spurverlassenswarner, engl. *Lane Departure Warning System*)

Level 1: Driver Assistance / Assistenzsysteme In einzelnen Fahrmodi kann die Lenkung *oder* Beschleunigung/Verzögerung durch ein ADAS übernommen werden, alle anderen Aspekte verbleiben beim Fahrer. Beispiel: LKAS (Spurhalteassistenzsystem, engl. *Lane Keeping Assist System*), Totwinkelassistent

Level 2: Partial Driving Automation / Teilautomatisierung In bestimmten Fahrmodi kann die Lenkung *und* Beschleunigung/Verzögerung des Fahrzeugs durch ADAS übernommen werden, alle anderen Aspekte verbleiben beim Fahrer. Beispiel: ACC, LCA (Spurwechselassistent, engl. *Lane Change Assist*)

In den oberen drei Stufen überwacht das Automatisierungssystem das Fahrzeugumfeld:

Level 3: Conditional Driving Automation / Bedingte Automatisierung In begrenzten Bereichen oder bei bestimmten Fahrmodi fährt das Fahrzeug automatisiert, und der Fahrer kann seine Aufmerksamkeit auf andere Dinge richten. Es wird allerdings erwartet, dass er ordnungsgemäß (d. h. innerhalb einer vorgegebenen Zeitspanne) auf eine Aufforderung zum Eingreifen reagiert. Beispiel: Highway Co-pilot, Traffic Jam Co-pilot, Valet Parking

Level 4: High Driving Automation / Hochautomatisierung In begrenzten Bereichen oder bei bestimmten Fahrmodi fährt das Fahrzeug automatisiert, und der Fahrer kann z. B. schlafen. Es wird nicht erwartet, dass das System unter allen Fahrbedingungen funktioniert. Der Fahrer muss nicht auf eine Aufforderung zum Eingreifen reagieren, und das System muss in der Lage sein, in einen sicheren Zustand überzugehen. Beispiel: Highway Pilot, Traffic Jam Pilot

Level 5: Full Driving Automation / Vollautomatisierung Das automatisierte System übernimmt Aspekte der dynamischen Fahraufgabe unter allen Fahrbahn- und Umgebungsbedingungen, die von einem menschlichen Fahrer bewältigt werden können. Beispiel: Robo-Taxi

Die Bundesanstalt für Straßenwesen (BASt) unterscheidet mit einem vereinfachten Modell die in Abb. 1.1 dargestellten drei Modi, die klar aufzeigen, wer gerade die Fahraufgabe innehat:

Assistierter Modus Nach Aktivierung des Systems unterstützt das System die Fahrer bei der Ausübung der Fahraufgabe. Ungeachtet des Umfangs der Unterstützung müssen die Fahrer permanent das System und das Umfeld überwachen und gegebenenfalls korrigierend eingreifen. [Dies umfasst die SAE-Level (SAE International) 1 und 2.] Beispiel: BMW Driving Assistant Professional, Tesla Autopilot

Automatisierter Modus Während das System das Fahrzeug fährt, können die Fahrer sich fahrfremden Tätigkeiten zuwenden. Voraussetzung ist, dass die Fahrer soweit aufmerksam bleiben, dass sie nach Aufforderung des Systems die Fahraufgabe wieder übernehmen können. Für die Wiederaufnahme des Fahrens bietet das

System den Fahrern hierbei ausreichend Zeit. [Dies entspricht dem SAE-Level 3.] Beispiel: Mercedes Drive Pilot

Autonomer Modus Das System fährt das Fahrzeug. Menschen an Bord sind Passagiere und haben keine fahrbezogenen Aufgaben. [Dies entspricht dem SAE-Level 5 und zeitweise dem Level 4.] Beispiel: autonom fahrende Busse in Monheim (NRW)

(Bundesanstalt für Straßenwesen [2021])

Trotz des Widerspruchs zur wortwörtlichen Auslegung des Begriffs *Autonomie* wird in dieser Arbeit im Folgenden vereinfachend vom *autonomen Fahren* gesprochen – dies entspricht auch der Definition in der Automobilindustrie:

Autonomes Fahren bedeutet das selbständige, zielgerichtete Fahren eines Fahrzeugs im realen Verkehr, ohne Eingriff des Fahrers. Dabei bezieht das Fahrzeug seine Eingangsdaten zunächst aus visuellen Informationsquellen, die auch dem Fahrer zur Verfügung stehen. In Vorstufen des autonomen Fahrens unterstützt die Technik die menschliche Wahrnehmung durch die Bereitstellung von Informationen, die eine sichere Entscheidungsfindung und schnelle Reaktion des Fahrers ermöglichen. Erfolgt die Reaktion des Fahrzeugs allerdings selbständig über Algorithmen und daran geknüpfte Reaktionen des Fahrzeugs ohne die aktive Einwirkung des Fahrers, spricht man vom autonomen Fahren. (Daimler AG [2018])

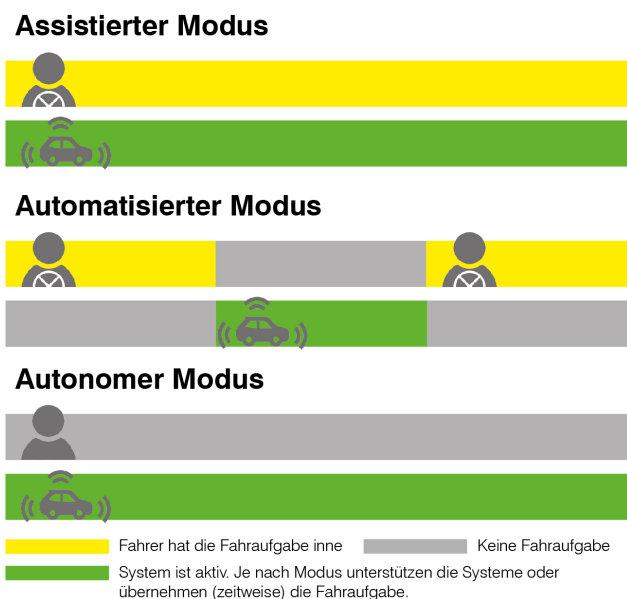


Abbildung 1.1: Die BASt unterteilt in drei Modi. Bildquelle: Bundesanstalt für Straßenwesen [2021]

1.3 Anforderungen an autonome Systeme

Die in Abschnitt 1.1 motivierten Anwendungen erfordern unterschiedliche Grade der Automatisierung. In diesem Abschnitt wird nun der Fokus auf Anwendungen für unbemannte Systeme und damit auf den höchsten Automatisierungsgrad *Autonomes Fahren* gelegt und betrachtet, welche Anforderungen sich daraus an die automatisierten Fahrzeuge ergeben.

Basis für die Nutzung autonomer Fahrfunktionen – insbesondere bei unbemannten Systemen – ist das Vorliegen eines Fahrauftrages, der sog. *Mission*, die definiert, welche *Aufgaben* das Fahrzeug zu erfüllen und welche Zielpunkte es zu erreichen hat.

Auf der funktionalen Seite muss ein autonomes Fahrzeug in der Lage sein, sein Umfeld mithilfe von Sensoren wahrzunehmen, bspw. durch die Verwendung von Kameras oder LiDAR-Sensoren (Light Detection And Ranging). Die Umfeldwahrnehmung besteht einerseits aus einer *positiven Wahrnehmung* von Leitstrukturen wie Straßen und Feldwegen, statischen Objekten wie Bäumen oder Schildern (die auch als sog. Landmarken bezeichnet werden) oder anderen Verkehrsteilnehmern wie Fahrzeugen oder Fußgängern. Für alle gilt, dass ihre Form, Lage im Raum und Dynamik durch Verwendung geeigneter mathematisch-physikalischer Modelle und Algorithmen anhand der Sensordaten bestimmt werden müssen. Andererseits müssen auch kleinere statische Hindernisse oder sogar Löcher erkannt werden, die nicht explizit als „Objekt“ modelliert sind und erkannt werden – dies wird hier als *negative Wahrnehmung* bezeichnet.

Grundlage für eine gute Umfeldwahrnehmung ist eine präzise Eigenwahrnehmung, d. h. eine hochfrequent aktualisierte Kenntnis über die aktuelle Lage und Bewegung des autonomen Fahrzeugs im Raum, primär unter Verwendung der serienmäßigen Fahrdynamiksensoren und einer inertialen Messeinheit. Mithilfe der Eigenbewegungsschätzung können die Daten der Umfoldsensoren räumlich-zeitlich korrigiert werden und die absoluten Geschwindigkeiten anderer Verkehrsteilnehmer bestimmt werden.

Für eine globale Lokalisierung des Fahrzeugs in der Welt dienen Informationen aus einem GNSS (Globales Navigationssatellitensystem, engl. *Global Navigation Satellite System*) sowie Algorithmen, die erkannte Landmarken mit einer Karte abgleichen. Die Lokalisierung ist essentiell, wenn nicht nur mit einem LKAS einer Straße gefolgt wird, sondern darüber hinausgehende Navigationsentscheidungen zu treffen sind.

Basierend auf dem Ziel der aktuellen Aufgabe, der Umfeldwahrnehmung und der Lokalisierung muss die Verhaltensebene des autonomen Fahrzeugs nun entsprechende Zielvorgaben für die Bahnplanung machen und die Wahrnehmungs- sowie Planungs- und Regelungsalgorithmen geeignet parametrieren.

Um die Stellgrößenvorgaben des Reglers in eine Bewegung des Fahrzeugs umsetzen zu können, muss dieses mit geeigneter Aktorik ausgerüstet sein, die eine Ansteuerung aller Fahrfunktionen durch den Rechner ermöglicht.

Im höchsten Automatisierungsgrad *Autonomes Fahren* gelten die auch höchsten Anforderungen an Zuverlässigkeit und funktionale Sicherheit und müssen in allen Entwicklungsschritten mitbetrachtet werden. Für Systeme in Level 3 bzw. Level 4 sind die Anforderungen etwas niedriger, diese werden in einem gemeinsamen White Paper vieler Automobilhersteller und Zulieferer [Wood et al., 2019] vorgestellt. Für Anwendungen wie das ACC, die nur einen geringeren Automatisierungsgrad erfordern, lassen sich die Anforderungen weiter abschwächen, sowohl im Umfang als auch in Bezug auf die Zuverlässigkeit.

Im Rahmen dieser Arbeit werden die Themen Zulassungsfähigkeit, funktionale Sicherheit, Zuverlässigkeit von Hard- und Software sowie rechtliche Fragestellungen explizit nicht betrachtet. Vielmehr geht es um die Erforschung der prinzipiellen technischen Möglichkeiten, den sog. Proof-of-Concept. Die letzten 5 bis 10 % der Zuverlässigkeit, die für den Weg zum Produkt notwendig wären, kann ein Uni-Institut mit 10 bis 15 Mitarbeitern nicht leisten, da diese nochmal ein Vielfaches des Entwicklungsaufwands bedeuten.

1.4 Forschungsfragen

Im Rahmen dieser Arbeit wird untersucht, wie eine Systemarchitektur für ein autonomes Straßen- und Geländefahrzeug gestaltet werden sollte, damit es in definierten militärisch geprägten Szenarien des Gütertransports eine komplexe Mission mit mehreren Teilaufgaben erledigen kann, und dieses experimentell nachgewiesen werden kann. Der Fokus liegt dabei auf der Gestaltung von Versuchsfahrzeugen und notwendigen Grundfunktionalitäten, auf deren Basis weitere Forscher an der Entwicklung von Wahrnehmung- und Navigationsmodulen arbeiten können.

Dabei soll das System ganzheitlich betrachtet werden:

- F-1 Wie kann ein Versuchsfahrzeug mit seiner Sensorik-, Aktorik- und Recherausstattung gestaltet werden?
- F-2 Wie sieht ein für die Entwicklungsphase geeignetes Sicherheitskonzept aus?
- F-3 Welche Hard- und Softwareumgebung ermöglicht einen vereinfachten Entwicklungsprozess für mehrere Wissenschaftler im Fahrzeug und im Labor?
- F-4 Kann die Eigenbewegungsschätzung für autonome Geländefahrzeuge mit kostengünstiger Sensorik erfolgen?
- F-5 Über welche Wahrnehmungs- und Navigationsmodule muss ein autonomes Fahrzeug für diese Szenarien verfügen?
- F-6 Wie können diese modular zu Verhaltensweisen und Aufgabentypen kombiniert werden, um Missionen mit unterschiedlichen Teilaufgaben abzubilden?
- F-7 Kann sich dieses System in einem Wettbewerbsszenario gegenüber anderen Systemen und Teams behaupten?

1.5 Forschungsbeitrag

Diese Arbeit liefert einen Forschungsbeitrag zu den untersten und obersten Ebenen eines autonomen Fahrzeugs, die in einem Gesamtsystem benötigt werden.

Ein Forschungsbeitrag dieser Arbeit liegt in der Konzeption und dem Aufbau von autonomen Versuchsfahrzeugen (Kapitel 3), um damit die Grundlagen für die Forschung anderer Wissenschaftlerinnen und Wissenschaftler zu legen. Hierbei werden Fragen nach Basisfahrzeugen für den Onroad- und Offroad-Bereich und nach geeigneter Sensorik zur Eigen- und Umweltwahrnehmung genauso behandelt wie flexible Rechnerarchitekturen, Aspekte der Energieversorgung, die zur Fahrzeugansteuerung notwendige Aktorik und schlussendlich ein für den Forschungsbetrieb geeignetes Sicherheitskonzept. Hierzu gehören auch die unteren, stark fahrzeugabhängigen Schichten einer kaskadierten Regelung (Abschnitt 6.3).

Neben den Hardware-Komponenten werden zudem grundlegende Software-Module benötigt, um Algorithmen zur maschinellen Wahrnehmung, Verhaltensgenerierung und Bahnplanung entwickeln und testen zu können. Im Rahmen dieser Arbeit wurden dazu die Grundlagen zur Inter-Prozess-Kommunikation gelegt, über die alle Informationen von rohen Kamera-, LiDAR- und Radar-Daten (Radio Detection And Ranging) bis hin zu den finalen Stellgrößen für die Fahrzeugansteuerung ausgetauscht werden, und viele der Schnittstellen zur Fahrzeug- und Umweltrepräsentation definiert (Kapitel 4 und Abschnitte 7.3.3 bis 7.3.5). Damit wurde ermöglicht, dass dieselben Softwarekomponenten einerseits in Fahrzeug und Labor entwickelt und getestet werden können, aber andererseits auch austauschbar bleiben, wenn Neuentwicklungen einzelner Module eine bessere Performance zeigen.

Ein weiterer Forschungsbeitrag liegt im entwickelten modularen Framework zur Eigenbewegungsschätzung und Lokalisierung (Kapitel 5), in dem flexibel unterschiedliche an den Fahrzeugen vorhandene Sensoren wie GNSS-Empfänger, IMUs (Inertiale Messeinheit, engl. *Inertial Measurement Units*) oder Raddrehzahlsensoren integriert werden können. In Abhängigkeit des verwendeten Fahrzeugs werden die Prozess- und Messmodelle sowie ein kombinierter Zustandsvektor für Fahrzeugzustand und Sensorparameter zur Laufzeit zusammengestellt. Dieses Verfahren ermöglicht auch unter suboptimalen GNSS-Bedingungen und auf der Basis von seriennaher Sensorik eine stabile Bewegungsschätzung, die als essentielle Grundfunktionalität für die Wahrnehmungs- und Navigationsalgorithmen benötigt wird.

Der dritte Schwerpunkt gehört zu den oberen Ebenen eines Autonomiesystems. Um die Ergebnisse aus unterschiedlichen, von Kolleginnen und Kollegen entwickelten Wahrnehmungs- und Navigationsmodulen so miteinander zu verknüpfen, dass das autonome Fahrzeug komplexere Szenarienabfolgen, bspw. in realen Wettbewerbsszenarien, bewältigen kann, wurde im Rahmen dieser Arbeit ein hierarchisches Konzept entwickelt und implementiert, welches aus Missionen, Aufgabentypen und Verhaltensweisen besteht (Kapitel 7). Hierzu gehört auch die Koordination des Missionsfortschritts sowie ein hierarchischer Zustandsautomat für die Koordination der Verhaltensweisen, Wahrnehmungs- und Navigationsmodule sowie zur Interaktion mit der Fahrzeugregelung.

1.6 Gliederung der weiteren Arbeit

Das folgende Kapitel 2 gibt einen breiten Überblick über die Entwicklungen im autonomen und automatisierten Fahren, die dort verwendeten Ansätze und beteiligte Forschergruppen und Firmen.

Kapitel 3 beschäftigt sich mit der Konzeption und dem Aufbau von autonomen Versuchsfahrzeugen für die Forschung, der Hardware-Seite des ersten großen Schwerpunkts dieser Arbeit, den die Grundlagen zur Softwarearchitektur und Synchronisierung in Kapitel 4 vervollständigen.

Die im Rahmen dieser Arbeit entwickelte Eigenbewegungsschätzung und Lokalisierung in Kapitel 5 ist eine Grundfunktionalität für autonome Fahrzeuge und bildet die Basis für viele weitere Softwaremodule anderer Forscher am Institut.

Kapitel 6 geht auf die auf unterschiedliche Versuchsfahrzeuge adaptierte Architektur zur Fahrzeugregelung ein.

Ein im Rahmen dieser Arbeit entwickeltes hierarchisches Konzept, welches aus *Missionen*, *Aufgabentypen* und *Verhaltensweisen* besteht, um unterschiedlichste Wahrnehmungs- und Navigationsmodule der Kolleginnen und Kollegen am Institut gemeinsam nutzbar zu machen, wird in Kapitel 7 vorgestellt.

Kapitel 8 fasst die mit dem Gesamtsystem erzielten Ergebnisse zusammen und gibt eine Antwort auf die oben gestellten Forschungsfragen.

2 Stand der Technik

In diesem Kapitel wird betrachtet, wie sich das automatisierte und autonome Fahren seit den Anfängen vor 100 Jahren entwickelt haben. Die Anfänge des autonomen Fahrens werden in Abschnitt 2.1 vorgestellt. Anfang der 2000er Jahre wurde seitens des Militärs der Bedarf an autonomen Fahrzeugen entdeckt, daraufhin wurden mehrere Wettbewerbe veranstaltet, um die Entwicklung in großen Schritten voranzutreiben, wie in Abschnitt 2.2 betrachtet. Abschnitt 2.3 gibt einen Überblick über die aktuelle Entwicklung in unterschiedlichen Anwendungsbereichen, einige dort tätige Firmen und Forschungsgruppen sowie die verwendeten Ansätze.

2.1 Anfänge des autonomen Fahrens

In seinen Aufsätzen *Fahrerlos und unfallfrei. Eine frühe automobiler Technikutopie und ihre populärkulturelle Bildgeschichte* [Kröger, 2012] und *Das automatisierte Fahren im gesellschaftsgeschichtlichen und kulturwissenschaftlichen Kontext* [Kröger, 2015] arbeitet der Autor heraus, dass die Geschichte des automatisierten Fahrens bereits Anfang des 20. Jahrhunderts beginnt: Die in den 1920er Jahren einsetzende Massenmotorisierung in den USA führte zu einem starken Anwachsen der tödlichen Verkehrsunfälle, insgesamt wurden in diesem Jahrzehnt ca. 200 000 US-Bürger getötet, vornehmlich Fußgänger. Als Hauptunfallursache wurden Fehler der menschlichen Autofahrer ausgemacht, andere Ursachen für die Unfallschwere aus den Bereichen Infrastruktur und Fahrzeugkonstruktion wurden erst später registriert. Daraus ergab sich die Idee, den fehleranfälligen Menschen durch Technik zu ersetzen.

In der Luftfahrt war der *Airplane Stabilizer* als automatisches Steuerungssystem, welches auf dem Kreiselkompass basierte und als erster Autopilot gilt, seit 1914 bekannt. Die Funkfernsteuerung war ein weiterer wichtiger Bestandteil, der u. a. vom US-Militär entwickelt und zur Fernsteuerung von Torpedos, Schiffen und Flugzeugen erprobt wurde. Basierend auf diesen Arbeiten wurden bereits Mitte der 1920er Jahre erste Straßenfahrzeuge aus Begleitfahrzeugen ferngesteuert, entweder per Kabel oder per Funk. Parallel zu Vorführungen überall in den USA in den 1930er Jahren nahm das *automatisierte Fahren* auch in Filmen eine starke Rolle ein, z. B. zur Verkehrserziehung.

Ende der 1940er Jahre wurde der Tempomat bzw. die Geschwindigkeitsregelanlage erfunden [Teetor, 1950] und in der 1950ern als *Speed-o-Stat* beworben [Rowsome Jr., 1954]. Bis allerdings die ersten komplexeren Automatisierungssysteme im Fahrzeug vorgestellt wurden, dauerte es noch mehrere Jahrzehnte.

In den 1950er und 1960er Jahren ging die Forschung in Richtung eines „elektronischen Highways“ zur automatisierten Fahrzeugführung unter Zuhilfenahme eines Leitdrahts in der Straße, 1957 erfolgte eine erste Demonstration auf der Straße [Kilbon, 1960]. Auch in Europa gab es Untersuchungen zu Leitdrähten in der Straße, bspw. wurde in den 1960er Jahren im Transport Research Laboratory in Großbritannien ein Citroën

DS präsentiert, der dem Leitdraht mithilfe von Magnetsensoren folgen konnte [Kaluski, 2016].

In den 1980er Jahren wurden dann die Grundlagen für die Autonomiefunktionen gelegt, sowohl in den USA als auch in Europa.

USA und Japan 1961 wurde das *Stanford Cart* vorgestellt, welches einer weißen Linie auf dem Boden mithilfe von Kameras autonom folgen konnte [Earnest, 2012].

In den USA fuhr die Carnegie Mellon University (CMU) seit 1984 mit den Navlab-Fahrzeugen [Thorpe et al., 1991a,b] in strukturierter Umgebung. Das Versuchsfahrzeug Navlab 5 legte Mitte der 1990er Jahre in einer Erprobungsfahrt „No Hands Across America“ 2797 von 2849 Meilen von Pittsburgh nach San Diego teilautomatisiert zurück (98,2 %) [Pomerleau und Jochem, 1995]. Die automatisierte Querführung erfolgte anhand von Kamerabildern mit dem System RALPH (Rapidly Adapting Lateral Position Handler) [Pomerleau, 1995], die Bedienung von Gas und Bremse erfolgte manuell durch den Sicherheitsfahrer. [Jochem et al., 1995] geht auf die verwendeten Komponenten ein.

[Tsugawa, 1994] gibt einen Überblick über die frühe Entwicklung kamerabasierter automatisierter Fahrzeuge in Japan. Zu nennen sind hier das ab Mitte der 1970er Jahre entwickelte *Intelligent Vehicle*, welches mit bis zu 30 km/h entlang eines mit Kartenwissen geplanten Pfads zu einer Zielkoordinate fahren konnte. Die Lokalisierung basierte auf einer einfachen Koppelnavigation und daher als Open-Loop-System. Mithilfe eines vertikal angeordneten Stereokamerasystems konnten Hindernisse erkannt und ihnen ausgewichen werden.

Das Personal Vehicle System (PVS) wurde Ende der 1980er Jahre von Fujitsu und Nissan entwickelt. Die Erkennung von Straßen erfolgte mithilfe einer monokularen Kamera, die auf einer um die Hochachse drehbaren Plattform hinter der Windschutzscheibe montiert war. Zur Hinderniserkennung kam eine Stereokamera zum Einsatz. Die Fahrzeugführung erfolgte im Closed-Loop-System unter Anwendung eines Fahrermodells, welches die Zielkoordinate, einen Punkt auf der getrackten Straße in Front des Fahrzeugs, sowie die Lage des Fahrzeugs in der Fahrspur berücksichtigte. Es wurden Geschwindigkeiten bis zu 60 km/h erreicht.

Primär durch Toyota wurde das Automated Highway Vehicle System (AHVS) entwickelt, welches ebenfalls als Closed-Loop-System mit bis zu 50 km/h automatisiert fahren konnte. Die Spurerkennung erfolgte ähnlich zu PVS auf Basis von Kameras. Schwerpunkt waren im AHVS eine Robustheitssteigerung gegen unterschiedliche Beleuchtungsbedingungen und Linsenreflexionen.

Europa In Europa war das Team um Ernst-Dieter Dickmanns der Universität der Bundeswehr München (UniBw M) wegweisend: 1986 konnten erste automatisierte Testfahrten mit VaMoRs (Versuchsfahrzeug für autonome Mobilität und Rechnersehen), einem umgerüsteten Mercedes-Benz 508 D-Kastenwagen, durchgeführt werden [Dickmanns und Zapp, 1987]. Dieses Fahrzeug war mit monochromen Weitwinkel-



(a) VaMoRs. Quelle: Archiv UniBw M (b) VaMP. Quelle: Archiv UniBw M (c) Daimler VITA-1 Quelle: [Daimler Communications, 2007]

Abbildung 2.1:

Autonome Fahrzeuge der UniBw M und aus dem PROMETHEUS-Projekt.

und Tele-Kameras auf einer in zwei Achsen ansteuerbaren Pan-Tilt-Plattform zur Wahrnehmung der Umwelt ausgestattet. Unter Verwendung spatiotemporaler, d. h. räumlich-zeitlicher Modelle in einem rekursiven Schätzprozess – dieses Verfahren wird auch als *4D-Ansatz* bezeichnet – konnten der Verlauf der Straße und die eigene Position in der Spur robust bestimmt werden. Die Rechenleistung der damals verwendeten Computer ließ keine vollständige Auswertung der Kamerabilder im Bildtakt zu. Stattdessen wurden nur kleine und vor allem relevante Bereiche ausgewertet, deren Position im Bild von der Zustandsschätzung im vorherigen Zeitschritt abhing. Auf Basis der Zustandsschätzung und dynamischer Fahrzeugmodelle konnte ein Closed-Loop-Betrieb realisiert werden, bei dem Aktoren für Gas, Bremse und Lenkung entsprechend des Reglerausgangs angesteuert wurden. Abbildung 2.1a zeigt dieses Fahrzeug.

In einer ersten großen Demonstration im Jahr 1987 fuhr VaMoRs mit seiner Maximalgeschwindigkeit von 96 km/h autonom auf der (noch nicht für den öffentlichen Verkehr freigegebenen) Autobahn 92 bei Dingolfing [Dickmanns et al., 1987]. Hier wurde aber allein das Spurfolgen gezeigt, es gab keine zu berücksichtigenden anderen Fahrzeuge.

Im von 1986 bis 1994 öffentlich geförderten EUREKA-PROMETHEUS-Projekt (PROgramme for a European Traffic of Highest Efficiency and Unprecedented Safety) arbeiteten nahezu sämtliche europäische Fahrzeughersteller, viele Zulieferfirmen sowie zahlreiche Universitäten und wissenschaftliche Institute an der Weiterentwicklung des autonomen Fahrens. „Der Ausgangspunkt von PROMETHEUS sind drängende Verkehrsprobleme der Zeit. Sie sollen unternehmensübergreifend unter Verwendung neuester Technik gelöst werden, um die Wettbewerbsfähigkeit der europäischen Automobilindustrie langfristig durch das Stärken von Schlüsselfeldern zu sichern“ [Daimler Communications, 2007].

Bei der Abschlussdemonstration im Oktober 1994 in Paris navigierten die Fahrzeuge VITA-2 (Vision Information Technology Application) von Daimler-Benz und VaMP (VaMoRs-PKW) von der UniBw M (siehe Abb. 2.1b), zwei umgerüstete Mercedes-Benz S-Klasse-Fahrzeuge, vollautonom auf der französischen Ringautobahn. Unter Verwendung von Kamerasystemen erfolgten sowohl die Spurerkennung auf mehrspuriger Straße als auch das Tracking sämtlicher Fahrzeuge vor und hinter dem

eigenen Fahrzeug. Auf dieser Grundlage wurden Funktionalitäten wie Spurfolgen, Abstandhalten und Spurwechsel realisiert. Allerdings mussten die Spurwechsel aus Sicherheitsgründen vom Sicherheitsfahrer freigegeben werden [Dickmanns, 2007].

Ein weiterer Meilenstein der Gruppe um Dickmanns war eine zum Großteil autonom durchgeführte Fahrt mit VaMP von München nach Odense/Dänemark im Jahr 1995 [Lützeler et al., 1996]. Über 95 % der Strecke von 1600 km konnten dabei autonom zurückgelegt werden, inklusive über 400 Spurwechseln. Dickmanns [2020] fasst die Aktivitäten seiner Gruppe auf der Webseite *Dynamic Machine Vision* zusammen.

Ebenfalls im Rahmen von PROMETHEUS zeigten das Fraunhofer-Institut für Informations- und Datenverarbeitung (IITB) und Bosch 1993 einen umgerüsteten Mercedes-Benz Transporter VITA-1 (siehe Abb. 2.1c). Dieser konnte einerseits im Nahbereich visuell die Straße, Kreuzungen und Kreisverkehre erkennen und diesen folgen, andererseits aber auch Kartenwissen für die größere Vorausschau mit einbeziehen [Struck et al., 1993]. Eine Demonstrationsfahrt ist als Video unter [Bosch Telecom und Fraunhofer IITB, 1993] zu finden.

Mehrere italienische Institute waren mit dem MOB-LAB (MOBile LABoratory) am PROMETHEUS-Projekt beteiligt, insbesondere die Universität Parma ist hier zu nennen. Aus dieser Beteiligung gingen das Projekt *ARGO* unter der Leitung von Alberto Broggi [Broggi et al., 1999a] sowie die Ausgründung Artificial Vision and Intelligent Systems Laboratory (VisLab) hervor. Das gleichnamige Versuchsfahrzeug *ARGO*, ein älterer Lancia Thema, war mit vergleichsweise kostengünstiger passiver Sensorik, Aktorik und Rechnerhardware ausgerüstet. Die Wahrnehmung basierte hier auf monochromen Kamerabildern, welche unter Annahme einer flachen Welt mithilfe eines *Inverse Perspective Mapping* in eine virtuelle Vogelperspektive projiziert wurden. Hier konnten sowohl einfache Verfahren zur Detektion von Straßenmarkierungen als auch eine Stereo-basierte Hinderniserkennung [Bertozzi und Broggi, 1998] angewandt werden. Bei der Demonstration *The MilleMiglia in Automatico Tour* [Broggi et al., 1999b] im Juni 1998 legte das Versuchsfahrzeug etwa 94 % der 2000 km langen Strecke mit automatisierter Querführung zurück. Dies führte nicht nur in der Fachwelt zu Beachtung, sondern wurde in Nachrichtenmagazinen als „billigste[r] Autopilot der Welt“ [Der Spiegel, 1998] vorgestellt. Videos und weitere Informationen finden sich auf der historischen Projektwebseite [Broggi, 2001].

Viele heutige Fahrerassistenzsysteme gehen nach Winner und Graupner [2015] auf die Vorarbeiten in PROMETHEUS zurück, welches als Vorzeigeprojekt für eine erfolgreiche vorwettbewerbliche herstellerübergreifende Zusammenarbeit in einem europäischen Rahmen gilt. Daimler sieht dies ähnlich: „In aktuellen Fahrzeugen und in künftigen voll automatisierten Automobilen lebt PROMETHEUS fort“ [Mercedes-Benz Museum GmbH, 2016].

Auch außerhalb der an PROMETHEUS beteiligten Gruppen gab es Aktivitäten im Bereich des automatisierten Fahrens, bspw. den Fahrroboter „Klaus“: Dieser entstand in Kooperation von Volkswagen, Bosch, der Technischen Universität Braunschweig und weiteren Partnern, wurde primär für die Automatisierung der Fahrerprobung unterschiedlicher Fahrzeuge auf einem Testgelände entwickelt und im Jahr 2000

vorgestellt [Meyer, 2000]. Die Aktorik belegte den kompletten Fahrersitz und nutzte die serienmäßigen „Schnittstellen“ Lenkrad, Brems- und Gaspedal, zur Umfeldwahrnehmung dienten Laserscanner und eine Stereo-Kamera.

People-Mover Erste Pilotprojekte im Bereich der People-Mover gehen mehr als zwei Jahrzehnte zurück [Parent und Blossville, 1998].

Als Beispiel soll hier das „ParkShuttle Schiphol Airport“ genannt werden, welches von 1997 bis 2004 in Betrieb war, um auf einer Strecke von 2 km große Langzeitparkplätze mit dem Busterminal zu verbinden.

Die Software stammte von der niederländischen Firma 2getthere (damals Teil der FROG Navigation Systems, seit 2019 zugehörig zu ZF) [Lohmann und Schneider, 2008]. Das ParkShuttle fuhr auf einer separaten Fahrbahn, der sog. *segregated lane*, die vom restlichen Straßenverkehr durch beschränkte Kreuzungen und Fußgängerüberwege mit Lichtsignalanlage getrennt war. Nicht weiter spezifizierte „intelligente Sensoren“ dienten zur Hindernisprüfung des Fahrweges, hinzu kamen sog. Bumper, die bei Hinderniskontakt einen Notstopp auslösten. Zur Lokalisierung dienten Magnete im Boden, die in einem bekannten Muster verlegt waren [van Helsdingen et al., 1989].

2.2 Große Schritte im autonomen Fahren

Anfang der 2000er Jahre nahm die Entwicklung im Bereich der automatisierten und autonomen Fahrzeuge massiv Fahrt auf. Die folgenden Abschnitte gehen auf Wettbewerbe in den USA und Europa ein und beleuchten einzelne Gruppen, die im Fokus der Öffentlichkeit standen.

2.2.1 DARPA Grand Challenges

Im *National Defense Authorization Act for Fiscal Year 2001* [106th Congress of the United States of America, 2000] wurde für den militärischen Bereich in den USA die Forderung erhoben, dass bis zum Jahr 2015 ein Drittel der Landfahrzeuge unbemannt sein sollten. Durch die Organisation von drei Wettbewerben verlieh die Defense Advanced Research Projects Agency (DARPA), die Forschungs- und Entwicklungsabteilung des US-Verteidigungsministeriums, der Entwicklung im autonomen Fahren sowohl zivil als auch militärisch einen großen Schub.

Grand Challenge 2004 Die erste Grand Challenge wurde – nach Ankündigung im Jahr 2002 – im März 2004 in der Mojave-Wüste nahe Las Vegas ausgerichtet. Mehr als 100 Teams nahmen am Qualifikationslauf teil, in dem Hindernisvermeidung, Geschwindigkeit und Sicherheit überprüft wurden. Nur 15 Teams konnten sich für das Finale qualifizieren.

Kurz vor dem Start erhielten die Teams ca. 2000 Wegpunkte, welche die innerhalb des Zeitlimits von 10 Stunden zu absolvierende Route von 241 km beschrieben. Die Hauptanforderung lag also einerseits auf der Lokalisierung – primär mittels GNSS (Globales Navigationssatellitensystem, engl. *Global Navigation Satellite System*) wie dem US-amerikanischen GPS (Global Positioning System) oder dem russischen GLONASS (Globales Satellitennavigationssystem, russ. *Globalnaja nawigazionnaja sputnikowaja sistema*). Andererseits war eine Hindernisvermeidung mithilfe lokaler Sensorik in anspruchsvollem Gelände unabdingbar: abseits der befestigten Straße, in engen Durchfahrten, in felsigem Terrain und in Serpentinaen. In dieser Umgebung kam es spätestens nach wenigen Kilometern zum Ausfall sämtlicher Roboter, wobei das *Team Red* der CMU unter der Leitung von William (Red) Whittaker noch das beste Team war: sein Fahrzeug *Sandstorm* konnte 11,9 km zurücklegen, bevor es sich am Straßenrand festfuhr und Feuer fing.

Das Fazit zu dieser ersten Grand Challenge 2004 war gemischt: einerseits wurde sie als Debakel bezeichnet [Hooper, 2004], was primär auf überzogene Erwartungen der Öffentlichkeit, die von der DARPA nicht gedämpft wurden, zurückzuführen ist. Primär wird die Veranstaltung aber als Erfolg gesehen, da sie die Entwicklung dieser neuen Technologie angestoßen und in der Öffentlichkeit Interesse für diese militärisch wichtige Technologie geweckt hat, was in den Augen von DARPA-Direktor Anthony J. Tether mit dem Apollo-Programm vergleichbar sei: „It has sparked an interest in science and technology in this country that we haven't seen since the 1960's with the Apollo program“ [Markoff und Broder, 2004].

Grand Challenge 2005 Die Grand Challenge 2005 wurde direkt nach der 2004er-Veranstaltung angekündigt und wurde ebenfalls in der Mojave-Wüste ausgerichtet. Zu dieser Veranstaltung bewarben sich 195 Teams, viele davon erst durch die Grand Challenge 2004 motiviert. 43 Teams wurden zu einem Qualifikationslauf, dem *National Qualification Event (NQE)*, eingeladen, in dem sich 23 von ihnen für das finale Rennen qualifizieren konnten.

Knapp 3000 Koordinaten wurden – analog zum Vorjahr – kurz vor dem Start zur Beschreibung der Wettbewerbsstrecke von 212 km bereitgestellt.

Fünf Teams konnten die komplette Strecke bewältigen, vier von ihnen innerhalb des Zeitlimits von 10 Stunden.

Sieger wurde das *Stanford Racing Team* unter der Leitung von Sebastian Thrun mit seinem in Abb. 2.2a gezeigten Fahrzeug *Stanley*, einem umgerüsteten VW Touareg, welcher nach 6:37 Stunden das Ziel erreichte. Details zu den verwendeten Lösungsansätzen und Algorithmen finden sich in [Thrun et al., 2006] sowie auf der Team-Webseite [Stanford Racing, 2005].

Das *Red Team* unter der Leitung von William (Red) Whittaker vom Robotics Institute der CMU nahm mit zwei Fahrzeugen teil: *Sandstorm*, ein 1986er Model 998 HMMWV, wurde zweiter vor seinem Schwesterfahrzeug *Highlander*, einem 1999er H1. Nach [Urmson et al., 2006, Kapitel 2] bedeuten zwei unterschiedliche Fahrzeuge zwar erhöhten Integrations- und Unterhaltungsaufwand, bieten aber

als Hauptvorteil eine erhöhte Zeit für Softwaretests, welches der Schlüssel für erhöhte Robustheit und bessere Fähigkeiten darstellt. Die Parametrierung der beiden Fahrzeuge verfolgte zudem unterschiedliche Ansätze: eines war eher aggressiv (mit dem Fokus auf „Schnelligkeit“), das andere eher defensiv (mit dem Fokus auf „Ankommen“) ausgelegt. Weitere Details zu den Fahrzeugen und dem Team finden sich auf der Webseite [Red Team Robot Racing, 2006].

Auf dem vierten Platz folgte das *Gray Team* mit dem Fahrzeug *Kat-5* [Trepagnier et al., 2006].

Das *Team TerraMax*, welches im Gegensatz zu den anderen Teams mit einem großen LKW antrat, setzte sich aus Mitgliedern von Oshkosh Truck, Rockwell Collins sowie VisLab, einem Spin-Off der Universität Parma unter Leitung von Alberto Broggi, zusammen [Braid et al., 2006]. Eine besondere Betonung legte das Team auf den hohen Anteil an Computer Vision. TerraMax erreichte als fünftes Fahrzeug das Ziel, allerdings mit 12:51 Stunden jenseits des Zeitlimits. Grund waren die großen Fahrzeugabmessungen, die eine langsamere Fahrt in enger Umgebung erforderten, und dejustierte Sensoren nach einer Kollision.

Alle anderen Teams schieden während des Rennens aus, wobei 22 von 23 eine größere Strecke als das beste Fahrzeug in 2004 zurücklegen konnten.

Insgesamt war eine extreme Verbesserung gegenüber 2004 zu erkennen, die Grand Challenge 2005 gilt bis heute als voller Erfolg. Hauptgrund war sicherlich eine wesentlich professionellere Arbeit in den einzelnen Teams, die die richtigen Lehren aus 2004 gezogen hatten. Zusätzlich wurde die Strecke durch die Veranstalter entschärft, damit Fahrzeuge ins Ziel gelangen konnten: „The 2005 Grand Challenge was made manageable by removing obstacles from the vehicles' path, and providing dense and precise GPS data of the race route to the finalists“ [Buehler et al., 2007, Seite XIV]. Die Strecke wurde mit einem Bulldozer präpariert, wodurch viele Hindernisse beiseite geräumt wurden und auf beiden Seiten kleine Sandwälle entstanden, welche den teilnehmenden Robotern durch gute Erkennbarkeit v. a. in 3D-LiDAR-Daten (Light Detection And Ranging) die Spurführung erleichterten.

In [Buehler et al., 2007] finden sich zahlreiche Artikel der einzelnen Teams, die tieferen Einblick in deren Fahrzeuge und Ansätze bieten.

Urban Challenge 2007 Im Vorfeld der Urban Challenge 2007 wurden in einem Bewerbungsverfahren elf Teams von der DARPA als *Track A Team* ausgewählt, denen Entwicklungskostenunterstützung in Höhe von je 1 Mio US\$ gezahlt wurde, alle anderen *Track B Teams* finanzierten die Entwicklung aus Eigenmitteln. Aus anfangs über 100 Bewerbern wurden 53 Teams im Rahmen von Site-Visits begutachtet, 35 von ihnen konnten sich für das Halbfinale, das sog. National Qualification Event (NQE), qualifizieren. Im Finale waren ursprünglich 20 Startplätze geplant, aufgrund von Sicherheitsbetrachtungen wurden allerdings nur elf Teams zugelassen. Sowohl das NQE als auch das Finale fanden auf einem vorstadtähnlichen Parcours in der kalifornischen George Air Force Base in Victorville statt.



(a) Siegerfahrzeug *Stanley* (Stanford Racing Team, Stanford University) der DARPA Grand Challenge 2005 Quelle: Volkswagen.



(b) Die drei Siegerfahrzeuge der DARPA Urban Challenge 2007 (von links): *BOSS* (Tartan Racing Team, CMU, 1. Platz), *Junior* (Stanford Racing Team, Stanford University, 2. Platz) and *Odin* (Team VictorTango, Virginia Tech, 3. Platz). Quelle: [DARPA, 2007a].

Abbildung 2.2:

Siegerfahrzeuge der DARPA Grand Challenge 2005 und Urban Challenge 2007.

Für diesen Parcours wurde ein für alle Teams identisches Road Network Definition File (RNDF) bereitgestellt [DARPA, 2007b]. Diese Kartendatei enthielt eine geometrische und topologische Beschreibung des Straßennetzes, wobei für jede Spur die Breite und Typ der linken bzw. rechten Begrenzungslinie sowie Nachbarspuren hinterlegt waren, sowie von Freiflächen. Zusätzlich waren einzelne Koordinaten als Checkpoint markiert.

Mindestens ein Team, das *Tartan Racing Team* der CMU, erweiterte das ursprünglich wenig detaillierte RNDF mithilfe von Luftbildern [Urmson et al., 2008, Sec. 9.1] sowie von im NQE gesammelten Informationen [Team Jefferson, 2007], um im Finale mit der sehr präzisen Karte die Navigationsfähigkeiten zu verbessern. Eine Gegenüberstellung der beiden Karten findet sich in [Team Jefferson, 2007, Seite 7].

Lag der Schwerpunkt in der Grand Challenge noch primär auf der präzisen Lokalisierung und Hindernisvermeidung in leichtem Offroad-Gelände, so war in der Urban Challenge vielmehr die komplexere Umgebung mit dynamischen Objekten die Herausforderung. Diese dynamischen Objekte galt es wahrzunehmen, ihr Verhalten vorherzusagen und mit ihnen zu interagieren [Berger und Rumpe, 2008]. Die entstehenden Verkehrssituationen mussten entsprechend interpretiert und die kalifornischen Verkehrsregeln [Schwarzenegger et al., 2007] wie der *All-Way-Stop* beachtet werden, zusätzlich zu den Regeln der Urban Challenge [DARPA, 2007c].

Alle Teams befanden sich gleichzeitig auf dem Parcours, wobei die Reihenfolge der zu absolvierenden Aufgaben, welche als Mission Data File (MDF) gemäß [DARPA, 2007b] kurz vor dem Start übergeben wurden, variierte. Zusätzlich war eine Reihe manuell gesteuerter Fahrzeuge unterwegs, um bestimmte Verkehrssituationen zu

simulieren. Die Wettbewerbsstrecke betrug für alle Teams ca. 96 km, welche innerhalb von sechs Stunden Netto-Fahrzeit zu absolvieren waren.

Insgesamt kamen sechs der elf Finalisten ins Ziel, die Sieger sind in Abb. 2.2b gezeigt: Das *Tartan Racing Team* der CMU konnte mit seinem Fahrzeug *BOSS* den mit 2 Millionen US-Dollar dotierten ersten Platz erreichen [Urmson et al., 2008]. Auf der Team-Webseite [Tartan Racing, 2007] sind weitere Details zu finden. Auf dem zweiten Platz folgte das *Stanford Racing Team* und gewann 1 Million US-Dollar. Das Team unter der Leitung von Sebastian Thrun bestand überwiegend aus Mitgliedern des Stanford Artificial Intelligence Laboratory der Stanford University sowie von Volkswagen [Montemerlo et al., 2008]. Der Versuchsträger *Junior*, ein umgerüsteter VW Passat, konnte das Rennen zwar in kürzerer Zeit als *BOSS* absolvieren, bekam allerdings wegen kleinen Verstößen gegen die Verkehrsregeln eine Zeitstrafe. Weitere Details finden sich auf der Team-Webseite [Stanford Racing Team, 2007]. Mit seinem Fahrzeug *Odin* erreichte das Team *VictorTango*, bestehend aus Virginia Tech University und TORC Technologies, den mit 500 000 US-Dollar dotierten dritten Platz [Bacha et al., 2008]. Auf der Team-Webseite [Team Victor Tango, 2007] finden sich weitere Informationen.

Unter den Finalisten waren auch zwei weitgehend deutsche Teams. Das Team *CarOLO*, beheimatet an Lehrstühlen unterschiedlicher Disziplinen der Technischen Universität Braunschweig, nahm mit dem Fahrzeug *Caroline*, einem VW Passat, am Rennen teil [Rauskolb et al., 2008]. „Nach einem Zusammenstoß [nach ca. 17 km] mit dem Team MIT, dessen Ursachen trotz Analyse der Sensorrohdaten bis heute ungeklärt sind, belegte das Team *CarOLO* als bestes nicht-amerikanisches Team den siebten Platz“ [Berger und Rumpel, 2008]. Es war das letzte von fünf durch die Schiedsrichter aus dem Rennen genommenen Teams [Belfiore, 2007a].

Das Team *AnnieWay* mit dem gleichnamigen Fahrzeug, ebenfalls einem VW Passat, bestand aus Mitgliedern des Sonderforschungsbereich Transregio 28 – „Kognitive Automobile“ (KogniMobil). Beteiligt waren die Universität und das Forschungszentrum Karlsruhe (heute zusammen das KIT), die Technische Universität München und das Institut für Technik Autonomer Systeme (TAS) der Universität der Bundeswehr München. Details zur Umsetzung finden sich in [Kammel et al., 2008], die Anteile von TAS sind in [von Hundelshausen et al., 2008] näher beschrieben. *AnnieWay* schied aufgrund eines Softwarefehlers frühzeitig aus [Belfiore, 2007b].

Während alle anderen Finalisten mit Personenkraftwagen (PKWs) teilnahmen, kam beim *Team Oshkosh* der *TerraMax* zum Einsatz, ein umgerüsteter militärischer Lastkraftwagen (LKW) [Chen et al., 2008]. Das Team bestand neben dem Fahrzeughersteller Oshkosh aus den Firmen Teledyne, dem italienischen VisLab der Universität Parma, der deutschen Ibeo Automotive Sensor GmbH sowie der Auburn University. Im Finale kam das Fahrzeug nach mehreren erfolgreich absolvierten Teilmissionen aus unterschiedlichen Gründen für längere Zeit zum Stillstand, anschließend fuhr es auf ein Gebäude zu und wurde von den Schiedsrichtern aus dem Rennen genommen [Belfiore, 2007b].

Insgesamt lässt sich festhalten, dass die Sensorausstattung der meisten Fahrzeuge vergleichbar war: 3D- und 2D-LiDAR-Sensoren zur Hinderniserkennung, GNSS-Empfänger zur Lokalisierung sowie Kamerasysteme in unterschiedlicher Ausprägung (Mono, Stereo) für verschiedene Einsatzzwecke. Die entscheidenden Unterschiede sind eher in der Robustheit der entwickelten Algorithmen zu finden. Bei den erfolgreichen *Track A Teams* konnte durch die wesentlich bessere Finanzierung eine professionellere Entwicklung und ein stark erhöhter Testaufwand betrieben werden als bei anderen, eher universitär geprägten Teams mit minimalem Budget. In [Buehler et al., 2009] finden sich weitere Artikel der einzelnen Teams, die tieferen Einblick in deren Fahrzeuge und Ansätze bieten.

Aus Sicht der DARPA war mit den Grand Challenges sowie der Urban Challenge die Entwicklung im autonomen Fahren stark genug angestoßen worden, sodass die weitere Entwicklung autonomer Landfahrzeuge in Richtung Produkt und Erfüllung der Forderung aus [106th Congress of the United States of America, 2000] unabhängig von Wettbewerben verlief. In der Nachfolgeveranstaltung DARPA Robotics Challenge [DARPA, 2015] (2012–2015) lag der Fokus auf multifunktionalen Robotern für Rettungseinsätze, wobei die Szenarien von den Ereignissen im japanischen Fukushima motiviert wurden. Die anschließende DARPA Subterranean Challenge [DARPA, 2019] (2018–2021) beschäftigte sich mit Rettungseinsätzen in unterirdischen Szenarien, z. B. in Bergwerken, Verkehrstunneln oder städtischen Untergrundbauten [Lindner, 2019].

2.2.2 ELROB

Auf europäischer Seite wurde 2006 die M-ELROB (Military European Land Robot Trial) als Pendant zu den DARPA-Wettbewerben geschaffen und findet seitdem alle zwei Jahre statt. Die Szenarien sind aus Militär- und Katastrophenschutzperspektive motiviert, u. a. Aufklärung in einem beschädigten Gebäude, die Bergung von Verletzten, der Lastentransport in *Konvoi* und *Mule* oder die Aufklärung und Entschärfung von Kampfmitteln (EOD, engl. *Explosive Ordnance Disposal*). Bei diesem Vergleich von Fähigkeiten der Robotikplattformen geht es den Veranstaltern weniger um die Kür eines Siegers, sondern vielmehr um das Aufzeigen der aktuellen Grenzen in realistischen Szenarien [Schneider et al., 2015]. Den teilnehmenden Teams – von der Rüstungsindustrie über Start-Ups bis hin zum akademischen Bereich – wird eine Möglichkeit gegeben, ihre Systeme in realitätsnahen Szenarien zu erproben, und der militärische Nutzer bekommt einen Einblick über die Leistungsfähigkeit aktueller Systeme.

Für autonom fahrende Fahrzeuge ab der PKW-Größe sind Konvoi und Mule geeignet, für LKW kommt nur der Konvoi in Betracht. Über die Jahre gab es hier eine Entwicklung zu mehr Teams aus der Industrie, es ist ein Trend in Richtung Produkt zu erkennen.

In den ungeraden Jahren von 2007 bis 2011 wurde die Schwesterveranstaltung C-ELROB (Civilian European Land Robot Trial) ausgerichtet, die Szenarien waren eher aus der Katastrophenschutzperspektive motiviert. Hier war für Fahrzeuge in

PKW-Größe nur die *Autonome Navigation* anhand einer Liste von Wegpunkten geeignet. Im Teilnehmerkreis gab es eine große Überdeckung.

Die euRathlon 2013 lief als Nachfolgeveranstaltung der C-ELROB und übernahm einige Szenarien. In den Folgejahren waren die Wettbewerbe nicht mehr für größere autonome Fahrzeuge geeignet, nach 2015 wurde dieser Ableger der ELROB eingestellt.

2.2.3 VisLab Intercontinental Autonomous Challenge

Auch über die Wettbewerbe hinaus ist VisLab zu nennen, ein Spin-Off der Universität Parma/Italien unter Leitung von Alberto Broggi. Bei der VisLab Intercontinental Autonomous Challenge (VIAC) fuhren vier identisch aufgebaute elektrisch angetriebene Fahrzeuge vom Typ Piaggio Porter im Konvoi (teil)autonom von Parma/Italien zur Expo 2010 nach Shanghai/China [Broggi et al., 2011] – eine Strecke von ca. 13000 km innerhalb von drei Monaten. Bei der Sensorausstattung liegt der Schwerpunkt auf günstigen Kameras, sowohl monokular als auch Stereo. Zusätzlich kommen vier einstrahlige LiDAR-Sensoren zur Hinderniserkennung im Nahbereich zum Einsatz, ein vierstrahliger LiDAR-Sensor dient zusätzlich zum Tracking von Fahrzeugen und anderen Objekten.

Bei dieser Demonstration konnten die einzelnen Fahrzeuge unterschiedliche Aufgaben übernehmen: Das erste Fahrzeug fuhr auf Autobahnen und anderen Abschnitten, die keine Entscheidungen erforderten, autonom, in anderen Situationen wurde es vom menschlichen Fahrer gesteuert. Das zweite Fahrzeug folgte im Konvoi-Modus auf Basis von übertragenen GNSS-Wegpunkten und der lokalen Sensorik, daher konnte hier ein wesentlich größerer zeitlicher Anteil autonom bewältigt werden. Die Fahrzeuge kamen wechsel- und paarweise zum Einsatz, während die anderen in Begleitfahrzeugen transportiert und aufgeladen wurden. [Bertozzi et al., 2011a,b] geben einen Überblick über die Fahrzeuge und Algorithmen.

2.3 Aktuelle Entwicklung

In diesem Abschnitt wird die Entwicklung vom „Wettbewerbsfahrzeug“ hin zum „Produkt“ betrachtet, die sich seit den 2010er Jahren ergeben hat.

Die Entwicklung in den Anwendungsbereichen verläuft nicht parallel, sondern aufgrund der unterschiedlichen Komplexität zeitlich versetzt:

[Der damalige ZF-Chef] Scheider rechnet übrigens damit, dass elektrifizierte und autonome Fahrzeuge zunächst bei den Nutzfahrzeugen zum Einsatz kommen werden, allein schon aus Gründen der Total Cost of Ownership (TCO), der Gesamtkostenbetrachtung. So gäbe es im Nutzfahrzeugsbereich genügend Einsatzbereiche, die nicht die volle Komplexität des Innenstadtbereichs haben. „Wenn ich auf dem Betriebshof anfangen oder ein Feld automatisch pflügen, dann ist das schneller zu

realisieren, als einen Pkw im Innenstadtbereich mit Fußgängern, Fahrradfahrern und einer engen Verkehrsdichte fahren zu lassen“, betont Scheider. „So etwas braucht Entwicklungszeit und sie braucht Erfahrung. Dieses Erfahrung werden wir über das Nutzfahrzeug und den People Mover erreichen, der Pkw kommt später.“ (Mertens [2018])

Nach einer kurzen Betrachtung der rechtlichen Entwicklung in Abschnitt 2.3.1 sind die kommenden Abschnitte dieser Reihung folgend gegliedert: In den Abschnitten 2.3.2 und 2.3.3 werden Anwendungen in der Logistikbranche vorgestellt, weitere Beispiele aus dem Nutzfahrzeugumfeld folgen in den Abschnitten 2.3.4 bis 2.3.6. Abschnitt 2.3.7 legt den Fokus auf militärische Anwendungen.

Die Entwicklung im PKW-Bereich, aber auch allgemein im Personentransport wird durch eine Reihe von Start-Ups bzw. Tech-Konzernen in den USA (Abschnitt 2.3.8) sowie von den klassischen Autoherstellern und Zulieferern (Abschnitt 2.3.9) vorangetrieben. Der Spezialbereich der People Mover bzw. der urbanen Mobilität wird in Abschnitt 2.3.10 vorgestellt.

Daneben gibt es aber auch weiterhin Wettbewerbe wie die Indy Autonomous Challenge (IAC)¹, die auf den DARPA Grand Challenges aufbaut. Die IAC ist ein internationaler, mit Preisgeld ausgestatteter Wettbewerb zwischen Universitäten zur Programmierung modifizierter Dallara IL-15-Rennwagen, um diese gemeinsam auf der Rennstrecke und im fahrdynamischen Grenzbereich mit Level-5-Autonomie zu betreiben.

2.3.1 Rechtlicher Rahmen

Das Wiener Abkommen von 1968 forderte: „Jedes Fahrzeug, und miteinander verbundene Fahrzeuge, müssen, wenn sie in Bewegung sind, einen Führer haben. Jeder Führer muss dauernd sein Fahrzeug beherrschen“ (vgl. Artikel 8, Absatz 1 und 5, United Nation Conference On Road Traffic, 19. Convention on Road Traffic, Vienna, 8. November 1968). Dieses Passus hat in vielen Ländern lange die rechtlichen Möglichkeiten für die Zulassung von hochautomatisierten Fahrzeugen eingeschränkt.

Die USA sind dem Wiener Abkommen nie beigetreten, daher konnten dort bereits 2011 in Nevada und 2012 in Florida, Kalifornien und Columbia Gesetze erlassen werden, die die Erprobung von automatisierten Fahrfunktionen im öffentlichen Straßenverkehr ohne Einschränkung erlauben [Dudenhöffer und Schneider, 2015]: „The operator need not pay attention while the car is operating itself.“

2016 wurde ein Änderungsvorschlag des Wiener Abkommens ratifiziert: „Systeme, die für den Fahrer übersteuerbar oder abschaltbar sind, erfüllen die geforderte Beherrschung durch den Fahrer. Hochautomatisierte Systeme sind damit unter diesen Bedingungen zulässig.“ Damit waren in Deutschland die Grundlagen für die nationale Gesetzgebung gelegt, die in [Deutscher Bundestag, Ausschuss für Verkehr und digitale Infrastruktur, 2017] erfolgte. Hier wurden auch die Haftungsfragen für den automatisierten Betrieb bis Level 3 geklärt: „Nunmehr haftet der Hersteller,

¹<https://www.indyautonomouschallenge.com/>

wenn das Fahrzeug selbstständig fährt und der Fahrer nur dann, wenn er das Steuer trotz Aufforderung der Fahrzeugelektronik nicht wieder selbst übernommen hat“ [Link, 2017].

In einem Folgegesetz wurden die Rahmenbedingungen für den Betrieb von Level-4-Fahrzeugen ohne Fahrer im Straßenverkehr gelegt [Bundesministerium für Digitales und Verkehr, 2021], die Erlaubnis erfolgt örtlich begrenzt. Die Einsatzszenarien umfassen bspw. Shuttle-Verkehre zwischen festen Stationen, People Mover oder Automated Valet Parking.

Die oben beschriebenen Einschränkungen gelten primär für den öffentlichen Straßenverkehr. Kommen automatisierte Fahrzeuge nur auf Privatgrund, bspw. einem Betriebsgelände oder einer landwirtschaftlichen Fläche, zum Einsatz, so gelten mit der Maschinenverordnung andere rechtliche Grundsätze, die den Betrieb erleichtern.

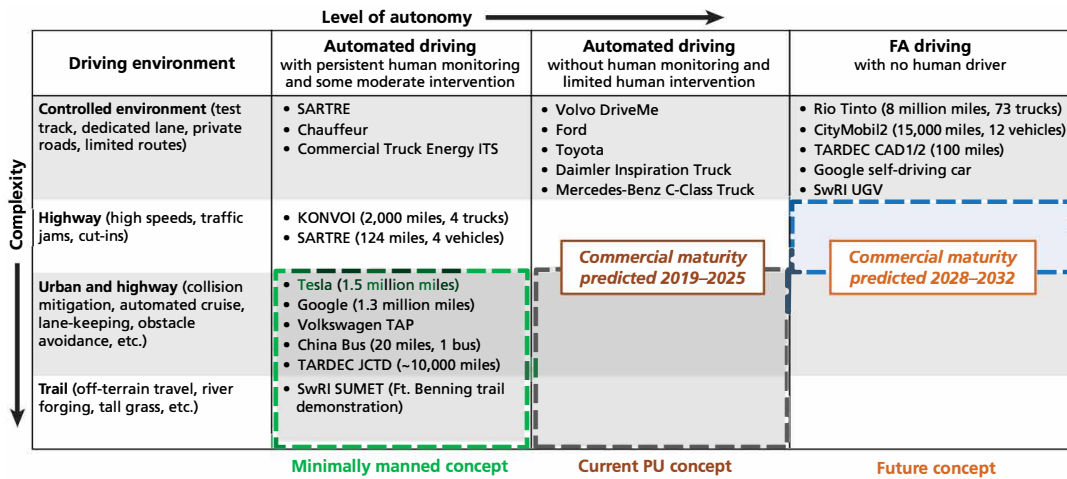
An dieser Stelle sei angemerkt, dass für die Zulassung von Erprobungsfahrzeugen der Automobilhersteller im öffentlichen Straßenverkehr schon lange Sonderregeln gelten:

Bislang wurde die Erprobung von Kraftfahrzeugen ausschließlich durch die zuständigen Landesbehörden nach § 19 Absatz 6 in Verbindung mit § 70 Absatz 1 Nummer 1 und 2 StVZO genehmigt. Mit Inkrafttreten der Verordnung zur Genehmigung und zum Betrieb von Kraftfahrzeugen mit autonomer Fahrfunktion in festgelegten Betriebsbereichen (AFGBV) erfolgt die Erteilung von Erprobungsgenehmigungen von automatisierten und autonomen Fahrfunktionen gemäß § 1i StVG in Verbindung mit § 16 AFGBV auf Antrag des Fahrzeughalters durch das KBA. (Kraftfahrt-Bundesamt [2023])

Da die Zulassung der institutseigenen Versuchsträger durch das Zentrum Kraftfahrwesen der Bundeswehr (ZKfWBw) erfolgt, gelten für diese Fahrzeuge Sonderregeln und andere Abnahmekriterien als im zivilen Bereich. Die Zulassung erlaubt explizit die Erprobung autonomer Fahrfunktionen auch im Straßenverkehr, allerdings unter gewissen Auflagen wie der Forderung nach eingewiesenen Sicherheitsfahrern und Operatoren. Damit wird auch für ein Uni-Institut, das nur über begrenzte Entwicklungskapazitäten verfügt, der Betrieb der Fahrzeuge mithilfe von organisatorischen und technischen Maßnahmen handhabbar.

2.3.2 Platooning im Straßengütertransport – zivil und militärisch

Wie in Abschnitt 1.1 schon motiviert, besteht im (Straßen-)Gütertransport aus mehreren Gründen der Bedarf nach stärkerer Automatisierung: Personal, Effizienz und Sicherheit. Die technischen Ansätze unterscheiden sich jedoch zwischen zivilen und militärischen Anwendungen. Abbildung 2.3 gibt einen Überblick über Projekte und Demonstrationsfahrten aus beiden Domänen und darüber hinaus.



NOTES: CAD = capabilities advancement demonstration; ITS = intelligent transportation system; JCTD = joint capability technology demonstrator; KONVOI = convoy [in German]; SARTRE = Safe Road Trains for the Environment; SUMET = small unit mobility enhancement technology; SwRI = Southwest Research Institute; TAP = temporary auto pilot; TARDEC = Tank Automotive Research, Development and Engineering Center; UGV = unmanned ground vehicle.

Abbildung 2.3:

Übersicht über Demonstration autonomer Fahrzeuge. Quelle: [McKay et al., 2020, S. xiii]

Zivil Im zivilen Straßengütertransport gibt es Bestrebungen, den Fahrer nur noch für die erste und letzte Meile sowie unvorhergesehene Situationen in die Verantwortung zu nehmen, sodass Lenk- und Ruhezeiten neu gedacht werden können. Beispielsweise solle ein Fahrer am Stück – teilweise schlafend – von Deutschland nach Portugal fahren können, damit würde sich die Kilometerleistung pro Fahrer drastisch erhöhen.

Hier kommt das sog. *Platooning* ins Spiel, eine Variante des *bedingt automatisierten Fahrens*, auch als *Minimally Manned* bezeichnet: Damit mehrere Fahrzeuge um Kraftstoff zu sparen dichter hintereinander fahren können als es die Straßenverkehrsordnung im Allgemeinen erlaubt, wird die Längsführung zum Vordermann sensorbasiert durchgeführt, meist mit Hilfe von Radar-Sensoren (Radio Detection And Ranging). Zusätzlich kommt V2V-Kommunikation (Fahrzeug-zu-Fahrzeug, engl. *Vehicle-to-Vehicle*) zum Einsatz, um Informationen über Beschleunigung und Verzögerung zu übertragen, damit die Folgefahrzeuge einen Ziehharmonikaeffekt vermeiden und auch bei Notbremsungen des Vorderfahrzeugs einen sicheren Abstand halten können. Die Querverführung geschieht in den meisten Fällen visuell zur Spur.

Die Entwicklung in diesem Bereich geht bis in die 1980er Jahre zurück [Bergenheim et al., 2012, Tsugawa et al., 2016]: Das kalifornische PATH-Programm (Partners for Advanced Transit and Highways) wurde 1986 gestartet, 1994 erfolgte eine Demonstration mit einem aus vier Fahrzeugen bestehenden *Platoon* in San Diego [Shladover, 2007].

Auf europäischer Seite sind EU-geförderten Projekte PROMOTE CHAUFFEUR I + II zu nennen (1996 bis 2003), in denen unter Federführung von Daimler die Grundlagen für das Platooning unter Einbezug von V2V-Kommunikation gelegt wurden [Harker, 2001].

Das vom Bundesministerium für Wirtschaft und Technologie (BMWi) geförderte Projekt KONVOI baute darauf von 2005 bis 2009 auf, durchgeführt wurde das Projekt von der RWTH Aachen mit Industriepartnern und weiteren öffentlichen Einrichtungen [Henning et al., 2009, Kunze et al., 2009]. Mit vier für automatisiertes Fahren umgerüsteten schweren LKW der Marken MAN und IVECO wurden insgesamt 3100 km als Platoon im Realverkehr zurückgelegt.

Die RWTH Aachen war mit Volvo und weiteren Industriepartnern ebenfalls am EU-geförderten Projekt SARTRE (Safe Road Trains for the Environment) beteiligt, hier wurde ein Misch-Platoon aus drei PKW und zwei LKW untersucht [Pander, 2012].

Auch international gab es ähnliche Projekte, bspw. wurde von der japanischen New Energy and Industrial Technology Development Organization das gEnergy ITS project gefördert. Eines der Ziele war die CO₂-Einsparung durch Platooning [Sugimachi et al., 2013].

Über viele Jahre ging es klar in Richtung Produkt, die Automobilhersteller forcierten diese Themen. Der Mercedes-Benz *Highway Pilot Connect* in Abb. 2.5a nutzte fahrzeuggebundene Sensorik sowie V2V-Kommunikation und ermöglichte so auf Autobahnen die Kopplung mehrerer autonomer Fahrzeuge zu einem Platoon [Brüninghaus, 2016]. Anfang 2019 beendete Daimler dann die Aktivitäten im Platooning, weil sich der Kraftstoffverbrauch nicht wie erwartet reduzierte, wenn die Platoons oft auseinanderbrechen [Hubik, 2019]. Stattdessen würden andere Aktivitäten im Bereich hochautomatisierter Nutzfahrzeuge forciert.

Militärisch Im militärischen Kontext kommt vor allem der Schutz von Menschenleben als Motivation für unbemannte Fahrzeuge hinzu, aber auch die Kostenreduktion durch Verzicht auf Panzerung. Aus Sicherheitsgründen ist hier im Regelfall ein gemischter Konvoi aus bemannten geschützten Fahrzeugen sowie unbemannten automatisierten Fahrzeugen angedacht.

Die Variabilität der Einsatzszenarien ist hier wesentlich größer als im zivilen Platooning, welches oft auf Autobahnen beschränkt ist. Daher müssen andere Konzepte für die Fahrzeugführung zum Einsatz kommen, da bspw. Spurmarkierungen nicht vorausgesetzt werden können. Ziel ist eine Querführung exakt in der Spur des Vordermanns bzw. Führungsfahrzeugs, hierzu muss dessen Position präzise getrackt werden. Je nach Lage müssen auch große Abstände zwischen den Fahrzeugen bis hin zum Sichtverlust realisiert werden – dann wird auch hier V2V-Kommunikation benötigt.

Die im Zivilen verwendete aktive Sensorik wie Radar oder LiDAR ist nicht in jedem militärischen Szenario zulässig, da die automatisierten Fahrzeuge dadurch leichter durch den Gegner aufklärbar werden. Durch den Verzicht auf aktive Sensorik und Funkkommunikation kann das elektromagnetische Spektrum reduziert werden, allerdings erzielen Lösungen auf Basis von passiven Sensoren wie (Stereo-)Kameras derzeit noch nicht die notwendige Performance – hier sind Abstriche in der Präzision hinzunehmen, und es ist zwingend Sichtverbindung notwendig.



(a) TULF und StrAsRob auf der M-ELROB 2018. (b) Konvoi der US Army. Quelle: Oshkosh (c) Gemeinsame Erprobung von InterRoC und US Army. Quelle: [Ball, 2022]

Abbildung 2.4: Gütertransport im militärischen Konvoi.

In Deutschland sind die BAAINBw-Studien (Bundesamt für Ausrüstung, Informationstechnik und Nutzung der Bundeswehr) TULF (Technologieträger Unbemanntes Landfahrzeug) (2011-2020) und StrAsRob (Straßentransport mit Assistenzfunktionen von Robotern) (2012-2015) zu nennen [Luettel et al., 2017], bei denen es primär um die Algorithmenentwicklung für den Gütertransport im Konvoi ging (siehe Abb. 2.4a), aber auch Mensch-Maschine-Aspekte [Baltzer et al., 2017] betrachtet wurden. Zur Wahrnehmung der Umwelt und zum Tracking des Führungsfahrzeugs kam ein Setup aus aktiven und passiven Sensoren zum Einsatz, zusätzlich wurde V2V-Kommunikation integriert. Das Institut für Technik Autonomer Systeme (TAS) war in beiden Studien beteiligt, die Versuchsfahrzeuge werden in den Abschnitten 3.5 und 3.6 vorgestellt.

[McKay et al., 2020] analysiert den aktuellen Stand für automatisierte Konvois der US Army (siehe Abb. 2.4b). Hier sind vor allem die Aktivitäten der TARDEC (Tank Automotive Research, Development and Engineering Center) ab den 2010er Jahren zu nennen, bei denen die Firma Robotic Research als Technikpartner involviert ist. Dieses System hat sich von Studien zum Beschaffungsprogramm weiterentwickelt [Lee, 2019], es sind schon größere Konvois im Testbetrieb [Tadjeh, 2022]. Dieses System zum spurtreuen Folgen basiert bei Sichtkontakt auf einer sensorbasierten Erkennung des Führungsfahrzeugs sowie einer Trilateration auf Basis von UWB-Sensoren (Ultrabreitband, engl. *Ultra Wide Band*), für große Abstände aber primär auf einer mittels LiDAR-Sensoren erstellten Karte, die zwischen den Fahrzeugen kommuniziert wird.

Im vom BAAINBw beauftragten Projekt InterRoC (Interoperable Robotic Convoy) wurden die Autonomie-Kits dieser Systeme in zwei Bundeswehr-LKWs eingerüstet [Hoffmann, 2022]. Abbildung 2.4c zeigt die derzeit laufende Erprobung, die teils gemeinsam mit der US Army stattfindet [Ball, 2022].

Auch in Australien laufen Studien zum Multi-Fahrzeug-Konvoi [Australian Road Research Board (ARRB), 2022]. In allen hier aufgeführten Systemen ist das Führungsfahrzeug bemannt.

In den österreichischen Projekten SafeCon (2011-2013) [Polzer und Huber, 2013] und RelCon (bis 2016) [Kubinger und RelCon-Team, 2016] wurden zusätzlich Szenarien



(a) Automatisiert fahrende LKW, über WLAN zum Platoon mit Straßenzulassung vernetzt. Quelle: Daimler



(b) Kamag E-Wiesel AGV zum innerbetrieblichen Warentransport. Quelle: KAMAG



(c) Schneeräumen auf dem Flughafenvorfeld. Quelle: [Valtra Inc., 2019]

Abbildung 2.5:

Fahrzeuge aus dem Bereich des Platooning, der innerbetrieblichen Logistik sowie für den Einsatz auf dem Flughafenvorfeld.

aus dem Bereich des Katastrophenschutzes betrachtet und auch Wahrnehmung mit passiver Sensorik entwickelt [Woeber et al., 2012].

2.3.3 Innerbetriebliche Logistik

Für den Warentransport von Waren innerhalb eines Betriebs – sowohl indoor in Lagerhäusern als auch outdoor – kommen häufig sog. Fahrerlose Transportfahrzeuge, engl. *Automated Guided Vehicles* (AGVs) zum Einsatz. Der Hauptzweck ist eine Kostenreduktion im Personalbereich, aber auch die Reduktion der Transportdauer und die Schonung von Material [Flämig, 2015]. Zur Lokalisierung und Navigation kommen Verfahren aus dem Bereich der *globalen* und der *geführten Navigation* zum Einsatz.

Indoor Die Gruppierung mehrerer AGVs und der zugehörigen Infrastruktur, wie einer zentrale Leitsteuerung, Kommunikationsinfrastruktur oder Lokalisierung, wird als Fahrerloses Transportsystem (FTS) bezeichnet und vielfach in Industrie- und Lagerhallen verwendet. Der Einsatz geht in Amerika bis in die 1950er Jahre, in Deutschland bis in die 1960er Jahre zurück. Der Begegnungsfall zwischen AGV und Mensch ist in diesem Anwendungsfall von vornherein eingeplant.

Outdoor In der innerbetrieblichen Logistik wird – im Gegensatz zum Einsatz automatisierter bzw. autonomer Fahrzeuge im öffentlichen Straßenverkehr – keine Straßenzulassung benötigt, sondern nur eine Maschinenzulassung. Durch die geringere Verkehrsdichte, eine oftmals niedrige Geschwindigkeit im Bereich von 15 bis 20 km/h und die organisatorische Beschränkung der Verkehrsteilnehmer auf einen informierten Personenkreis lässt sich zusätzlich die Komplexität reduzieren. Dadurch eignen sich Betriebshöfe sehr gut als Testfeld.

Hafenlogistik und Containerumschlag In Häfen kommen seit den 1990er Jahren AGVs für das fahrerlose Umsetzen von Containern zwischen Schiff bzw. Containerbrücke, Containerlager und Bahn oder LKW für den Weitertransport zum Einsatz. Bei elektrisch angetriebenen AGVs erfolgt auch der Ladevorgang respektive der Batteriewechsel automatisiert.

Neben Häfen sind auch größere Firmengelände lukrative Einsatzgebiete für die Automatisierung, beispielsweise nennt [BASF, 2017] die um Faktor 20 schnellere Tankcontainerumsetzung im Werk Ludwigshafen.

Wechselbrückenumschlag Auf Betriebshöfen von Speditionen und Logistikdienstleistern findet ein intensiver Umschlag von Wechselbrücken – LKW-Aufbauten, die zum Be- und Entladen auf ausklappbaren Füßen abgestellt werden können – und Sattelaufliegern statt.

Hier gibt es vielfältige Ansätze von Fahrerassistenz [StreetScooter, Department of Autonomous Logistics, 2020] bis hin zu vollautomatisierten Systemen wie dem ZF Innovation Truck“ oder dem in Abb. 2.5b gezeigten KAMAG-Fahrzeug „E-Wiesel AGV“ [Kuther, 2017].

Schlepper und LKWs Eine andere Art der AGV sind autonome Schlepper mit mehreren Anhängern, beispielsweise der „TractEasy“ von Smart Airport Systems [TLD Group, 2018]. Mit der Entwicklung und Erprobung autonomer, digitaler Nutzfahrzeuge für Flughäfen beschäftigte sich auch das vom Bundesministerium für Wirtschaft und Energie (BMWi) geförderte Projekt „AirPortMover“ [Ibeo AS, 2019].

Im Projekt *AutoTruck* [Burkert, 2018] wurden bis 2019 Möglichkeiten des autonomen Fahrens in der Logistik untersucht worden. Als Hauptprobleme werden Pendelfahrten zwischen zwei oder mehr Positionen betrachtet, die auch über öffentliche Straßen führen können, sowie automatische Rangierfahrten mit Andockmanövern innerhalb von Logistikzentren.

2.3.4 Arbeitsmaschinen auf Flughäfen

Ein weiteres Einsatzfeld automatisierter bzw. autonomer Fahrzeuge befindet sich auf Flughäfen. Zu nennen sind hier einerseits Mäharbeiten auf den weitläufigen Grünflächen, andererseits die Flugfeldreinigung und das Schneeräumen.

Zum Beispiel treten seit 2011 in der jährlichen „ION Autonomous Snow Plow Competition“ [The Institute of Navigation, 2023] studentische Teams vornehmlich aus den USA mit selbst entwickelten autonomen Schneeräum-Robotern gegeneinander an. Dieses Event wird von ION (The Institute of Navigation) gesponsort und dient dem Transfer von theoretischem Wissen auf ein echtes Problem aus der Praxis.



(a) Autonomes Traktorkonzept mit integriertem Anbaugerät. Quelle: John Deere
 (b) Cat 793F CMD Mining Truck. Quelle: Caterpillar
 (c) Autonomer Bagger und autonomer Traktor. Quelle: [Fraunhofer IOSB, 2021]

Abbildung 2.6:

Autonome Fahrzeuge in der Landwirtschaft, im Bergbau und im Baubereich.

In jüngerer Zeit hat sich der Fokus verschoben, und es werden industrieseitig echte Prototypen wie in Abb. 2.5c gezeigt erprobt [Daimler Communications, 2017, Ball, 2018], teils bis zur automatisierten Betankung [Valtra Inc., 2019].

2.3.5 Landwirtschaftliche Nutzung

Im Agrarsektor wird seit mehreren Jahrzehnten an der Automatisierung gearbeitet, dies ist unter den Begriffen *Precision Farming* und *Computer-Aided Farming* bekannt. Durch die Verringerung der Überlappung in der spurbasierten Bearbeitung des Feldes sowie durch die Optimierung der Wendemanöver sinken Bearbeitungszeit und -kosten. *Site Specific Farming* [Bongiovanni und Lowenberg-Deboer, 2004] hilft, anhand von präzisen Karten den Dünger- bzw. Pflanzenschutzmitteleinsatz zu senken. Weit verbreitet ist in dieser Domäne die *globale Navigation* mithilfe von GNSS, aber auch die *geführte Navigation* anhand des Bewuchses. Während das automatisierte Fahren schon nutzbar ist, liegen Schwierigkeiten noch in der Beurteilung der Arbeitsqualität [Feuerborn, 2022]. Es werden einerseits klassische Traktoren automatisiert, aber auch neue Fahrzeugkonzepte wie in Abb. 2.6a untersucht.

2.3.6 Anwendungen in der Bauwirtschaft und im Bergbau

Die Gründe für die Automatisierung sind auch im Bergbau vielfältig: Personalkostenreduktion, Gefahrenreduktion durch den Wegfall der menschlichen Fehlerquellen, Effizienzsteigerungen durch 24/7-Betrieb, sowie reduzierter Kraftstoffverbrauch und Reifenverschleiß. [Bird et al., 2019] gibt einen aktuellen Überblick zur verfügbaren autonomen Ausrüstung für den Bergbau.

Australische Minen sind Vorreiter bei der Automatisierung im Tagebau: seit Ende der 1990er Jahre wurden Autonome Transportsysteme, engl. *Autonomous Haulage Systems* (AHSs), große automatisierte Muldenkipper [Reimann, 2017] eingesetzt (siehe Abb. 2.6b), seitdem ist eine fortlaufende Steigerung der Automatisierung zu beobachten. Rio Tinto liegt mit über 80 Mio Meilen mit Abstand auf Platz 1, vgl. Abb. 2.3).

Deutsche Betreiber sehen u. a. wegen des Bergbaurechts noch keine Einsatzmöglichkeiten [Stumpe, 2016]. Bei nahezu allen Fahrzeugtypen basiert die Lokalisierung auf RTK-GNSS. Zur Wahrnehmung der Umgebung, v. a. zur Hinderniserkennung auf der Fahrspur, werden LiDAR- und Radar-Sensoren verwendet.

Die im Tagebau gängigen GNSS-basierten Lokalisierungssysteme sind im Untertagebergbau nicht verwendbar. Jacob und Triebel [2002] nennen die Lokalisierung und Spurführung mithilfe von Kameras und an der Decke montierten Lichtschienen. In neueren LHD-Systemen (Fahrlader, engl. *Load Haul Dump*) der schwedischen Anbieter Sandvik AB [2018] oder Volvo [2016] kommt für die Lokalisierung LiDAR-Sensorik zur Stützung der Odometrie zum Einsatz.

Während im Bergbau autonome Fahrzeuge schon seit Jahren an der Tagesordnung sind, findet man in der Bauwirtschaft derzeit vor allem Prototypen, wie einen autonomen Unimog für das Gelände [Wolf et al., 2018], einen Bagger für die Altlastensanierung [Fraunhofer IOSB, 2021] (siehe Abb. 2.6c), oder neue LKW-Konzepte [Jansen, 2016].

2.3.7 Militärische Nutzung

Militärische Verwendungszwecke für autonome Fahrzeuge sind nicht nur der in Abschnitt 2.3.2 betrachtete Konvoi bzw. Platoon, wie bspw. in den TARDEC-Projekten oder TULF und StrAsRob demonstriert, sondern wesentlich vielfältiger. Über die Robotik-Aktivitäten im deutschen Militär gibt [Smolarek, 2019] einen guten Überblick.

Ein Teil der militärischen Systeme sind kleine teleoperierte und teilautonome Aufklärungsroboter sowie Entschärfungsroboter für den EOD-Bereich (explosive ordnance disposal) – vom kleinen Roboter der 20 kg-Klasse bis hin zum unbemannten Kleinpanzer des „Route Clearance System“.

Ein großer Anwendungsbereich ist der Lastentransport als *Mule*, der einige hundert Kilo Nutzlast für den Soldaten befördern kann. Diese Systeme verfügen meist über die Funktion, einer Person auch durch das Gelände zu folgen, oder auf Basis von Wegpunkten autonom eine Strecke abzufahren. Die US-Army hat seit 2020 über 400 S-MET-Fahrzeuge (Small Multipurpose Equipment Transport) bei General Dynamics Land Systems bestellt, erste Testsysteme sind ausgeliefert [Keller, 2022]. Europäische Anbieter in diesem Bereich sind Rheinmetall mit der 2019 übernommenen kanadischen Provectus Robotics Solutions – mit deren auf LiDAR-Sensoren und Kameras basierter Lösung das Mule-Szenario der M-ELROB 2018 gewonnen wurde, oder Diehl Defence mit dem Ziesel.

In einer Vielzahl an Entwicklungsprojekten ist Milrem Robotics aus Estland vertreten, die ihre Plattform THeMIS für unterschiedliche Anwendungen anpassen, vom Feuerlöschroboter über den Mule im Projekt iMUGS (integrated Modular Unmanned Ground System) [Heiming, 2021] bis hin zur autonom oder teleoperiert fahrenden Waffenstation.

Am oberen Ende der Größenskala ist MGCS (Main Ground Combat System) zu nennen, das geplante deutsch-französische Nachfolgesystem für die Kampfpanzer Leopard II und Leclerc. Dieses soll als System aus einem bemannten schweren Kampfpanzer und mehreren unbemannten kleineren Panzern ausgelegt werden, wobei mindestens letztere auch autonom im Gelände fahren sollen [Dean, 2023].

Hier lässt sich noch einmal ein Blick auf die ambitionierten Ziele aus dem *National Defense Authorization Act for Fiscal Year 2001* [106th Congress of the United States of America, 2000] werden, bis zum Jahr 2015 ein Drittel der Landfahrzeuge des US-Militärs unbemannt zu betreiben. Dies wurde laut Markoff [2013] nicht erreicht, aber mittlerweile – 10 Jahre später – sind Produkte für viele Anwendungszwecke am Markt verfügbar.

2.3.8 Start-Ups und Tech-Konzerne in den USA

Die erfolgreichen Teams der DARPA Grand Challenges waren nach den Wettbewerben ein Pool an Talenten. Aus den beiden besten Urban Challenge-Teams – Team CMU und Team Stanford – wurden die wichtigsten Personen zum *Google Self-Driving Car Project* (mittlerweile *Waymo*) geholt, wie in Abbildung 2.7 dargestellt ist. Einige sind noch dort, andere haben selbst Start-Ups in diesem Bereich gegründet. Aus einigen dieser Start-Ups sind längst Tech-Konzerne geworden, andere wurden geschluckt.

Die Aktivitäten lassen sich grob in zwei Bereiche einteilen: Fahrdienstleister mit Robotaxis und Technologieanbieter für Fahrzeughersteller.

Robotaxis Im Bereich der Robotaxis ist zuerst *Waymo* zu nennen, das 2009 als *Google Self-Driving Car Project* von Sebastian Thrun (Stanford Racing Team) gegründet wurde. *Waymo* entwickelt und betreibt den als *Waymo One* bezeichneten Robotaxi-Dienst in einigen Städten in den USA mit umgerüsteten Serienfahrzeugen [Eckl-Dorna, 2018]. Hier ist in einigen Bundesstaaten wie Arizona oder Kalifornien nicht nur eine Erprobung mit oder ohne Sicherheitsfahrer zugelassen, sondern teilweise auch ein kommerzieller Taxi-Dienst möglich [Nordenbrock, 2022].

Der nächstgrößte Anbieter ist *Cruise Automation* (teilweise zu General Motors gehörig) [Bellan, 2021], der vor allem in San Francisco testet und teilweise kommerzielle Taxidienste anbietet [Lee, 2023].

Die Amazon-Tochter *Zoox* betreibt ebenfalls Robotaxis auf öffentlicher Straße, ist jedoch erst allmählich auf dem Weg vom auf eigene Mitarbeiter eingeschränkten Nutzerkreis hin zum kommerziellen Betrieb [Dent, 2023].

Auf der Seite der Fahrdienstleister sind noch Uber und Lyft zu nennen, die zwischenzeitlich ebenfalls Expertise im autonomen Fahren aufgebaut oder zugekauft [Thompson, 2015] hatten. Deren Entwicklungsabteilungen wurden mittlerweile an Aurora [Korosec, 2020] bzw. Toyota [Doll, 2021] verkauft, der Fokus liegt nun eher auf der Integration von Robotaxis in die eigenen Buchungsplattformen.

The DARPA Self-Driving Diaspora

The members of the top two teams followed from the 2007 contest are now leading startups at the forefront of robotic driving.

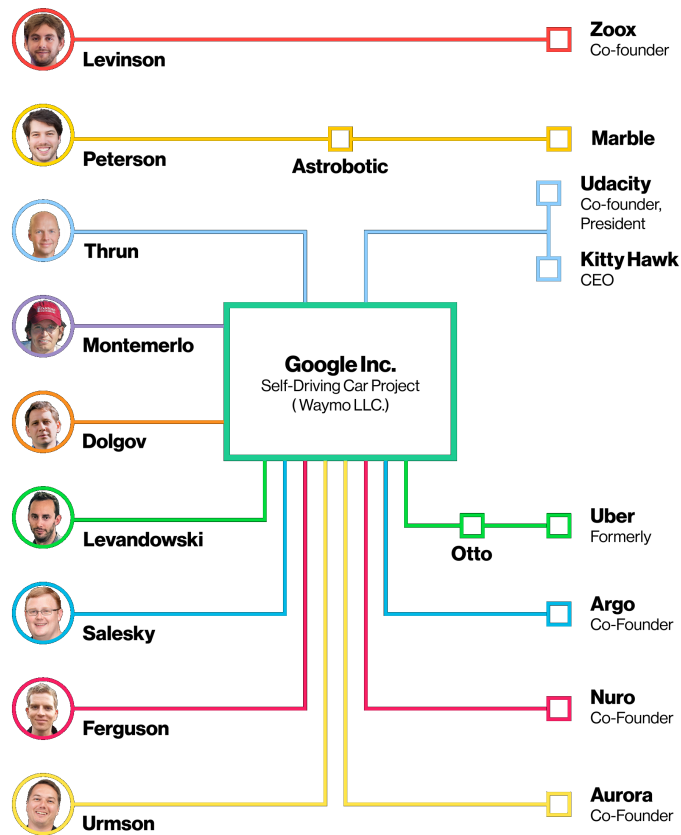


Abbildung 2.7:

Personalentwicklung nach der DARPA Urban Challenge, mit Google im Mittelpunkt. Quelle: [Hull, 2017].



(a) Waymo-Robotaxi der 5. Generation – Jaguar I-PACE. Quelle: Waymo

(b) Automatisierter LKW. Quelle: Aurora

Abbildung 2.8: Flotten- und Versuchsfahrzeuge der US-Start-Ups.

Die Robotaxis unterscheiden sich allesamt von „normalen“ Serienfahrzeugen dadurch, dass sie an der Außenhaut, meist auf dem Dach und an den Kotflügeln, mit vielen zusätzlichen Sensoren ausgestattet sind (siehe Abb. 2.8a). Hier kommen fast überall rotierende LiDAR-Sensoren zum Einsatz.

Die Robotaxis werden primär als Level-4-Fahrzeug ohne Sicherheitsfahrer betrieben: im Zweifels- oder Fehlerfall bremst das System in den Stand, und ein Operator muss per Teleoperation unterstützen. Aus Sicht der Stadt San Francisco, wo viele Anbieter ihre Robotaxis testen und betreiben, Sorge dies für Behinderungen in den Hauptverkehrszeiten, die es zunächst durch zeitliche Einschränkungen der Betriebszeiten zu reduzieren gelte, bis die Systeme weiter ausgereift seien [Petereit, 2023].

Einige Zeit sah es so aus, als ob die US-amerikanischen Tech-Konzerne den Erstausrüster, engl. *Original Equipment Manufacturers* (OEMs) weit überlegen wären und quasi sofort Level-4- und Level-5-Systeme bereitstellen könnten, womit die klassischen Automobilhersteller – auch als OEMs bezeichnet – zum reinen Fertigungspartner der Robotaxi degradiert würden. Einen ersten großen Knick gab es 2018 nach einem tödlichen Unfall eines mit Sicherheitsfahrer fahrenden Uber-Fahrzeugs, nachdem alle Anbieter den Fokus stärker Richtung Sicherheit verschoben [Lee, 2023]. Danach nahmen Entwicklung und Optimismus wieder zu, werden aber unterschiedlich beurteilt:

Die Softwareriesen, die seit Jahren Milliardensummen in die Entwicklung pumpen, stoßen an technologische Grenzen. Das voll autonome Fahren lässt sich nicht so schnell entwickeln wie ursprünglich gedacht. In der Breite funktionieren die sogenannten Robotaxis noch nicht. (Tyborski und Hubik [2021])

Nach [Lee, 2023] sei der Großteil der Anbieter bei weitem nicht profitabel, so würden Cruise und Waymo jährlich jeweils mehr als 1 Milliarde US-Dollar Verlust machen. Allerdings sei das Produkt mittlerweile für den kommerziellen Einsatz geeignet – herausfordernd sei nur die Skalierung: wie viele Taxifahrten müssen mit welcher Anzahl Robotaxis durchgeführt werden, bis die bisherigen Investitionen wieder eingespielt sind?

Technologieanbieter Im Bereich der Technologieanbieter seien beispielhaft Argo AI, Aurora, Nvidia, Mobileye und Qualcomm genannt.

Der Automobilhersteller Ford investierte seit 2017 in die 2016 gegründete *Argo AI* als Anbieter von Hard- und Software für autonomes Fahren, 2020 investierte Volkswagen ebenfalls und fusionierte seine Tochter AID mit Argo AI [Boston und Colias, 2019]. Aufgrund von Milliardenverlusten wurde Argo AI im Herbst 2022 aufgegeben [Conrad und Leicht, 2022].

Aurora Innovation wurde 2017 u. a. von Chris Urmson (CMU Tartan Racing, danach Google) gegründet und sieht sich als primär als Technologielieferant mit Fokus auf Software und nicht als Fahrzeug- oder Sensorhersteller [Hecking, 2019]. Neben Partnerschaften mit unterschiedlichen PKW-Herstellern wie Hyundai oder Fiat Chrysler

[Dzikiy, 2019] liegt der Schwerpunkt im Bereich von Autonomie-Kits für LKWs (siehe Abb. 2.8b), erste kommerzielle Einsätze sind für 2024 geplant [Bursztynsky und McCracken, 2023]. Zusammen mit dem Automobilzulieferer Continental soll bis 2027 die Autonomie als HaaS-Modell (engl. *Hardware as a Service*) für Spediteure angeboten werden [Pillau, 2023a].

Als weiterer Technologieanbieter sei hier *Nvidia* genannt. Die Firma hat ihre Wurzeln im Bereich der Grafikkartenentwicklung für PCs und hat sich über auf KI-Anwendungen spezialisierte Rechner sowohl für Rechenzentren als auch für den Automotive-Einsatz hin zum Technologiepartner für autonomes Fahren entwickelt. Neben Simulationssystemen wird mit Nvidia DRIVE auch ein Framework für Level-3- bis Level-4-Anwendungen für die Integration bei Fahrzeugherstellern angeboten.

Die 1999 gegründete israelische Firma *Mobileye*, seit 2017 eine Tochter von Intel, gilt als der führende Anbieter von Software für kamerabasierte ADAS (Fahrerassistenzsystem, engl. *Advanced Driver Assistance System*) und ist bei vielen OEMs vertreten. Mobileye entwickelt unterschiedliche Plattformen für Level-2- bis Level-4-Funktionen, dort dann auch mit Radar- und LiDAR-Sensoren.

Der US-amerikanische Halbleiterhersteller *Qualcomm*, bereits länger als Hardwarelieferant für ADAS-Systeme im Geschäft, hat sich durch die Übernahme der Fahrerassistenz-Sparte der schwedischen Firma Veoneer [Gelowicz, 2021] zum Technologieanbieter weiterentwickelt. Seine Snapdragon-Ride-Plattform kommt bei unterschiedlichen OEMs für die Entwicklung von Level-2- bis Level-4-Funktionen zum Einsatz [Bellmer, 2023].

2.3.9 Autohersteller und große Zulieferer

Im Gegensatz zu den US-amerikanischen Tech-Konzernen, die direkt mit dem Ziel der vollautomatisierten Systeme entwickeln, kommen die OEMs aus dem ADAS-Bereich, in dem bereits Geld verdient wird, und arbeiten schrittweise an der Erhöhung der SAE-Level. Während die ADAS-Systeme bis in die 2010er Jahre fast immer inklusive der Sensorik von Zulieferern wie Bosch, Delphi/Aptiv oder Continental eingekauft wurden, geht der Trend zum Aufbau eigener Entwicklungskapazitäten, aber auch sehr stark zu Entwicklungspartnerschaften, um möglichst schnell hohe Automatisierungsgrade zu erreichen. Dies dient einerseits dem Ziel, diese sog. Ownership-Fahrzeuge an Endkunden zu verkaufen, aber langfristig auch, um selbst mit automatisierten Fahrzeugen zum MaaS-Anbieter (engl. *Mobility as a Service*) zu werden und weitere Erlösquellen zu erschließen. Nach aktuellen Erkenntnissen wird aber die Marge der OEMs massiv sinken, so müsse Mercedes-Benz 40 % des Umsatzes mit Softwarefunktionen an den Technologiepartner Nvidia weiterreichen [Hubik und Hofer, 2022].

Ferres und Klein [2019] zeigen die Verflechtungen zwischen OEMs, Zulieferern, Tech-Konzernen und Start-Ups zum Zeitpunkt 2019 in vielfältiger Weise grafisch auf. In Europa konsolidiert sich die Branche derzeit, viele Kooperationen der großen OEMs untereinander und mit weiteren Partnern sind bereits wieder Geschichte:

Die große Euphorie der Tech-Szene sowie der Autohersteller für das autonome Fahren hat sich mittlerweile gelegt. Technische Fortschritte werden nicht in der vorausgesagten Geschwindigkeit erreicht, zudem gibt es nach wie vor Hürden bei den Zulassungsbestimmungen in einer Vielzahl von Ländern. Autonomes Fahren nach Level 4, also mit Sicherheitsfahrer an Bord, ist aktuell nur im Rahmen von Tests im öffentlichen Nahverkehr auf exakt begrenzten Routen oder auf abgesperrtem Terrain zu sehen. (Conrad und Leicht [2022])

Im Folgenden wird eine kurze Zusammenfassung der Aktivitäten rund die OEMs und großen Zulieferer gegeben, mit dem Fokus auf Deutschland. Der Schwerpunkt liegt dabei auf der Entwicklung autonomer Fahrfunktionen für die sog. Ownership-Fahrzeuge, deren Sensorik im Vergleich zu den Robotaxis möglichst perfekt, d. h. unsichtbar, in die Karosserie integriert werden muss.

Tesla 2015 wurde von Tesla der „Autopilot“ vorgestellt, ein zusammen mit Mobileye entwickeltes Level-2-System auf Basis von Kamera- und Radar-Sensoren. Nach einem tödlichen Unfall 2016 wurde die Zusammenarbeit beendet und Tesla entwickelte eine Eigenlösung, später wurden hier Radar- und Ultraschallsensoren eingespart. Auch das aktuell als „Full Self Driving“ beworbene System ist weiterhin nur Level 2 [Honsel, 2023].

Insgesamt wird dieses System trotz der positiven Selbstvermarktung eher negativ gesehen. Der Autopilot benötige eine hohe Aufmerksamkeit des Fahrers und häufige Eingriffe, um kein regelwidriges Verhalten zu zeigen [Barry, 2019]. Über Unfälle im automatisierten Betrieb – teils durch zu hohe Interpretation des Begriffs Autopilot – wird schon seit Jahren berichtet [Yadron und Tynan, 2016], aktuellen Recherchen zufolge soll der Autopilot für zahlreiche Unfälle verantwortlich sein [Bolliger, 2023].

Im Gegensatz zu den meisten anderen OEMs stellt Tesla Software- und Funktionsupdates hochfrequent und „Over-the-Air“, d. h. ohne Werkstattbesuch, bereit.

General Motors Das Level-2-System „Super Cruise“ von General Motors kommt in diversen Konzernmarken wie Cadillac, GMC oder Chevrolet zum Einsatz und basiert auf Kamera- und Radar-Sensoren. Die Funktion ist auf vorher kartierte und baulich getrennte Highways limitiert. In US-amerikanischen Vergleichstests wurde „Super Cruise“ mehrfach besser als der Tesla „Autopilot“ bewertet [Olsen, 2018, Monticello, 2020], wobei der Fokus allerdings nicht nur auf der technischen Performance lag, sondern auch die Einbindung des Fahrers einfluss.

Die Weiterentwicklung „Ultra Cruise“ wurde gemeinsam mit Qualcomm vorangetrieben und integriert zusätzlich einen LiDAR-Sensor. Ab Ende 2023 soll hiermit eine Level-2-Assistenz auf 95 % aller befestigten Straßen der USA ermöglicht werden [Hawkins, 2023].

Die deutschen OEMs sind hier konservativer in der Vermarktung, aber teils weiter in der Zulassung mit Level 3.

BMW Im Vergleich zu den anderen deutschen Automobilherstellern hat BMW schon in den 2010er Jahren eine große Abteilung für die Softwareentwicklung von Assistenz- und Autonomiefunktionen aufgebaut, um unabhängiger von Zulieferern zu werden.

Aus der 2017 gestarteten Kooperation mit Mobileye (Vision) und Intel (Hardware) sowie weiteren Zulieferern und Entwicklungspartnern sind vor allem Level-2-Systeme hervorgegangen, allerdings bisher noch nicht das für 2021 angekündigte Level-3-System im BMW iX. Für Neuentwicklungen von Level-2-Plus- und Level-3-Systemen wechselte BMW die Partner, hier gibt es eine Kooperation mit Qualcomm und Arriver [Floemer, 2022], langfristig wird auch hier Level 4 angestrebt.

Mercedes-Benz & Daimler In aktuellen S-Klasse- und EQS-Fahrzeugen (siehe Abb. 2.9a) wird der „Drive Pilot“ [Mercedes-Benz Research & Development North America, Inc., 2019] angeboten, ein Level-3-System für die Verwendungen auf ausgewählten Autobahnabschnitten bis 60 km/h, welches eine Straßenzulassung für Deutschland besitzt und als ersten Seriensystem mit US-Zulassung (in Nevada [Pillau, 2023b] und Kalifornien [Krempf, 2023]) gilt.

Im 2017 gestarteten Athena-Projekt von Mercedes-Benz und Bosch sollten innerhalb weniger Jahre Robo-Taxis auf die Straße gebracht werden, bereits 2021 wurde diese Kooperation wieder beendet, ohne das Ziel erreicht zu haben [Kunkel, 2021].

Eine Zusammenarbeit von BMW und Mercedes-Benz [BMW Group, 2019] für die Entwicklung von ADAS sowie dem automatisierten Fahren auf Autobahnen und für automatisierte Parkfunktionen (jeweils bis hin zu Level 4 für Serienstart 2024) wurde bereits 2020 aus Kostengründen wieder auf Eis gelegt [Hubik, 2021]. Stattdessen wurde eine Entwicklungspartnerschaft mit Nvidia für Level-3 und Level-4-Funktionen mit Serienstart 2024 eingegangen [Reichel, 2020a], hier soll auch ein neuer LiDAR-Sensor von Luminar zum Einsatz kommen.

Im Nutzfahrzeugbereich bei Daimler wird ebenfalls Level 4 angestrebt, hier soll das von Torc Robotics übernommene Know-How helfen [Daimler Trucks, 2019].

Volkswagen & Audi Der Audi A8 wurde speziell für hochautomatisiertes Fahren in Stausituationen entwickelt und sollte das erste Level-3-Fahrzeug im öffentlichen Straßenverkehr werden [Ross, 2017], geplant war eine nachträgliche Freischaltung der Funktionen für 2018 – 2020 wurde dies aus technischen und rechtlichen Gründen abgesagt [Schmidt, 2020].

Ab 2020 wurden die Softwareentwicklungsaktivitäten aus allen Marken des Volkswagen-Konzerns in die neu gegründete Tochtergesellschaft CARIAD verschoben und mit externen Zukäufen verstärkt, allerdings konnten die geplanten Ziele bisher nicht erreicht werden [Schesswendter, 2023]. Die 2022 gestartete Allianz aus CARIAD und Bosch [Bosch und CARIAD, 2022] arbeitet an Level-3-Fahrzeugen, perspektivisch nach dem Ende von Argo AI auch an Level 4, erste Testfahrzeuge für Level 3 sollen 2023 in Betrieb gehen [Floemer, 2023], als Hardwarelieferant ist Qualcomm

involviert. Ebenfalls als Technologiepartner involviert ist die Intel-Tochter Mobileye [Hanley, 2023], unter anderem Konzerntochter Porsche arbeitet mit Mobileye an Level-2-Systemen [Linden, 2023].

Volkswagen will mit der Ride-Sharing- und Robotaxi-Tochter *Moia* in Hamburg bis 2025 das erste typenzertifizierte autonome MaaS-System Europas auf die Straße bringen. Auf dem Weg wurden mehrere Technologiepartnerschaften geschlossen und wieder aufgegeben, anfangs mit *Aurora* [McGee, 2019], dann mit Argo AI [Conrad und Leicht, 2022]. Stand 2023 wird das System mit Apex.AI weiterentwickelt [Pertschy, 2023].

Weitere OEMs Jaguar Land Rover war in den letzten Jahren eher in Randbereichen aktiv, bspw. bei Assistenz- und Autonomiefunktionen für den Geländeeinsatz [Jaguar Land Rover, 2016]. 2023 hat die Firma die Aktivitäten massiv ausgeweitet und will ab 2025 alle neuen Modelle mit der vom Technologiepartner gelieferten Lösung Nvidia DRIVE ausstatten, um Level-3-Funktionen zu ermöglichen [Prawitz, 2022]

Ford hat 2017 mit dem Ziel von Level-4-Systemen eine Mehrheitsbeteiligung an Argo AI erworben [Ford Motor Company, 2017]. Da die angestrebten Ziele nicht erreicht wurden, wurden Argo AI [Conrad und Leicht, 2022] und damit vorerst die Level-4-Aktivitäten aufgegeben.

Volvo gründete 2017 zusammen mit dem Zulieferer Autoliv die Firma Zenuity, die auf Basis von Nvidia-Hardware bis 2021 Level-4-Funktionen in die Serie bringen sollte. Nach einigen Umstrukturierungen geht die Zusammenarbeit mit Nvidia weiter, das für Anfang 2024 geplante Fahrzeug EX90 soll mit Nvidia DRIVE und Level-3-Funktionen ausgestattet werden [Shapiro, 2022] und über einen LiDAR-Sensor von Luminar verfügen [Wittich et al., 2023].

General Motors hat 2016 für 60 Mio US-Dollar das Start-Up Cruise Automation übernommen (siehe Abschnitt 2.3.8).

Die Toyota-Tochter Woven Planet hat 2021 die Entwicklungsabteilung „Level 5“ des Fahrdienstleisters Lyft übernommen [Doll, 2021]. Nach eigenen Aussagen treibt man die Entwicklung mit dem Fokus auf Vision-Only voran.

Die großen Zulieferer wie Bosch, Continental oder ZF sind einerseits als Technologiepartner mit einzelnen OEMs verbunden. Andererseits legen sie klassisch den Fokus auf Eigenentwicklungen, um diese an diverse OEMs verkaufen zu können.

2.3.10 Urbane Mobilität

Unter dem Oberbegriff der urbanen Mobilität werden zwei Fahrzeugkategorien zusammengefasst, die in den letzten Jahren einen massiven Hype erfahren. Zum einen sind dies die *Robotaxis*, also autonome Fahrzeuge für den Transport von ca. 4 bis 5 Personen, welche in den vorherigen Abschnitten erwähnt wurden. Ihnen ist gemeinsam, dass sie v. a. im innerstädtischen Bereich zum Einsatz kommen.



(a) Mercedes EQS mit Level-3-Funktionen. Quelle: Mercedes-Benz



(b) EVA-Shuttle in Karlsruhe, ein Easymile EZ10 mit projektspezifischer Sensorik. Quelle: Paul Gärtner

Abbildung 2.9: Fahrzeuge von OEMs und People-Mover.

Zum anderen sind es sog. *People-Mover*, autonome Kleinbusse für 10–15 Personen, welche seit circa 2016 an vielen Orten erprobt werden.

Ebenfalls in die Klasse der urbanen Mobilität gehören die sog. *Cargo-Mover*, autonome Fahrzeuge zur Güterbeförderung, auch wenn hier der Übergang zu Fahrzeugen aus der innerbetrieblichen Logistik (siehe Abschnitt 2.3.3) fließend ist. [Sokolov, 2020] arbeitet als Fazit aus dem Automated Vehicles Symposium 2020 heraus, dass der autonome Gütertransport aufgrund geringerer Sicherheits- und Komfortanforderungen sowie eines besseren Geschäftsmodells eher in Serie als der Personentransport gehen würde.

In einem Interview mit der Süddeutschen Zeitung geht Torsten Gollewski, Leiter Vorentwicklung der ZF Friedrichshafen AG, auf die Besonderheiten von Shuttle-Fahrzeugen ggü. autonomen PKW insbesondere im Bereich der Zulassung ein:

Wir haben mittlerweile alle technischen Voraussetzungen, um urbane Mobilität neu zu gestalten: Die Sensoren, die Rechenleistung, die Künstliche Intelligenz. Und wir haben in Deutschland eine neue und weltweit führende Gesetzgebung, die autonome Shuttles (Level-4) im Serieneinsatz in der Stadt erlaubt. Deshalb hat das Interesse von Mobilitätsdienstleistern, Flottenbetreibern und Zustelldiensten an dieser Technologie nicht nachgelassen, im Gegenteil. [...] Wir arbeiten mit den Städten und ihrer Infrastruktur eng zusammen. Wenn man eine festgelegte Route und eine abgetrennte Spurführung für Shuttle-Fahrzeuge hat, lässt sich die Komplexität des autonomen Fahrens entscheidend reduzieren – und wir erfüllen die Wünsche unserer Kunden: Die wollen nämlich am Stau vorbeifahren. (Becker [2021])

International, aber auch an vielen Orten in Deutschland werden Fahrzeuge unterschiedlicher Hersteller in verschiedenen Umgebungen erprobt.

Der EZ10 des französischen Herstellers EasyMile ist derzeit das am weitesten verbreitete Fahrzeug und existiert in mehreren Generationen. Es nutzt vor allem GNSS

und Odometrie zur Lokalisierung sowie LiDAR- und Radar-Sensoren zur Hindernis- und Objekterkennung. Im Regelfall erfolgt die Fahrzeugführung zentimetergenau in derselben Spur. In vielen Projekten kommt noch eine Begleitperson zum Einsatz, die im Zweifelsfall eingreifen und das Fahrzeug steuern kann, zum Beispiel auf dem EUREF-Campus in Berlin-Schöneberg (2017-2018), seit Ende 2017 in Bad Birnbach oder seit 2020 in Monheim am Rhein. Auf dem GreenTEC-Campus in Schleswig-Holstein wurde zum ersten Mal ein Level-4-Betrieb erprobt, die Überwachung und ggf. Fernsteuerung erfolgt von außen [Reichel, 2020b] – aufgrund der zum damaligen Zeitpunkt gültigen Gesetze nur auf Privatgelände. Für einzelne Forschungsprojekte kommt der EasyMile EZ10 auch mit modifizierter Sensorik und Autonomiefunktionen von Partnern zum Einsatz [FZI Forschungszentrum Informatik, 2019] (Abb. 2.9b).

Ein weiterer französischer Anbieter ist Navya, dessen Fahrzeug Arma u. a. in Frankreich und der Schweiz erprobt wird.

Die ZF-Tochter 2getthere betreibt seit den 1990er Jahren (vgl. Abschnitt 2.1) People-Mover, in den meisten Fällen auf *segregated lanes*. Die Entwicklung ging in jüngeren Projekten hin zum Mischverkehr und damit zu wesentlich höheren Anforderungen an die Sensorik und Software [Wildemann, 2019].

Daneben gibt es noch viele weitere Projekte im Bereich der People-Mover, von denen aber oft nur Einzelfahrzeuge existieren, wie z. B. das Projekt HEAT der Hamburger Hochbahn, welches von 2019 bis 2021 erprobt wurde [IAV GmbH, 2020].

Weiterhin gibt es am Reißbrett neu entwickelte Konzeptfahrzeuge, die eine Bewegungsplattform (teuer, mit Sensorik ausgestattet, hohe Auslastung angestrebt) mit unterschiedlichen Wechsellaufbauten (günstiger, verbleiben oft für Stunden am selben Ort) verbinden, z. B. das MetroSnap-Fahrzeug [Schmidt, 2019] oder das U-Shift-Projekt rund um das DLR [Münster et al., 2022]). Ebenso zu nennen ist die im Projekt UNICARagil entwickelte Fahrzeugfamilie mit autoSHUTTLE oder autoCARGO [Keilhoff et al., 2019].

3 Versuchsfahrzeuge

Im Bereich des autonomen Fahrens sind Versuchsfahrzeuge ein essentielles Werkzeug für die Forschung, wenn man sich nicht ausschließlich mit Wahrnehmungsalgorithmen oder der Simulation beschäftigt. Neben der Erprobung des Gesamtsystems im geschlossenen Regelkreis erlauben sie auch die Entwicklung und Erprobung von Teilkomponenten oder dienen zur Aufzeichnung von Messdaten. Die *Forschungsfrage 1* dieser Arbeit beschäftigt sich damit, wie ein Basisfahrzeug geeignet mit Sensorik, Aktorik und Rechentechnik ausgestattet werden sollte, um die Anforderungen für den Forschungsbetrieb bestmöglich zu erfüllen, und dies so zu gestalten, dass das Fahrzeug überhaupt mit einem kleinen Team betrieben werden kann. Vergleicht man dies mit der Automobilindustrie, so beschäftigen sich dort schnell mehr als 1000 Personen mit der Entwicklung automatisierter Fahrfunktionen, und ganze Teams sind nur für Fahrzeugaufbau und -betrieb zuständig – diese Komplexität kann an einem kleinen Uni-Institut nicht geleistet werden.

Im Rahmen dieser Arbeit wurden diverse Versuchsfahrzeuge aufgebaut und für eigene Experimente genutzt, vor allem aber im Hinblick auf eine gute Nutzbarkeit durch das komplette Institut ausgelegt. Dabei handelt es sich einerseits um die beiden institutseigenen Fahrzeuge MuCAR-3 (Munich Cognitive Autonomous Robot Car 3rd Generation) und MuCAR-4 (Munich Cognitive Autonomous Robot Car 4th Generation), die schon länger in der Nutzung sind und bereits mehrere Überarbeitungen erfahren haben, sowie um MuCAR-5 (Munich Cognitive Autonomous Robot Car 5th Generation), der sich derzeit im Aufbau befindet (Abschnitte 3.2 bis 3.4).

Andererseits wurde der Aufbau weiterer Versuchsfahrzeuge durch Industriepartner in diversen Forschungs- und Entwicklungsprojekten beratend begleitet, um die langjährige Erfahrung einzubringen und eine gute Nutzbarkeit der Versuchsfahrzeuge für alle Projektteilnehmer und insbesondere die TAS-Anteile (Institut für Technik Autonomer Systeme) zu erzielen (Abschnitte 3.5 bis 3.8).

Allen diesen Fahrzeugen ist gemeinsam, dass sie auf handelsüblichen PKW bzw. LKW basieren und nachträglich mit Aktorik, Sensorik sowie Rechentechnik ausgestattet worden sind. Ebenso verfügen sie über Fahrer- und Beifahrersitz, um einen menschlichen Sicherheitsfahrer und Operator im autonomen Fahrzeug befördern zu können und um – im Gegensatz zu vielen universitären Selbstbau-Plattformen – unter allen Witterungsbedingungen „im Trockenen“ entwickeln zu können. Als Basisplattform für die MuCARs kommen Geländelimousinen, engl. *Sport Utility Vehicles (SUVs)* zur Verwendung – diese bieten einen guten Kompromiss für Einsatzszenarien zwischen Onroad- und leichtem Offroad-Bereich.

Bevor die einzelnen Versuchsfahrzeuge mit ihrer Hardwarearchitektur vorgestellt werden, werden die Gemeinsamkeiten im Ausstattungskonzept betrachtet. Bei den Versuchsfahrzeugen wird der Schwerpunkt auf MuCAR-3 gelegt, für die weiteren Fahrzeuge werden nur die maßgeblichen Unterschiede aufgeführt.

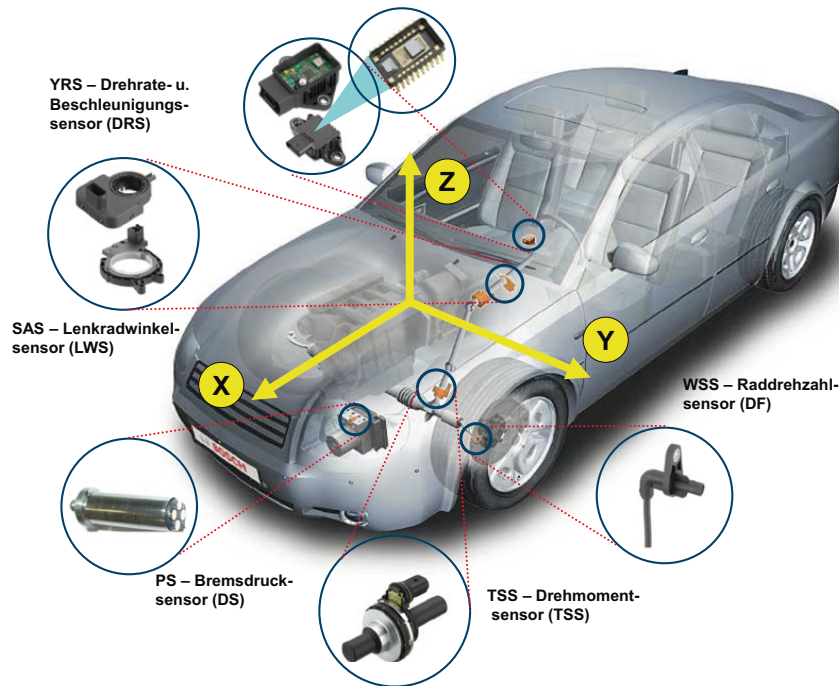


Abbildung 3.1:
Übersicht über die Fahrodynamiksensoren. Quelle: [Mörbe, 2015, S. 229]

3.1 Gemeinsamkeiten in der Ausstattung

Als *Aktorik* werden alle Einbauten bezeichnet, die dem Rechner im autonomen Betrieb die Ansteuerung der Primär- und Sekundärsysteme zur Fahrzeugführung ermöglichen, die ansonsten vom Menschen bedient werden. Diese Aktoren bilden gemeinsam das sog. DbW-System (engl. *Drive-by-Wire*). Die primären Systeme dienen der Bewegung des Fahrzeugs an sich und lassen sich in die Längs- und Querführung (üblicherweise über Gas- und Bremspedal resp. das Lenkrad) unterteilen. Hinzu kommen der Fahrstufenwählhebel und die Feststellbremse. Als Sekundärsysteme zählen die Ansteuerung von Licht, Blinker, Scheibenwischer oder der Hupe sowie Zündung und Motorstart.

Die *Sensorik* autonomer Fahrzeuge gliedert sich im Allgemeinen in drei Kategorien: Fahrodynamiksensoren wie in Abb. 3.1 gezeigt liefern Informationen über die eigene Fahrzeugbewegung [Mörbe, 2015, S. 226ff.]. Lokalisierungssensoren, meist auf Basis von GNSS (Globales Navigationssatellitensystem, engl. *Global Navigation Satellite System*), stellen eine – je nach Empfängertyp und Umgebung – bis auf Zentimeter-Level genaue Position bereit. Umfeldsensoren wie Kameras, LiDAR- (Light Detection And Ranging) oder Radar-Sensoren (Radio Detection And Ranging) dienen dagegen der Wahrnehmung des Fahrzeugumfelds. Die genaue Ausprägung der Sensorsetups unterscheidet sich jedoch stark zwischen den Versuchsfahrzeugen, daher wird sie in den folgenden Abschnitten detaillierter behandelt.

Allen Versuchsfahrzeugen gemein ist eine Unterteilung der Rechentechnik auf zwei Ebenen, einen Low-Level- und einen High-Level-Rechner, die über eine definierte Schnittstelle miteinander kommunizieren.

Der *High-Level-Rechner*, auch als A-Kit oder Autonomie-Kit (dann meist inklusive der Sensoren und Algorithmen) bezeichnet, ist ein leistungsstarker Rechner, welcher über Schnittstellen die Messdaten der Umfeldsensorik empfängt sowie mit dem Low-Level-System kommuniziert. Vor allem dient er zum Betrieb der Algorithmen zur Umfeldwahrnehmung und weiterer Module des autonomen Fahrens bis hin zur Bahnregelung. In den Versuchsfahrzeugen kommen hier im Regelfall handelsübliche Server- oder Workstation-PCs zum Einsatz, die mit einer leistungsstarken CPU ((Haupt-)Prozessor eines Computers, engl. *Central processing unit*) und oftmals zusätzlich mit einer GPU (Grafikprozessor, engl. *Graphics processing unit*) ausgestattet sind.

Der *Low-Level-Rechner*, auch B-Kit oder DbW-Kit genannt, übernimmt die fahrzeugspezifische Ansteuerung der Aktoren, die unteren, extrem fahrzeugabhängigen Reglerebenen, die Kommunikation mit dem fahrzeugeigenen Bussystem sowie die Aggregation von Messdaten aus dem Fahrzeug und dem DbW-System für das High-Level-System. Üblicherweise kommen hier echtzeitfähige Systeme mit einer Vielzahl an analogen, digitalen und Bus-Schnittstellen zum Einsatz. Auf diesem System wird neben der Ansteuerung oft ein Sicherheitssystem implementiert, welches bei Fehl- oder Falschfunktion des High-Level-Systems eine definierte Prozedur ausführt und das Fahrzeug in einen sicheren Zustand überführt.

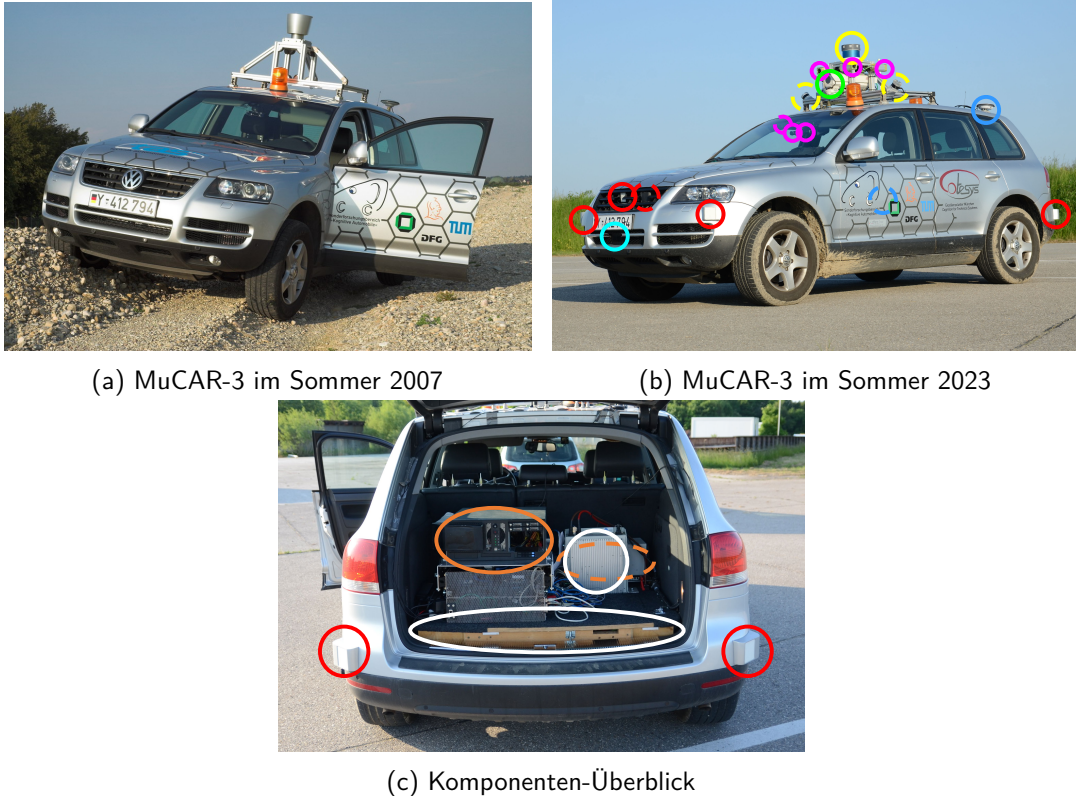


Abbildung 3.2:

Versuchsfahrzeug MuCAR-3. Während 2007 nur ein LiDAR-Sensor auf dem Dach und Kameras hinter der Windschutzscheibe montiert waren, sind bis 2023 weitere Kameras auf dem Dach, aktualisierte LiDAR-Sensorik sowie eine Rundum-Abdeckung mit Radar-Sensoren ergänzt worden: LiDAR Velodyne VLS-128 (gelb), LiDAR Ouster OS0-32 (gelb gestr.), LiDAR Ibeo Lux 8L (cyan), Radar (rot), Fernbereichs-Radar (rot gestr.), RGB-Kameras (magenta), RGB-NIR-Kamera (magenta gestr.), Wärmebildkamera (grün), GNSS-Antennen (blau), INS (blau gestr.). Im Kofferraum befinden sich die Komponenten der Energieversorgung (weiß), High-Level-Rechner (orange) und Low-Level-Rechner (orange gestr.).

3.2 Versuchsfahrzeug VW Touareg – MuCAR-3

Das Munich Cognitive Autonomous Robot Car 3rd Generation (MuCAR-3) ist das erste Versuchsfahrzeug am Institut für Technik Autonomer Systeme (TAS) und wurde ab 2006 aufgebaut. Als Fahrzeug der dritten Generation folgt es auf VaMoRs (Versuchsfahrzeug für autonome Mobilität und Rechnersehen) und VaMP (VaMoRs-PKW), die vom Team um Prof. Ernst D. Dickmanns am Institut für Systemdynamik und Flugmechanik (ISF) der Universität der Bundeswehr München (UniBwM) in den 1980er und 1990er Jahren entwickelt wurden (aus dem ISF heraus wurde TAS gegründet).

Bei MuCAR-3 handelt es sich um einen VW Touareg 3.0 V6 TDI, der mit einem Automatikgetriebe ausgerüstet ist und für einen flexiblen Einsatz auf der Straße und im Gelände eine höhenverstellbare Luftfederung sowie eine zuschaltbare Gelände-

umsetzung besitzt. In Anlehnung an die Umbauten beim Stanley des Stanford Racing Teams [Thrun et al., 2006] wurden von der Volkswagen Konzernforschung erste DbW-Umbauten durchgeführt und MuCAR-3 zusätzlich mit einem massiven Unterbodenschutz aus Aluminium ausgestattet. Zu Beginn meiner Arbeiten am Institut war MuCAR-3 bereits geliefert.

Im folgenden Abschnitt 3.2.2 wird die für das autonome Fahren notwendige Aktorik vorgestellt, die zum Großteil durch die VW-Umbauten bereitgestellt wird. Dagegen wurden alle Rechner sowie die in Abschnitt 3.2.2 vorgestellten Sensoren, die das Fahrzeug für die Wahrnehmung der eigenen Bewegung sowie der Umwelt benötigt, in der institutseigenen Werkstatt eingerüstet.

Abbildungen 3.2 und 3.3 geben einen Überblick über das Gesamtsystem und seine Veränderungen – diese erfolgten eher evolutionär als revolutionär.

3.2.1 Energieversorgung und Kühlung

Ein Versuchsfahrzeug sollte seine Messtechnik-Zusatzausstattung bestehend aus Sensorik, Aktorik und Rechnern dauerhaft mit Strom versorgen können. Eine rein Batterie¹-gespeiste Versorgung fällt aus, da diese nur kurze Laufzeiten ermöglichen.

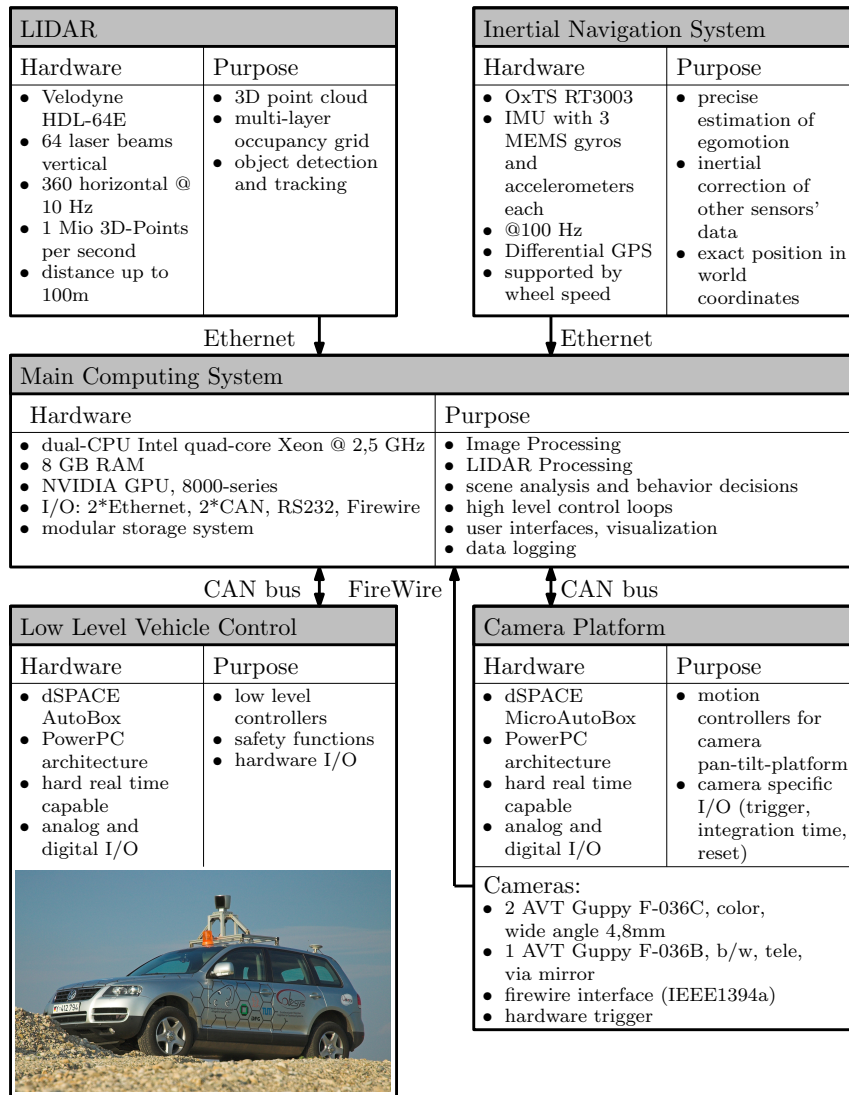
Ein Vorteil von Serien-Fahrzeugen wie dem VW Touareg ist eine bereits vorhandene Energieversorgung durch die Lichtmaschine, die solange Strom liefert wie das Fahrzeug über Sprit verfügt. Durch eine geschickte Auswahl an Sonderausstattungen mit (optional) hohem Energieverbrauch, wie einer Sitzheizung, Standheizung und Vier-Zonen-Klimaanlage, konnte beim Fahrzeugkauf erreicht werden, dass bereits ab Werk eine leistungsfähige Lichtmaschine verbaut wurde. In diesem Fall liefert sie einen Maximalstrom von 180 A bei einer Spannung von ca. 14 VDC, der allerdings nicht nur für die Messtechnik bereitsteht, sondern alle Funktionen des Basisfahrzeugs mit versorgen muss.

Im Kofferraum wurden zwei zusätzliche Batterien integriert, die über ein elektronisches Relais vom Bordnetz getrennt werden können, um in Standpausen ohne Motorlauf die Starterbatterie nicht zu belasten und trotzdem die Messtechnik zu betreiben. Zu beachten ist eine Verkabelung zwischen Lichtmaschine und Messtechnik mit hohen Querschnitten, um den Spannungsabfall und damit die Verlustleistung auch bei hohem Energieverbrauch durch die Messtechnik gering zu halten – hier haben sich 50 mm² bewährt. Für den Anschluss sämtlicher Komponenten ist auf einen zentralen Massepunkt zu achten, damit alle Analogsignale auch dann einen gemeinsamen Bezugspunkt haben, wenn keine differentiellen Ein-/Ausgänge vorhanden sind.

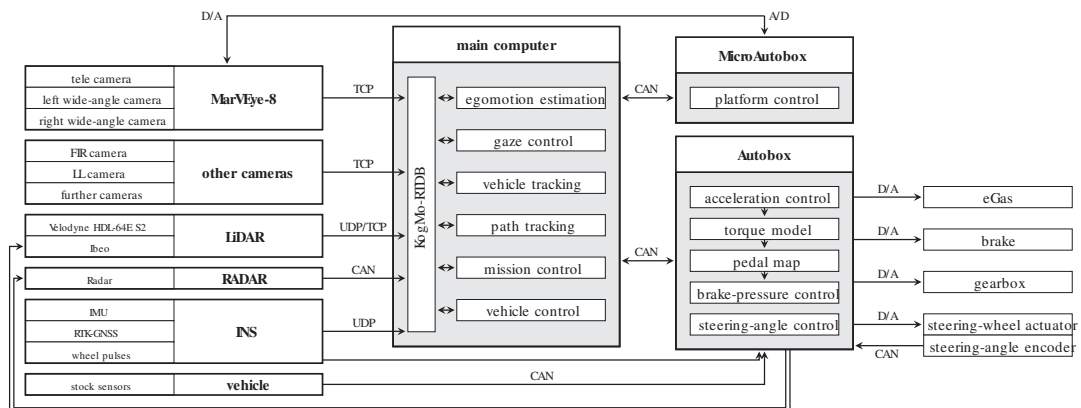
DC-DC-Konverter erzeugen weitere Gleichspannungen, bspw. 30 VDC für den Betrieb der Kameraplattform, oder 24 VDC für spezielle Sensoren.

¹In dieser Arbeit wird – wie im Automotive-Umfeld üblich – der Begriff „Batterie“ verwendet, wenn Akkumulatoren, Akkus bzw. wiederaufladbare Sekundärbatterien gemeint sind. Nicht-wiederaufladbare Primärbatterien gehören formal auch zu den Batterien, sind in dieser Arbeit aber nicht gemeint [Umweltbundesamt, 2012].

3 Versuchsfahrzeuge

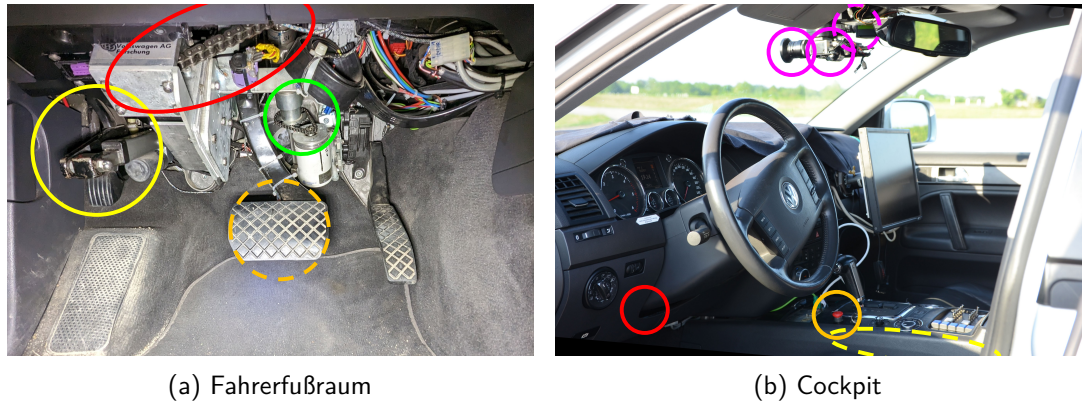


(a) Gesamtsystem zur C-ELROB 2009. Bildquelle [Himmelsbach et al., 2009b].



(b) Gesamtsystem 2017. Bildquelle: [Heinrich et al., 2017]

Abbildung 3.3: Evolution des Hardware-Gesamtsystems von MuCAR-3 bis 2017.

**Abbildung 3.4:**

Aktorik und Sicherheitseinrichtungen im Versuchsfahrzeug MuCAR-3: Lenkmotor mit Kettengetriebe (rot), Lenkwinkelsensor (grün), Feststellbremse (gelb), Gangwahlaktuator (gelb gestr.), Bremsengriffdetektion (orange gestr.), Not-Aus-Pilz zur Fahrerübernahme (orange).

Ein Sinus-Inverter erzeugt aus der Bordnetz-Gleichspannung eine Spannung von 230 VAC, sie dient primär zum Betrieb von IT-Kleinkomponenten wie z. B. Ethernet-Switches oder für Entwickler-Notebooks. Für den Laborbetrieb wurden ein 1 kW-Ladegerät integriert, welches die komplette Messtechnik versorgen kann und sämtliche Serien- und Messtechnik-Batterien aus dem sog. Landstrom lädt.

Zur Kühlung des High-Level-Rechners als größtem Energieverbraucher wurde ein Kaltluftkanal der ab Werk verbauten Vier-Zonen-Klimaanlage abgegriffen und direkt in den Rechner geführt, um durch „Umluftbetrieb“ im Kofferraum auftretenden Wärmestau zu verhindern.

3.2.2 Aktorik

E-Gas Der Touareg ist serienmäßig mit einem E-Gas-System ausgestattet, d. h. das Gaspedal wirkt nicht direkt auf die Drosselklappe sondern gibt eine winkelabhängige elektrische Spannung an das Motorsteuergerät. Diese E-Gas-Schnittstelle wird in MuCAR-3 im autonomen Betrieb vom Low-Level-Rechner angesteuert. Da das Motorsteuergerät zwei Spannungsverläufe mit unterschiedlichen Kennlinien erwartet, sorgt eine Analogschaltung für die entsprechende Signalkonditionierung.

Bremse Ebenfalls serienmäßig ist im Touareg ein aktiver Bremskraftverstärker vorhanden, der in der Regel für einen vorausschauenden Bremsdruckaufbau für das ESP (Elektronisches Stabilitätsprogramm, engl. *Electronic Stability Control*), Eingriffe des elektronischen Bremsassistenten oder für das ACC (Abstandsregeltempomat, engl. *Adaptive Cruise Control*), Verwendung findet [Heißing et al., 2011, S. 183f.]. Die Ansteuerung erfolgt über ein PWM-Signal, im autonomen Betrieb wird dieses vom Low-Level-Rechner erzeugt. Je nach Einstellung des Duty-Cycle erhöht oder verringert sich der Bremsdruck im Hauptbremszylinder.

Lenkung Eine elektro-hydraulische, geschwindigkeitsabhängige Servolenkung (Servotronic [ZF Lenksysteme GmbH, 2007]) sorgt im Touareg für reduzierte Lenkmomente am Lenkrad [Volkswagen AG, 2002a,b]. Da die Servotronic keine Schnittstelle für Lenkmomente oder Lenkwinkel anbietet, ist die Integration eines in Abb. 3.4a gezeigten separaten Lenkaktors erforderlich. Mittels Kettentrieb (Übersetzung $i^{\text{Kette}} = \frac{21}{21}$) ist ein Gleichstrommotor vom Typ Maxon RE40 148866 [maxon motor GmbH, 2023, S. 160f] mit integriertem Planetengetriebe GP42C 203120 (Untersetzung $i^{\text{Planetengetriebe}} = \frac{43}{1}$) [maxon motor GmbH, 2023, S. 447f.] an die Lenksäule angebunden². Über einen Servoverstärker vom Typ Maxon 50/10 gibt der Low-Level-Rechner per Analogsignal einen zum gewünschten Lenkmoment proportionalen Strom vor. Das maximale Moment auf die Lenksäule ergibt sich nach Gleichung (3.1).

$$\begin{aligned} M^{\text{Lenksäule}} &= M^{\text{Motor}} \cdot i^{\text{Kette}} \cdot i^{\text{Planetengetriebe}}, \\ M^{\text{Lenksäule, Dauer}} &= 0,1 \text{ N m} \cdot 1 \cdot 43 = 4,3 \text{ N m}, \\ M^{\text{Lenksäule, max}} &= 1,69 \text{ N m} \cdot 1 \cdot 43 = 72,67 \text{ N m}. \end{aligned} \quad (3.1)$$

Um den Lenkaktor zu entlasten und dauerhaft die volle Lenkunterstützung durch die Servotronic zu erhalten, wird dem Servotronic-Steuergerät im autonomen Betrieb ein künstliches Signal vom Low-Level-Rechner eingespielt, das einer niedrigen Fahrzeuggeschwindigkeit entspricht.

An dieser Stelle sei angemerkt, dass der Fahrer im manuellen Fahrbetrieb den Lenkmotor mitsamt Getriebe mitbetätigen muss, was in einem leichten, aber nicht behindernden Widerstandsmoment resultiert.

Gangschaltung Das Fahrzeug ist mit einem Sechs-Gang-Automatikgetriebe mit Wandlerkupplung ausgestattet. Bei MuCAR-3 kommt ein gegenüber dem Serienfahrzeug modifiziertes Getriebesteuergerät zum Einsatz, bei dem die Adaption der Schaltpunkte an den Fahrstil deaktiviert ist. Für die Wählhebelverstellung wurde ein Gleichstrommotor mit Trapezgewindespindel mechanisch an den Wählhebel angebunden, die Positionsmessung erfolgt über ein im Antrieb integriertes Potentiometer. Ein separates Steuergerät vom Typ Kadomo-Control [KADOMO GmbH, 2010], das normalerweise für die Umrüstung von Behindertenfahrzeugen zum Einsatz kommt, sorgt für die Verstellung des Wählhebels. Die Zielposition wird im manuellen Fahrbetrieb per Drucktaster gewählt und im autonomen Fahrbetrieb vom Low-Level-Rechner per Nachricht auf dem CAN-Bus (Controller Area Network) vorgegeben.

Feststellbremse Die Feststellbremse im Touareg wird im Serienfahrzeug über ein Fußpedal aktiviert und einen Handhebel gelöst. Analog zur Wählhebelverstellung kommt hier ebenfalls ein aus dem Behindertenumbau stammender Umrüstsatz mit separatem Steuergerät zum Einsatz. Ein Gleichstrommotor mit Trapezgewindespindel betätigt bzw. löst das Fußpedal, Endschalter sorgen dabei für eine Deaktivierung des

²Das ursprüngliche Getriebe mit einer Untersetzung von $i^{\text{Planetengetriebe, alt}} = \frac{16}{1}$ konnte das im Gelände notwendige Lenkmoment nicht erzeugen und wurde daher ersetzt.

Tabelle 3.1: Fahrdynamiksensoren

Sensor	Wertebereich		Auflösung	Freq. (Hz)
	min	max		
Radgeschwindigkeitssensoren	0	326 km/h	0,01 km/h	0
Lenkradwinkelsensor (VW)	-522,3°	522,3°	0,043 75°	100
Lenkradwinkelsensor (sep.)	0	0	0,029 296 875°	500
Luftfederung	-127 mm	126 mm	1 mm	33

Motors. Die Ansteuerung erfolgt über Digitalsignale vom Low-Level-Rechner oder im manuellen Fahrbetrieb per Drucktaster.

Sekundärfunktionen Im Touareg können nur die Blinker als Sekundärfunktion angesteuert werden.

3.2.3 Sensorik

3.2.3.1 Fahrdynamiksensoren

Geschwindigkeit Der Touareg besitzt pro Rad eine aktive Drehzahlsensoreinheit [Volkswagen AG, 2002b, S. 38]. Diese besteht aus einem mit dem Rad drehenden Polrad und einem feststehenden, am Achsschenkel montierten aktiven Sensor. Das am Radlager montierte Polrad ist hier als magnetischer Encoder ausgeführt, ein Inkrement wird hier aus zwei Nord-Südpol-Arealen gebildet. Der aktive Sensor tastet berührungslos das am Radlager montierte Polrad ab und nutzt zur Messung den anisotropen magnetoresistiven Effekt (AMR) [Volkswagen AG, 2007a, S. 11]. Eine im Sensor integrierte Elektronik erzeugt aus der Frequenz dieser Signale ein zur Raddrehzahl proportionales Rechtecksignal. Zusätzlich werden Informationen über die Drehrichtung und die aktuelle Signalgüte über den Tastgrad aufmoduliert³.

Gegenüber den früher verwendeten passiven Drehzahlsensoren, die nach dem induktiven Prinzip arbeiten, können aktuelle Sensoren Geschwindigkeiten bis hin zum Stillstand messen [Heißing et al., 2011, S. 192f.] [Wallentowitz und Reif, 2006, S. 118].

Die Drehzahlsensoreinheit des Touaregs erzeugt 100 Pulse pro Radumdrehung oder ca. 42,1 Pulse/m. Diese Signale werden vom Steuergerät des ABS (Anti-Blockier-System, engl. *Anti-lock Braking System*) bzw. ESP (Typ Continental/Teves MK25) aufbereitet und als Geschwindigkeitssignal in $\frac{\text{km}}{\text{h}}$ pro Rad sowie zusätzlich als Mittelwert bereitgestellt.

Aufgrund der Latenz in den gefilterten Geschwindigkeitsmesswerten wurde eine zusätzliche Strommessung zwischen Sensoreinheit und ABS-Steuergerät integriert,

³<http://www.kfztech.de/kfztechnik/elo/sensoren/drehzahlsensor.htm>

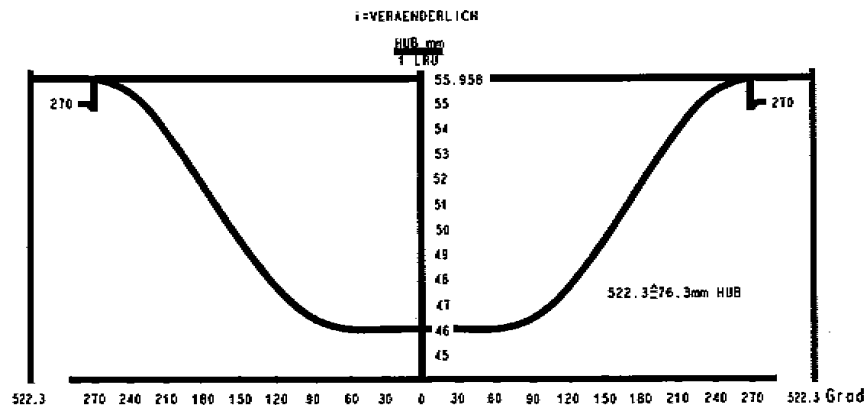


Abbildung 3.5:

Lenkgeometrie des VW Touareg mit variabler Übersetzung. Quelle: [ZF Lenksysteme GmbH, 2008].

die über einen Schmitt-Trigger (eine elektronische Komparator-Schaltung mit Hysterese) latenzfreie Rechteckpulse für die direkte Weiterverarbeitung z. B. in der Lokalisierungssensorik generiert.

Eine Alternative mit wesentlich höherer Auflösung sind externe Radimpulsgeber (z. B. mit bis zu 5000 Impulsen pro Radumdrehung [Kistler Gruppe, 2020]), die auf der Felge montiert werden und gegenüber der Karosserie einen Verdrehschutz benötigen. Damit stehen sie aus der Fahrzeugkontur heraus und können gerade im Offroad-Einsatz leicht beschädigt werden. Auf den Einsatz externer Radimpulsgeber wird daher verzichtet, zudem mit der Seriensensorik eine hinreichende Auflösung für den Anwendungszweck besteht.

Lenkung Der Lenkradwinkel wird vom serienmäßigen Lenkwinkelsensor bereitgestellt und üblicherweise vom ESP zur Ermittlung des Fahrerwunsches [Mörbe, 2015, S. 98] verwendet. Zusätzlich wurde ein Multiturn-Drehgeber vom Typ Hengstler RA58-T1212 [Hengstler GmbH, 1999] integriert und mechanisch mit einer Kette (Übersetzung Lenksäule:Drehgeber 30:10) an die Lenksäule gekoppelt. Neben einer wesentlich höheren Auflösung ($\frac{12 \text{ bit}}{360^\circ}$, d. h. $\frac{360^\circ}{2^{12}} \cdot \frac{10}{30} = \frac{360^\circ}{4096} \cdot \frac{10}{30} = 0,029\,296\,875^\circ$ pro Inkrement) bietet dieser auch eine höhere Abtastfrequenz von 500 Hz. Über die bekannte Lenkgeometrie [ZF Lenksysteme GmbH, 2008, Volkswagen AG, 2008] aus Abb. 3.5 kann damit auf den Winkel an den einzelnen Vorderrädern geschlossen werden (Übersetzung vom Lenkradwinkel auf die Verschiebung der Lenkstange sowie von der Verschiebung der Lenkstange auf den Winkel am jeweiligen Rad).

Federbeine Die Federbeine des Luftfederungs-Fahrwerks des Touaregs sind mit Sensoren ausgestattet [Volkswagen AG, 2002b,c]. Für jedes Rad steht die Einfederung als separater Messwert zur Verfügung. Da die Federbeine nicht senkrecht stehen sondern Teil der Fahrwerksaufhängung sind, ist dieser Messwert nur eine grobe Näherung für die Höhenänderung des Rades entlang der Hochachse.

Serienmäßige Beschleunigungs- und Drehratensensoren Serienmäßig vorhanden sind in allen modernen Fahrzeugen Beschleunigungs- und Drehratensensoren, die für das ESP den Ist-Zustand des Fahrzeugs in jeweils allen drei Achsen messen. Beim Touareg sind nur die Längsbeschleunigung a_x sowie die Gierrate $\dot{\Psi}$ verfügbar, beide mit starkem Rauschen sowie Offset behaftet.

Alle bisher genannten Messwerte der Fahrdynamikensensoren stehen serienmäßig auf dem CAN-Bus CAN-Antrieb zur Verfügung und können vom Low-Level-Fahrzeugrechner ausgewertet werden. Tabelle 3.1 zeigt technische Daten zu den Sensoren.

3.2.3.2 Lokalisierungssensorik

INS Zusätzlich zu den serienmäßigen bzw. -nahen Fahrdynamikensensoren wurde ein Inertiales Navigationssystem, engl. *Inertial Navigation System* (INS) vom Typ RT3003v1 des Hersteller Oxford Technical Solutions (OxTS) in MuCAR-3 integriert. Die Montage erfolgte im Bereich der Mittelarmlehne und damit in der Nähe des Fahrzeugschwerpunktes.

Dieses INS beinhaltet zum einen eine IMU (Inertiale Messeinheit, engl. *Inertial Measurement Unit*), bestehend aus je drei MEMS-basierten (Mikrosystem, engl. *Micro-Electro-Mechanical System*) Drehratensensoren und Servo-Beschleunigungssensoren. Zum anderen ist ein GNSS-Empfänger integriert, der zusätzlich über terrestrischen Funk übertragene Korrekturdaten einer RTK-Referenzstation (Echtzeitkinematik, engl. *Real Time Kinematic*) im RTCM-v3-Format (*Radio Technical Commission for Maritime Services*) verarbeiten kann. Damit kann die Positionsmessung bei freier Sicht zum Himmel auf Zentimeter-Genauigkeit gesteigert werden. Die Verwendung des integrierten sekundären GNSS-Empfängers ermöglicht eine Messung der Ausrichtung des Systems auch im Stand, welches für die Kaltstartfähigkeit des autonomen Fahrzeugs wichtig ist.

Ein sog. Strapdown-Algorithmus (siehe Abb. 3.6) berechnet mittels Integration der Beschleunigungs- und Drehratenmessungen unter Berücksichtigung von 24

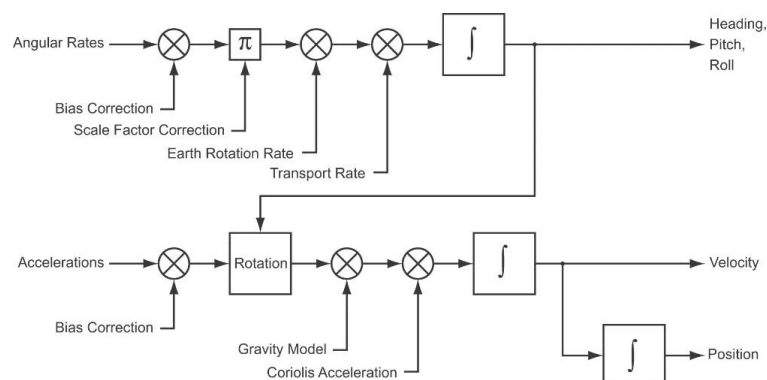


Abbildung 3.6:

Schematische Darstellung des Strapdown-Filters des INS OxTS RT3003. Bildquelle [Oxford Technical Solutions Ltd., 2020, S. 103]

Parametern⁴, die mithilfe eines Kalman-Filters geschätzt werden, die Position, Geschwindigkeit und Orientierung im Raum sowie korrigierte und kalibrierte Drehraten- und Beschleunigungsmessungen. Details finden sich in [Oxford Technical Solutions Ltd., 2020, S. 102ff.].

Als ergänzende Messung werden die in Abschnitt 3.2.3.1 erwähnten digitalen Radimpulse eines (ungelenkten) Hinterrades im INS genutzt, sie dienen zur Stillstandserkennung, zur Geschwindigkeitsmessung bei GNSS-Abschattung sowie zur Beobachtung des Bias des Gierratensensors. Bei guter Korrektur der Fehler von Geschwindigkeits-schätzung und Gierratensensormessung kann die sich durch Integration ergebene Drift der Positionsschätzung reduziert werden.

Im seit 2007 genutzten RT3003 v1 werden das amerikanische GPS (Global Positioning System) sowie seit 2009 das russische GLONASS (Globales Satellitennavigationssystem, russ. *Globalnaja nawigazionnaja sputnikowaja sistema*) genutzt, unter sehr guten Bedingungen sind bis zu 15 Satelliten parallel nutzbar. Seit 2021 kommt das Nachfolgesystem OxTS RT3000 v3 zum Einsatz, welches mit Quad-Band-Empfängern ausgestattet ist und zusätzlich das europäische Galileo sowie das chinesische Beidou nutzt. Damit sind unter optimalen Bedingungen mehr als 35 Satelliten nutzbar. Dies bringt insbesondere unter anspruchsvollen GNSS-Bedingungen einen Gewinn, da nun auch in starker Bebauung oder im Wald noch genügend Satelliten für eine Positionslösung sichtbar sind, unter denen der alte Sensor sich rein auf die Inertialsensoren verlassen musste. Ebenso wurde mithilfe eines sog. NTRIP-Modems (*Networked Transport of RTCM via Internet Protocol*) die Möglichkeit geschaffen, auch außerhalb der Funkreichweite der eigenen RTK-Referenzstation Korrekturdaten zu empfangen und damit die Positionsgenauigkeit auf Zentimeter-Level zu erhöhen.

Alle Messdaten werden mittels Ethernet/UDP (User Datagram Protocol) für den High-Level-Rechner und über den CAN-Bus für den Low-Level-Rechner bereitgestellt.

⁴Bei den Parametern handelt es sich um 3D-Positions- und -Geschwindigkeitsfehler (jeweils North, East, Down) sowie den 3D-Lagefehler (Heading, Pitch, Roll), die den Fehler der aufintegrierten Strapdown-Lösung ggü. der geschätzten besten Lösung beschreiben. Diese werden aus Stabilitätsgründen nur langsam auf die nach außen kommunizierten Zustände appliziert, um nachgelagerte Algorithmen nicht mit springenden Signalverläufen zu beeinträchtigen.

Weitere neun Parameter modellieren den sog. Bias der drei Beschleunigungssensoren sowie Bias und Skalierungsfehler der drei Drehratensensoren. Diese Parameter sind über der Zeit quasi-konstant und werden vor allem von der Temperatur und Alterungseffekten beeinflusst. Mit Bias wird eine systematische Verschiebung der Messwerte beschrieben, z. B. dass ein Sensor trotz Null-Anregung im Stillstand Werte ungleich Null liefert. Skalierungsfehler sorgen für proportionale Fehler. Beide führen bei der ein- und zweifachen Integration im Strapdown-Algorithmus zu wachsenden Fehlern, der sog. Drift, in Geschwindigkeit, Position und Orientierung. Daher werden diese Parameter mitgeschätzt und im Strapdown-Algorithmus kompensiert.

Die 3D-Position der primären GNSS-Antenne ggü. dem INS sowie zwei Winkel zur Beschreibung der Lage zwischen sekundärer und primärer GNSS-Antenne werden – ausgehend von einer initialen Kalibrierung – zur Laufzeit mitgeschätzt, um eine präzisere Positions- und Lagebestimmung im INS zu ermöglichen.

Schlussendlich wird auch das Heading bzw. der Gierwinkel zwischen INS und dem Fahrzeug geschätzt, um die beiden Koordinatensysteme achsenparallel auszurichten.

An dieser Stelle werden noch kurz die Hauptanforderungen an ein INS zusammengefasst, die die initiale Sensorauswahl für das autonome Fahren mit Fokus auf den Offroad-Bereich beeinflusst haben.

Die Orientierungs- und Dynamikmessungen müssen durchgängig auch bei GNSS-Ausfall, der unter starker Bewaldung oder im Gebirge häufig eintritt, bereitgestellt werden – dies begründet die Entscheidung, nicht nur einen GNSS-Empfänger, sondern ein INS zu verwenden.

Sowohl im urbanen Bereich (bspw. an Gebäuden) als auch im Gelände (bspw. an Felswänden) treten Reflexionen der Satellitendaten auf – daher ist eine robuste Mehrwegeunterdrückung der GNSS-Empfänger und -Antennen erforderlich, um möglichst nur die auf direktem Weg empfangenen Satellitendaten und keine reflektierten Signale zu verwenden und so Falschmessungen zu reduzieren. Mehrsystemempfänger sind hier reinen GPS-Empfängern vorzuziehen, da die Anzahl der nutzbaren Satelliten signifikant steigt.

Im Rahmen der Sensorauswahl wurden Benchmark-Fahrten im urbanen Bereich inklusive eines Parkhauses mit INS mehrerer Hersteller durchgeführt. Hier erwiesen sich die Kreiseltechnologie und allgemein die Datenblattangaben als weniger bedeutsam im Vergleich zu guten Datenfusionsalgorithmen und einer guten Anpassung auf das Versuchsfahrzeug. Da die Fahrzeuge nicht langfristig mit völliger GNSS-Abschattung betrieben werden, wie sie bspw. in Parkhäusern oder im Untertagebetrieb auftritt, kann auf die kostenintensiven Ringlaserkreisel oder faseroptische Kreisel verzichtet werden, stattdessen genügen hochwertige MEMS-Kreisel. Vor allem bei MEMS-basierten INS ist wichtig, dass ergänzend Wegimpulse aus mindestens einem Raddrehzahlsensor in die integrierte Datenfusion mit einfließen, um die Drift beobachten und kompensieren zu können.

Sowohl für die Wahrnehmungsmodule als auch zur Fahrzeugführung werden sprungfreie Orientierungs- und Dynamikmessungen benötigt. Zur inertialen Korrektur (d. h. zur Kompensation der Eigenbewegung während der Aufnahme) der hochfrequent eintreffenden Daten der Umfeldsensorik ist eine Abtastrate von mindestens 100 Hz – dazwischen kann linear interpoliert werden.

Für die autonome Fahrt auf Basis von Umfeldsensoren wird keine globale Lokalisierung in Zentimeter-Genauigkeit benötigt, und für Navigationsentscheidungen genügt eine Genauigkeit im Bereich einiger Meter. Eine hohe Lagegenauigkeit im Zentimeter-Bereich ist jedoch für Ground-Truth-Anwendungen (vgl. Abschnitt 7.3.3.5), also die quantitative Evaluierung von Wahrnehmungsalgorithmen, unerlässlich – einerseits unter Open-Sky-Bedingungen auf dem Testgelände, andererseits unter schlechten GNSS-Bedingungen im realen urbanen Umfeld oder auch Offroad. Für letzteren Anwendungsfall müssen die GNSS-Empfänger in der Lage sein, RTK-Korrekturdaten verwenden zu können, für den autonomen Betrieb sind sie jedoch nicht permanent erforderlich.

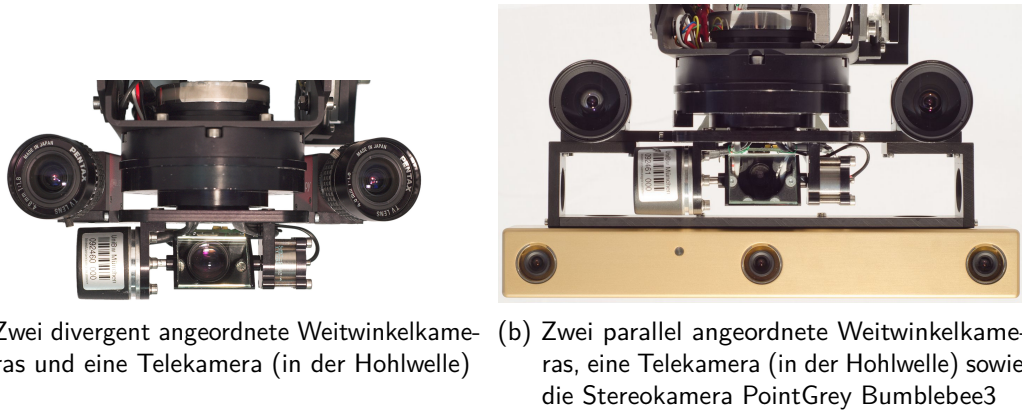


Abbildung 3.7: Kameraplattform MarVEye-8 mit unterschiedlicher Kamerabestückung.

3.2.3.3 Umfeldsensirik

Kameras Zur visuellen Umfelderkennung kommen RGB-Farbkameras (Rot-Grün-Blau) zum Einsatz, die mithilfe eines Objektivs die 3D-Welt auf eine 2D-Bildebene abbilden. Seit Beginn der Fahrzeugnutzung sind Kameras auf der drehbaren Kameraplattform MarVEye-8 (Multifokales aktives / reaktives Fahrzeug-Auge der 8. Generation, engl. *Multifocal active / reactive Vehicle Eye 8th Generation*), die mit dem Gierwinkel $\Psi_{\text{plattform}}$ um ihre Hochachse rotieren kann, hinter der Windschutzscheibe montiert und somit vor Witterungseinflüssen weitgehend geschützt.

Ein flexibler Plattformaufbau ermöglicht über Adapterplatten verschiedene Bestückungen der Plattform mit ein bis fünf Kameras [Unterholzner und Wuensche, 2009, 2010, 2012, Unterholzner, 2016], wie in Abb. 3.7 gezeigt. Im Regelfall sind links und rechts zwei Weitwinkelkameras mit gleicher Brennweite montiert, entweder in divergenter Anordnung oder parallel als Stereopaar. Eine weitere Kamera mit mildem Teleobjektiv ist in der Hohlwelle des Motors montiert und ermöglicht über einen beweglichen Spiegel eine inertielle Bildstabilisierung bei Bodenwellen [Unterholzner et al., 2010] und eine Nachführung um den Nickwinkel $\Theta_{\text{plattform}}$.

Aufgrund des rasanten technischen Fortschritts wurden die Kameras und Objektive regelmäßig durch moderne Typen ersetzt (von FireWire-Kameras mit Wide-VGA-Auflösung und abgesetztem Sensorkopf hin zu Standard-Industriekameras nach dem GigE Vision-Standard mit 5 Mpix). Derzeit verfügen die beiden Kameras über ein Horizontales Sichtfeld, engl. *Horizontal Field of View* (HFOV) von ca. 55° (links) resp. ca. 80° (rechts). Heutige hochauflösende Kamerasensoren benötigen optisch hochwertige Objektive, welche vergleichsweise schwer sind und damit das Massenträgheitsmoment der drehbaren Plattform erhöhen.

Die am Institut entwickelte Plattformregelung schätzt unter anderem dieses Massenträgheitsmoment im Betrieb mit und passt die Reglerparametrierung ohne Benutzereingriff entsprechend an [Unterholzner und Wuensche, 2009, 2010]. Damit ist die Anpassung an neue Kameras problemlos möglich, und das Grundkonzept der Plattform konnte bislang problemlos beibehalten werden. Ein hohes Massenträgheitsmoment schränkt allerdings die Dynamik der Plattform insbesondere bei

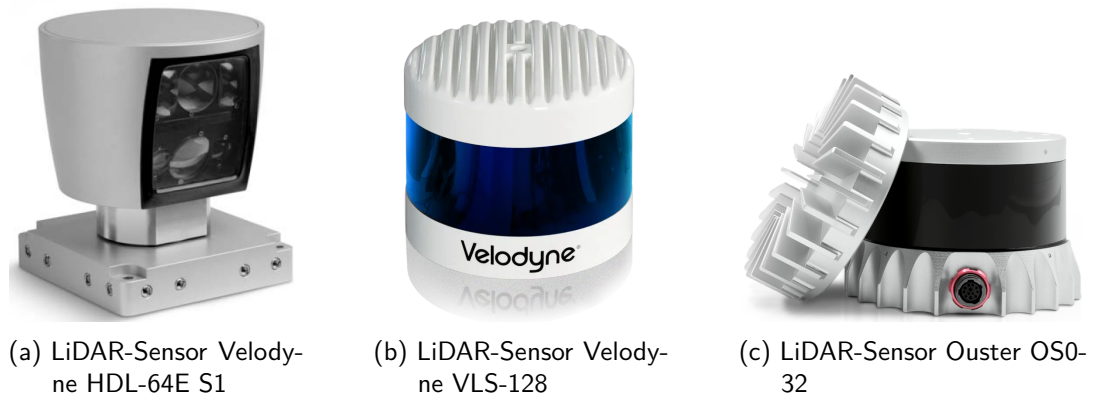


Abbildung 3.8: Verwendete LiDAR-Sensoren

Sakkaden, d. h. sprungartigen Blickrichtungsänderungen, ein. Details zur Ansteuerung der Plattform finden sich in Abschnitt 7.3.1.

Zusätzlich ergänzen fest stehende Kameras das Sensor-Setup für Anwendungen in der Dämmerung und bei Nacht: eine Wärmebildkamera vom Typ Flir A615 auf dem Dach sowie eine hochempfindliche Farbkamera vom Typ Flir Tau CNV hinter der Windschutzscheibe. Alternativ zur Flir Tau CNV kann an dieser Position auch eine kombinierte RGB-NIR-Kamera (Nahes Infrarot, engl. *Near-Infrared*) vom Typ JAI FS-3200D-10GE genutzt werden, die über ein Prisma das Licht des Objektivs anhand der Wellenlänge auf zwei Sensorchips aufspaltet und für die Vegetationsklassifikation zum Einsatz kommt.

Weitere vier Farbkameras mit Ultraweitwinkelobjektiven ($\text{HFOV} > 100^\circ$) wurden auf dem Dach des Fahrzeugs unterhalb des LiDAR-Sensors so angebracht, dass sie das komplette 360° -Sichtfeld abdecken. Diese dienen einerseits zur Einfärbung der 3D-Punktwolke, als Hilfestellung für den Menschen bei der Annotation von 3D-Punktwolken für maschinelle Lernverfahren, aber auch direkt zur Verwendung in der maschinellen Wahrnehmung.

Auf die genaue Funktionsweise der Kamerasensoren wird an dieser Stelle nicht weiter eingegangen.

LiDAR Neben den Kamerasensoren, die nur eine 2D-Abbildung der Welt erfassen können, verfügt MuCAR-3 über 3D-LiDAR-Sensoren, die eine 3D-Messung des Fahrzeugumfelds liefern.

Es wurde als erstes Versuchsfahrzeug in Europa mit dem hochauflösenden Velodyne HDL-64E S1 ausgestattet [Velodyne Acoustics, Inc., 2011]. Der Velodyne HDL-64E S1 ist dergestalt auf dem Dach des Fahrzeugs montiert, dass der unterste Laserstrahl gerade nicht auf die Karosserie trifft.

Der Sensorkopf rotiert standardmäßig mit einer Frequenz von 10 Hz um seine Hochachse. Seine 64 Laserdioden senden ca. 1 Million Laserpulse pro Sekunde mit einer Wellenlänge von ca. 850 nm aus, die von Objekten in der Umwelt reflektiert werden.

Optische Empfänger detektieren die reflektierten Pulse und ermöglichen die Bestimmung der Laufzeit sowie die Stärke der Reflexion. Aus der Laufzeit kann mithilfe der bekannten Lichtgeschwindigkeit die Entfernung für jeden reflektierten Puls berechnet werden.

Mithilfe der bekannten Diodenanordnung innerhalb des Sensorkopfes, der sog. intrinsischen Kalibrierung, und unter Berücksichtigung des aktuellen Rotationswinkels wird aus diesen Entfernungsmessungen eine 3D-Punktwolke für jede Umdrehung berechnet. Eine alternative Darstellung ist das sog. Range Image, eine Darstellung der gemessenen Entfernungen einer Rotation in einem Bild mit 64 Zeilen.

Der Sensor verfügt über ein Vertikales Sichtfeld, engl. *Vertical Field of View* (VFOV) von $\Theta_{\min} = -2^\circ$ nach oben und $\Theta_{\max} = 24,33^\circ$ nach unten, wobei die oberen 32 Dioden mit einer Spreizung von $\Delta\Theta_{\text{top}} = \frac{1}{3}^\circ$ dichter angeordnet sind als die unteren 32 mit $\Delta\Theta_{\text{bottom}} = \frac{1}{2}^\circ$. Die horizontale Auflösung ist abhängig von der eingestellten Rotationsgeschwindigkeit, in unserem Fall ergibt sich ein Wert von $\Delta\Psi_{10\text{Hz}} = 0,23^\circ$.

Bei den verwendeten Sensoren ist die optische Strahlaufweitung wesentlich geringer als die Spreizung der Dioden. Daher wird die Umgebung nur über einzelne Punkte abgetastet, sodass es passieren kann, dass kleinere Hindernisse von keinem Laserpuls getroffen werden und daher in den Sensordaten unsichtbar bleiben. Durch die geringere vertikale Spreizung $\Delta\Theta_{\text{top}}$ kann dieser Effekt für zu detektierende Objekte in der Entfernung etwas abgeschwächt werden. Die nominelle Reichweite des Velodyne HDL-64E S1 liegt bei über 100 m. Im Außenbereich ist die Punktwolke allerdings sehr dünn besetzt, sodass die Daten eher nur in der ersten Hälfte nutzbar sind.

Im Laufe der Fahrzeugnutzung wurde 2010 der Nachfolgesensor Velodyne HDL-64E S2 integriert, der eine um 30 % höhere Punktdichte liefert und intrinsisch besser kalibriert ist [Velodyne LiDAR, Inc., 2011].

Anfang 2020 folgte der Wechsel zur aktuellen Sensorgeneration: Der Velodyne VLS-128 verfügt mit 128 Laserdioden über eine doppelt so hohe vertikale Auflösung und wesentlich bessere Reichweite [Velodyne LiDAR, Inc., 2021]. Der Sensor verfügt über ein VFOV von $\Theta_{\min} = -15^\circ$ nach oben und $\Theta_{\max} = 25^\circ$ nach unten. Im Normalbetrieb liefert der Sensor ca. 2,4 Millionen 3D-Punkte pro Sekunde, im sog. Dual-Return-Modus entsprechend das Doppelte.

Zur Reduktion der seitlichen toten Winkel neben dem Fahrzeug wurden 2022 zwei Nahbereichssensoren vom Typ Ouster OS0-32 [Ouster, Inc., 2020] mit einem VFOV von $\Theta_{\min/\max} = \pm 45^\circ$ ergänzt, die um 45° nach unten gekippt montiert sind.

Temporär wurde der seriennahe LiDAR-Sensor Ibeo Lux 8L [Ibeo Automotive Systems GmbH, 2012] im Touareg verwendet, er befindet sich mittig in der vorderen Stoßstange. Hier handelt es sich um einen 8-Ebenen-Scanner, der abwechselnd die oberen 4 Ebenen für ein integriertes Objekttracking und die unteren 4 Ebenen für eine mögliche Hinderniserkennung abtastet. Das VFOV beträgt $6,4^\circ$, das HFOV umfasst den Bereich von -60 bis 50° .

Die verwendeten Sensoren sind in Abb. 3.8 dargestellt.

Radar Im Vergleich zu LiDAR-Sensoren und Kameras arbeiten Radar-Sensoren robust unter verschiedenen Witterungsbedingungen und sind daher eine gute Ergänzung für ein heterogenes Sensor-Setup. Zusätzlich können sie mithilfe des Dopplereffekts direkt eine Relativgeschwindigkeit messen.

In Serienfahrzeugen war der erste Anwendungsfall für Radar-Sensoren das ACC, heutzutage kommt eine Vielzahl von Radar-Sensoren für diverse Funktionen zum Einsatz. Der Touareg verfügt serienmäßig über keine Radar-Sensorik, daher wurde 2011 ein im 24 GHz arbeitender Fernbereichsradar vom Typ UMRR09-29 des Kleinserienherstellers smart microwave sensors GmbH integriert. Dieser Sensor enthält eine Sende- und zwei Empfangsantennen, die ein HFOV von 36° sowie ein VFOV von 8° abdecken. Die integrierte Datenauswertung stellt über den CAN-Bus neben den über der Zeit verfolgten Objekten auch ungefilterte sog. Detektionen, die stärksten Echos, bereit. Der Vorteil ggü. Sensoren von großen Herstellern wie Continental oder Aptiv ist die offene Schnittstelle, über die man auch Detektionen erhält.

Um allein mithilfe von Radar-Sensoren Objekte rund um das Fahrzeug detektieren und verfolgen zu können, wurden 2021 im Rahmen einer Studie moderne Sensoren desselben Herstellers ergänzt. Es kommen fünf Sensoren UMRR-96 Type-153 [Smart Microwave Sensors GmbH, 2020] mit großem Öffnungswinkel an den Fahrzeugecken sowie im Kühlergrill zum Einsatz [Smart Microwave Sensors GmbH, 2020], die Reichweite beträgt im Long-Range-Modus bis 120 m bei einem HFOV von $\pm 50^\circ$ bzw. im Medium-Range-Modus bis 55 m bei einem HFOV von $\pm 65^\circ$. Das Setup wird ergänzt durch einen Fernbereichs-Radar UMRR-11 Type-132 [Smart Microwave Sensors GmbH, 2021] mit einer Reichweite bis 175 m bei einem HFOV von $\pm 16^\circ$ bzw. bis 64 m bei einem HFOV von $\pm 50^\circ$. Im Gegensatz zum alten Sensor verfügen die aktuellen Sensoren auch über eine Trennfähigkeit im vertikalen Bereich, wenn auch mit geringerer Winkelgenauigkeit. Die Anbindung erfolgt über Automotive Ethernet 100BASE-T1 (auch als BroadR-Reach bezeichnet) und einen Ethernet-Medienkonverter.

3.2.4 Rechnerarchitektur, Kommunikation

Die Rechnerarchitektur von MuCAR-3 orientiert sich an den in Abschnitt 3.1 aufgezeigten Gemeinsamkeiten und ist in Low-Level- und High-Level-Rechner unterteilt.

Low-Level-Rechner Als Low-Level-Rechner kommt eine modulare dSPACE AutoBox mit einer PowerPC-Prozessorkarte DS1005 und einer I/O-Karte DS2202 zum Einsatz. Letztere bietet mehrere CAN-Busse für die Kommunikation mit dem Basisfahrzeug (über ein Gateway) und dem High-Level-PC sowie eine Vielzahl an potentialfreien analogen und digitalen Schnittstellen für die Ansteuerung von Aktoren oder dem Einlesen von Mess- und Statuswerten.

Bei der dSPACE AutoBox handelt es sich um ein Echtzeitsystem, das das enthaltene Programm mit einer Frequenz von 1000 Hz ausführt. Die Software wurde mit Matlab Simulink entwickelt und umfasst neben der Signalkonditionierung für die I/O-Module

Tabelle 3.2: Evolution der High-Level-Rechner in MuCAR-3

Jahr	CPU-Typ	#CPUs x #Kerne	Frequenz (GHz)	RAM (GB)	GPU
2006	AMD Opteron 270HE	2x 2	2,0	4	
2008	Intel Xeon L5420	2x 4	2,5	8	
2010	Intel Xeon L5640	2x 6 + HT	2,26	12	
2015	Intel Xeon E5-2667v3	2x 8 + HT	3,2	64	Titan RTX
2022	AMD Threadripper PRO 3975WX	1x 32 + HT	3,5	256	RTX A6000

vor allem die unterlagerten Regler für Lenkwinkel, Lenkwinkelgeschwindigkeit, Geschwindigkeit und Beschleunigung sowie entsprechende Vorsteuerungen. Auch die Überwachung der Sicherheitseinrichtungen ist Teil dieser Software.

Als Besonderheit kommt im Touareg eine separate dSPACE MicroAutoBox DS1401/1501 für die Ansteuerung der Kameraplattform MarVEye-8 zum Einsatz. Dies ist allerdings historisch bedingt, da [Unterholzner, 2016] parallel zu dieser Arbeit entstanden ist und die anfängliche Entwicklung durch getrennte Hardware entkoppelt und parallelisiert werden konnte.

Insgesamt ist das Low-Level-System in Hard- und Software als langzeitstabil zu betrachten.

High-Level-Rechner Grundvoraussetzung bei der Systemauslegung der High-Level-Rechner ist, dass die Energieversorgung dauerhaft durch die Lichtmaschine erfolgen kann und damit die schnellsten CPU- und GPU-Typen nicht verwendet werden können. Aus Kosten- und Verfügbarkeitsgründen hat es sich bewährt, handelsübliche Server- bzw. Workstation-Mainboards für die x86-64-Architektur in einem kompakten 19“-Rack-Gehäuse zu integrieren. Entscheidender Unterschied zu Rechnern im Labor ist ein leistungsfähiges ATX-Netzteil, welches aus dem 12 VDC-Bordnetz betrieben wird. Aufgrund der rasanten Entwicklung bei der Leistungsfähigkeit von CPUs (steigende Anzahl der Kerne, Einführung von Hyper-Threading (HT)) und GPUs sowie der steigenden Anforderungen an die Rechenleistung durch moderne Sensoren und Algorithmen ist der High-Level-Rechner weniger langzeitstabil als der Low-Level-Rechner und wird im Abstand weniger Jahre aktualisiert. Die größte Änderung im Systemkonzept wurde durch Machine-Learning-Verfahren notwendig, die eine leistungsstarke GPU benötigen. Zuvor wurde diese rein für Visualisierungszwecke verwendet und konnte energiesparend ausgewählt werden. Für Betriebssystem, Anwendungen und zur Messdatenaufzeichnung kamen bis zur 2008er-Generation Serverfestplatten zum Einsatz, danach erfolgte die Umstellung auf die wesentlich leistungsfähigeren und robusteren SSDs. Die Evolution der Rechner ist in Tabelle 3.2 dargestellt.

Die Schnittstellen haben sich mit der Weiterentwicklung der Sensoren geändert, FireWire wird heute nicht mehr benötigt. Stattdessen kommen mittlerweile 10 Gbit/s-

Ethernetschnittstellen zur Verwendung. Des Weiteren stehen CAN-Schnittstellen für die Kommunikation zu den beiden Low-Level-Rechnern bereit.

Der High-Level-Rechner ist mit einem Benutzerarbeitsplatz im Fahrzeug montiert, der Monitor ist am Armaturenbrett befestigt und vom Beifahrer- und Fahrersitz einsehbar. Damit wird den Wissenschaftlern ermöglicht, auch direkt im autonomen Fahrzeug an ihrer Software zu arbeiten.

Softwareseitig kommt Linux als Betriebssystem zum Einsatz, wobei die Distributionen über die Zeit gewechselt wurden (Debian, Arch Linux, Ubuntu). Zur Interprozesskommunikation, engl. *Inter-Process Communication* (IPC) kam bis 2020 die in Abschnitt 4.1.2 vorgestellte KogMo-RTDB (Echtzeitdatenbank für kognitive Automobile, engl. *KogniMobil Real Time Data Base*) zum Einsatz, danach erfolgte ein schrittweiser Umstieg auf ROS (Robot Operating System).

Kommunikation Zur Kommunikation mit weiteren Fahrzeugen ist MuCAR-3 mit einem Long-Range-WLAN-System (Drahtloses lokales Netzwerk, engl. *Wireless Local Area Network*) ausgestattet, um Ground-Truth-Messdaten auszutauschen. Auf dem Testgelände der UniBw M kann dieses WLAN ebenfalls zur Kommunikation mit dem Uni-Netz genutzt werden und ermöglicht den Fernzugriff vom Labor auf das Fahrzeug. Für Datenverbindungen außerhalb des Testgeländes steht eine kommerzielle Lösung über das Mobilfunknetz zur Verfügung.

3.2.5 Sicherheitseinrichtungen

Das Versuchsfahrzeug MuCAR-3 ist zwar zur Entwicklung und Erprobung autonomer Fahrfunktionen aufgebaut worden, allerdings wird es nur bemannt betrieben. Dies begründet sich einerseits in der Zulassung für den Straßenverkehr, andererseits darin, dass TAS als Uni-Institut nur über begrenzte Entwicklerkapazitäten verfügt: Ziel bei der Entwicklung ist nicht die Sicherstellung von funktionaler Sicherheit und Robustheit, die für einen unbemannten Betrieb und ein kommerzielles Produkt unerlässlich wären, sondern die Erprobung der entwickelten Algorithmen als Proof-of-Concept im echten Umfeld und – für einzelne Szenarien – auch mit einer für Wettbewerbe und Demonstrationen erforderlichen Robustheit.

Der menschliche Sicherheitsfahrer ist dabei das Kernelement des mit dem Zentrum Kraftfahrwesen der Bundeswehr (ZKfWBw) abgestimmten Sicherheitskonzepts: Der Sicherheitsfahrer überwacht das Fahrzeug an sich und dessen Umfeld. Im Fehlerfall oder in Gefahrensituationen kann er jederzeit das Fahrzeug in den manuellen Modus überführen und die volle Kontrolle übernehmen.

Dazu stehen ihm mehrere Möglichkeiten offen, die in Abb. 3.4 dargestellt sind: Ein Notaus-Knopf unterbricht elektrisch alle Ansteuerungen vom Low-Level-System in die Fahrzeugaktorik und stellt diesbezüglich den Serienzustand des Touareg wieder her. Eine Betätigung des Bremspedals wird über eine drucksensitive, widerstandsveränderliche Folie im Pedalgummi erfasst, sodass softwareseitig im Low-Level-System

alle Aktorikansteuerungen abgeschaltet werden. Ebenso kann anhand des Spannungssignals des Gaspedals ein Eingriff erkannt werden.

Eine Übersteuerung der Lenkaktorik durch den Fahrer ist jederzeit möglich und war Teil der Abnahme im Rahmen der Zulassung. Dagegen wurde eine Erkennung der Lenkradberührung nur im Rahmen einer Studienarbeit von Böhm [2007] untersucht, aber mangels Robustheit nicht final integriert. Hier kam die Wicklung der Lenkradheizung als kapazitiver Sensor zum Einsatz, bei dem die Hand des Fahrers das Dielektrikum und damit die Kapazität veränderte, was die Frequenz in einem Schwingkreis beeinflusste. Diese Arbeit orientierte sich an einer Patentoffenlegungsschrift [Gensler und Breitenberger, 2002].

Die zweite wichtige Person ist der Operator, er überwacht primär die Funktion der Software. Dazu dient ihm eine Visualisierung der unterschiedlichen Softwaremodule auf der am Armaturenbrett montierten Konsole. Diese Visualisierung ist extrem wichtig, da Fehlfunktionen in den Wahrnehmungsalgorithmen direkt sichtbar sind, sich aber erst während der Fahrt durch eine fehlerhafte Bahnplanung auswirken. Damit erhält der Sicherheitsfahrer eine „Zeitreserve“ für den prädiktiven Eingriff vor dem Auftreten des Fahrfehlers.

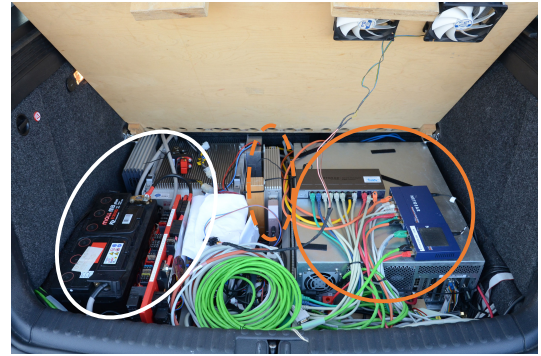
Für einige Anwendungsfälle, z. B. Wettbewerbsteilnahmen oder Demonstrationen, wurde zusätzlich ein externer Not-Stopp-Taster integriert. Eine Betätigung sorgt für eine kontrollierte Notbremsung durch das Low-Level-System bis in den Stand und kann nur über einen separaten Freigabeschalter wieder aufgehoben werden.

Ein Watchdog im Low-Level-System prüft permanent die Fahrdynamiksensoren und Aktoren auf korrekte Funktion sowie das Alter der Steuergrößen. Fällt ein Signal oder Teilsystem aus, so kann entweder der sog. Abwurf geschehen, dies ist eine durch optische oder akustische Signale angekündigte Schnellübergabe an den Sicherheitsfahrer, oder – im Wettbewerbsszenario – eine kontrollierte Notbremsung.

Im Rahmen der Zulassung wurde eine gelbe Rundumkennleuchte (RKL) gefordert, um nach außen zu signalisieren, ob das Fahrzeug im manuellen oder autonomen Modus betrieben wird. Da aufgrund der veränderlichen Dachaufbauten nicht gewährleistet werden kann, dass eine einzelne RKL aus allen Richtungen sichtbar ist, wurde je eine RKL am vorderen und hinteren Ende des Daches montiert.



(a) Foto MuCAR-4



(b) Komponenten-Überblick

Abbildung 3.9:

Versuchsfahrzeug MuCAR-4 mit montierter Sensorik: LiDAR Velodyne VLS-128 (gelb), RGB-Kameras (magenta), SW-Stereo-Kameras (grün), Nerial-Stereokamera (magenta gestr.), GNSS-Antennen (blau), INS (blau gestr.). Unter dem Kofferraumboden befinden sich die Energieversorgung (weiß), High-Level-Rechner (orange), Low-Level-Rechner (orange gestr.).

3.3 Versuchsfahrzeug VW Tiguan – MuCAR-4

Das in Abb. 3.9a gezeigte Munich Cognitive Autonomous Robot Car 4th Generation (MuCAR-4) ist ein VW Tiguan 2.0 TSI Track&Field, der ab Ende 2009 als zweites Versuchsfahrzeug des Instituts aufgebaut wurde. Die ersten zwei Jahre kam es primär als Sensorträger zum Einsatz, die Ausstattung mit Aktorik bzw. Schnittstellen erfolgte schrittweise bis Anfang 2012.

In den folgenden Abschnitten liegt der Fokus auf Unterschieden zu MuCAR-3.

3.3.1 Energieversorgung und Kühlung

Die Energieversorgung ist sehr ähnlich zum Touareg aufgebaut, aber in mehreren Aspekten etwas schwächer dimensioniert. Die Lichtmaschine liefert einen Maximalstrom von 140 A bei einer Spannung von ca. 14 VDC, und es kommt nur eine Pufferbatterie für die Messtechnik zum Einsatz. Zur Kühlung wurde im Mitteltunnel ein Kaltluftkanal der serienmäßigen Zwei-Zonen-Klimaanlage abgegriffen und in den Kofferraum-Unterboden geführt, wo nach Abb. 3.9b alle Komponenten montiert sind.

3.3.2 Aktorik

E-Gas Der Tiguan ist genau wie der Touareg serienmäßig mit einem E-Gas-System ausgestattet. Diese E-Gas-Schnittstelle wird in MuCAR-4 im autonomen Betrieb vom Low-Level-Rechner angesteuert.

**Abbildung 3.10:**

Aktorik und Sicherheitseinrichtungen im Versuchsfahrzeug MuCAR-4: Lenkmotor als Teil der Servolenkung (rot), redundante Bremsaktorik (grün), Bremseingriffdetektion (orange gestr.), Not-Aus-Pilz zur Fahrerübernahme (orange).

Bremse Zur Ansteuerung der Bremse kommt ein Nachrüstsystem des niederländischen Herstellers Bever zum Einsatz [BEVER TMC B.V., 2008]. Dieses besteht aus zwei redundanten Gleichstrommotoren mit integriertem Getriebe, die über Bowdenzüge auf das Bremspedal wirken (siehe Abb. 3.10a). Die zugehörige Elektronik ist ebenfalls redundant ausgelegt und erhält die wegproportionalen Sollwerte vom Low-Level-Rechner über eine Anlogschnittstelle. Dieses Nachrüstsystem kommt normalerweise mit einem an den Fahrer angepassten Bedienelement in Behindertenfahrzeugen zum Einsatz.

Lenkung Der Tiguan ist serienmäßig mit einer elektro-mechanischen Servolenkung vom Typ APA-BS ausgestattet [Volkswagen AG, 2007b,c]. Neben der geschwindigkeitsabhängigen Lenkunterstützung – vergleichbar der elektro-hydraulischen Servolenkung im Touareg – bietet sie diverse Komfortfunktionen. Hier sind eine verbesserte Lenkradrückstellung genauso zu nennen wie die Funktion Fahrer-Lenkempfehlung, engl. *Driver Steering Recommendation* (DSR), die z. B. eine Seitenwindkorrektur bietet, sowie die Lenkradwinkelschnittstelle für den Parklenkassistent, engl. *Park Assist* (PA), der das Fahrzeug sensorgestützt in Parklücken längs der Fahrbahn manövrieren kann [Volkswagen AG, 2007d] – hier ist allerdings nur die Querführung automatisiert. Diese Servolenkung wird mit einer speziellen Firmware aus der VW-Vorentwicklung betrieben, die diese Schnittstellen für die externe Ansteuerung und beliebige Geschwindigkeitsbereiche öffnet. Der Low-Level-Rechner kann einerseits die Lenkradwinkelschnittstelle des PA mit einer Winkelauflösung von $\Delta\lambda_{\text{Lenkrad}} = 0,04375^\circ$ ansteuern, und andererseits die Lenkmomentschnittstelle des DSR für eigene Lenkraten- oder Lenkwinkelregler verwenden. Zur Modifikation der CAN-Botschaften vom PA-Steuergerät zum Servolenkungssteuergerät wurde ein Gateway in den Bus eingeschleift, welches bei aktiver Ansteuerung einzelne Bytes der Nachricht austauscht und die Prüfsummen neu berechnet.

Gangschaltung MuCAR-4 ist – genau wie MuCAR-3 – mit einem Sechs-Gang-Automatikgetriebe mit Wandlerkupplung ausgestattet. Für die Wählhebelverstellung wurde ebenfalls ein Gleichstrommotor mit Trapezgewindespindel und im Antrieb integrierten Potentiometer integriert. Dieser ist mechanisch über einen Bowdenzug parallel zum Wählhebel an das Getriebe gekoppelt. Die Ansteuerung erfolgt identisch zu MuCAR-3 mithilfe eines separaten Steuergeräts vom Typ Kadomo-Control [KADOMO GmbH, 2010].

Sekundärfunktionen Das Kadomo Control ermöglicht desweiteren die Ansteuerung einiger Sekundärfunktionen und wird mittels CAN-Botschaften vom Low-Level-Rechner angesteuert. Neben den Blinkern handelt es sich um die Scheibenwischer, die Fensterheber, das Abblend- und Fernlicht, die Warnblinkanlage sowie die Hupe.

Für die Ansteuerung von Abblendlicht und Warnblinkanlage kommen Relais im Kadomo Control zum Einsatz. Für die anderen Funktionen modifiziert es als Gateway im Komfort-CAN-Bus die Botschaften aus dem Bereich der Lenkstockscharter und des Lenkrades.

3.3.3 Sensorik

Die Sensorausstattung von MuCAR-4 ist an die in Abschnitt 3.2.3 vorgestellte Sensorausstattung von MuCAR-3 angelehnt. Bei den serienmäßige Beschleunigungs- und Drehratensensoren gibt es keine Unterschiede.

3.3.3.1 Fahrdynamiksensorik

Geschwindigkeit Beim Tiguan kommen an allen vier Rädern aktive Drehzahlsensoreinheiten zum Einsatz, die den Hall-Effekt nutzen (vgl. Abschnitt 3.2.3). Dabei werden 43 Pulse pro Radumdrehung oder ca. 20,6 Pulse/m erzeugt, vom Steuergerät des ABS aufbereitet und als Geschwindigkeitssignal in $\frac{\text{km}}{\text{h}}$ pro Rad sowie zusätzlich als Mittelwert bereitgestellt.

Lenkung Die Messung bzw. Berechnung des Lenkwinkels erfolgt innerhalb der APA-BS-Lenkaktorik und wird auf dem CAN-Bus als Lenkradwinkel mit einer Winkelauflösung von $\Delta\lambda_{\text{Lenkrad}} = 0,04375^\circ$ bereitgestellt. Über die bekannte Lenkgeometrie kann damit auf den Winkel an den einzelnen Vorderrädern geschlossen werden. Dazu kommt eine feste Übersetzung vom Lenkradwinkel auf die Verschiebung der Lenkstange von $\frac{53,4\text{mm}}{360^\circ}$ sowie eine nichtlineare Kennlinie von der Verschiebung der Lenkstange auf den Winkel am jeweiligen Rad zum Einsatz [Volkswagen AG, 2012].

Als weitere Messung wird das durch einen Lenkmomentsensor in der Lenksäule detektierte Lenkhandmoment auf dem CAN-Bus bereitgestellt [Volkswagen AG, 2007c, S. 25f.].

Federbeine Der Tiguan besitzt im Gegensatz zu MuCAR-3 keine Luftfederung, sondern nur eine Adaptive Fahrwerksregelung (DCC) an der Hinterachse. Auf dem CAN-Bus werden zwar Einfederungswerte kommuniziert, allerdings nur für die beiden vorderen Federbeine separat und hinten als Mittelwert.

3.3.3.2 Lokalisierungssensorik

INS In MuCAR-4 wurde zu Beginn ebenfalls ein INS vom Typ OxTS RT3003 v1 integriert, welches 2021 durch das Nachfolgesystem OxTS RT3000 v3 ersetzt wurde. Die Montage erfolgte unter der Mittelarmlehne auf dem Kardantunnel und damit in der Nähe des Fahrzeugschwerpunktes.

3.3.3.3 Umfeldsensorik

Kameras Analog zu MuCAR-3 ist auch MuCAR-4 mit einer aktiven Kameraplattform MarVEye-8 hinter der Windschutzscheibe ausgestattet, auf der sich ein Stereokamerasystem befindet. In einer frühen Ausbaustufe kam das kommerziell verfügbare Stereokamerasystem PointGrey Bumblebee3 mit Firewire-Schnittstelle zum Einsatz, welches in Abb. 3.7b gezeigt ist. Mittlerweile kommen RGB-Industriekameras nach dem GigE Vision-Standard mit einer Auflösung von 5 Mpix zum Einsatz, die mit leichten Weitwinkelobjektiven (HFOV ca. 50°) ausgestattet sind.

Des Weiteren verfügt MuCAR-4 über mehrere Stereokamerasysteme mit unterschiedlichen Basisabständen auf dem vorderen Fahrzeugdach, wie in Abb. 3.9a gezeigt. Hier kommt zuerst ein Nerian SP1 Stereovision-Sensor [Nerian Vision Technologies, 2017] in Verbindung mit zwei Industriekameras mit USB-Schnittstelle (Universal Serial Bus) zum Einsatz, welches über eine auf einem FGPA (Field Programmable Gate Array) integrierte Tiefenbildberechnung verfügt. Zusätzlich wurde ein am Institut entworfenes Trifokales Stereosystem integriert. Dieses besteht aus einem horizontalen Stereopaar mit großem Basisabstand und einer dritten Kamera, die mit der linken Kamera ein weiteres vertikales Stereopaar bildet [Kallwies und Wuensche, 2018]. Hier kommen Standard-Industriekameras nach dem GigE Vision-Standard mit einer Auflösung von 5 Mpix zum Einsatz, allerdings mit einem Grauwert-Sensor.

Zeitweise war rückwärtig über der Heckklappe ein weiteres, leicht nach unten geneigtes Stereokamerapaar für das Rangieren integriert, bestehend aus RGB-Industriekameras mit 2 Mpix und starken Weitwinkelobjektiven.

LiDAR MuCAR-4 war seit Anfang 2010 mit einem LiDAR-Sensor vom Typ Velodyne HDL-64E S2 ausgestattet. Dieser wurde 2022 durch den höher auflösenden Velodyne VLS-128 ersetzt.

Radar MuCAR-4 verfügt derzeit über keine Radar-Sensoren.

Ultraschallsensoren Das Basisfahrzeug verfügt über je vier Ultraschallsensoren in der vorderen und hinteren Stoßstange, die für die Parkdistanzkontrolle, engl. *Park Distance Control* (PDC) zum Einsatz kommen⁵. Zwei zusätzliche Sensoren in den vorderen Kotflügeln dienen dem PA zur Detektion möglicher seitlicher Parklücken. Die acht PDC-Sensoren decken einen Messbereich bis 2,54 m ab, ihre Messwerte stehen auf dem CAN-Bus zur Verfügung und werden vom Low-Level-Rechner eingelesen.

3.3.4 Rechnerarchitektur, Kommunikation

Die Rechnerarchitektur von MuCAR-4 orientiert sich ebenfalls an den in Abschnitt 3.1 aufgezeigten Gemeinsamkeiten und ist in Low-Level- und High-Level-Rechner unterteilt.

Low-Level-Rechner Hier wurde eine gemeinsame dSPACE MicroAutoBox integriert, die die Funktionalität der im Touareg noch getrennten Systeme für die Fahrzeugansteuerung sowie die Ansteuerung der Kameraplattform MarVEye-8 vereint. Anfangs kam eine zum Touareg baugleiche DS1401/1501 zum Einsatz, die zur Kommunikation mit weiteren Rechnern nur über CAN-Bus-Schnittstellen verfügte. Diese wurde später um ein proprietäres Ethernet-Interface erweitert. Ein Nachteil dieser Hardware-Version ist, dass die Analog-Ein- und Ausgänge nur einen Spannungsbereich von 0 bis 5 V ggü. der Versorgungs-Masse unterstützt (keine differentiellen Ein-/Ausgänge), daher ist für viele Signale eine zusätzliche Signalkonditionierung und Entkopplung der Massen notwendig.

2021 erfolgte eine Migration auf die dSPACE MicroAutoBox II 1401/1513, die über zusätzliche CAN-Bus-Schnittstellen verfügt sowie vor allem eine native Ethernet-Unterstützung bietet. Ebenso verfügen die Analog-Ein- und Ausgänge hier über einen Spannungsbereich von 0 bis 5 V, allerdings weiterhin ggü. einer gemeinsamen Masse und nicht differentiell. Diese Migration erlaubt einige Vereinfachungen in der externen Signalkonditionierung.

High-Level-Rechner Im Tiguan kommt ebenfalls eine handelsübliche Workstation im 19“-Rack-Gehäuse als High-Level-Fahrzeugrechner zum Einsatz, Tabelle 3.3 gibt einen Überblick über die Veränderungen in der Hardware über die Jahre.

Bis 2018 kam – analog zu MuCAR-3 – der CAN-Bus zur Kommunikation mit dem Low-Level-Rechner zum Einsatz, dieses wurde durch eine Ethernet-Schnittstelle mit dem UDP-Protokoll abgelöst. Neben einer Reduzierung der Latenz führte dies vor allem zu einfacheren Schnittstellenmodulen und zu einer höheren Robustheit.

Für die Anbindung der zahlreichen Sensoren kommen primär Ethernet-Verbindungen, welche über 10 Gbit/s-Switche aggregiert und zum Rechner geführt werden, zum Einsatz.

⁵MuCAR-3 verfügt zwar auch über eine PDC, allerdings als gekapseltes System ohne Kommunikation nach außen.

Tabelle 3.3: Evolution der High-Level-Rechner in MuCAR-4

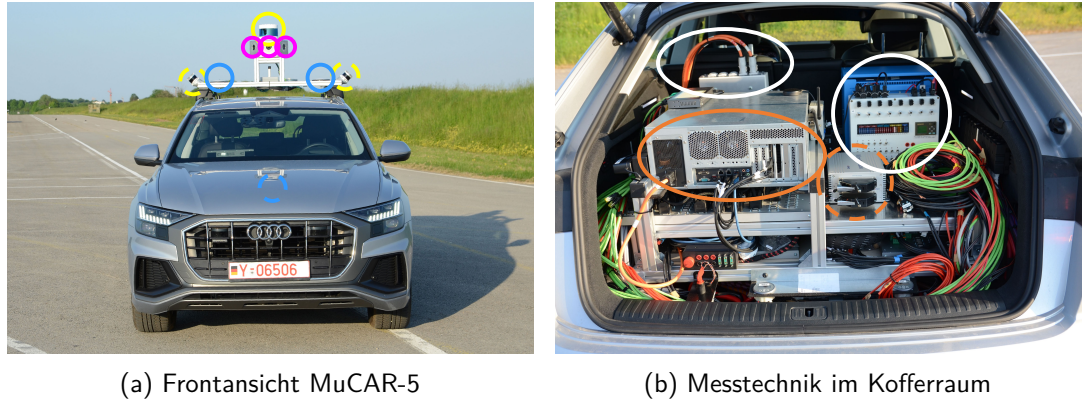
Jahr	CPU-Typ	#CPUs x #Kerne	Frequenz (GHz)	RAM (GB)	GPU
2010	Intel Xeon L5420	2x 4	2,5	8	
2013	Intel Xeon E5-2667	2x 6 + HT	2,9	32	GTX 1060
2018	Intel Xeon Gold 6144	2x 8 + HT	2,666	96	RTX A5000

Kommunikation Zur Kommunikation mit weiteren Fahrzeugen und dem Testgelände-WLAN ist auch MuCAR-4 mit einem Long-Range-WLAN-System sowie einer Anbindung über das Mobilfunknetz ausgestattet.

3.3.5 Sicherheitseinrichtungen

Die Sicherheitseinrichtungen von MuCAR-4 sind ähnlich zum Touareg ausgeführt, siehe Abschnitt 3.2.5 und Abb. 3.10b.

Dem Sicherheitsfahrer stehen zur Übernahme der Kontrolle mehrere Möglichkeiten offen: Ein Notaus-Knopf unterbricht elektrisch alle Ansteuerungen vom Low-Level-System in die Fahrzeugaktorik und stellt diesbezüglich den Serienzustand des Tiguan wieder her, dem Bremsaktor wird eine konstante Spannung, welche zu Nullbetätigung führt, übergeben. Neben einer Betätigung von Brems- und Gaspedal kann im Tiguan auch ein Lenkradeingriff durch den Fahrer detektiert werden, um softwareseitig im Low-Level-Rechner alle Aktorikansteuerungen abzuschalten. Zur Detektion des Lenkradeingriffs wird im Tiguan der in der APA-BS integrierte Lenkmomentsensor [Volkswagen AG, 2007c, S. 25f.] genutzt. Bei der Parametrierung der Schwellwerte ist hier darauf zu achten, dass die Massenträgheit des Lenkrads bei Aktorikeingriff ebenfalls zu gemessenen Momenten führt.

**Abbildung 3.11:**

Versuchsfahrzeug MuCAR-5 mit montierter Sensorik: LiDAR Velodyne VLS-128 (gelb), LiDAR Ouster OS0-32 (gelb gestr.), RGB-Kameras (magenta), GNSS-Antennen (blau), INS (blau gestr.). Im Kofferraum befinden sich Energieversorgung (weiß), High-Level-Rechner (orange), Low-Level-Rechner (orange gestr.).

3.4 Versuchsfahrzeug Audi Q8 – MuCAR-5

Das Munich Cognitive Autonomous Robot Car 5th Generation (MuCAR-5) ist ein Audi Q8 50 TDI quattro und wird – nach langer Spezifikations- und Beschaffungsphase – seit Ende 2021 als drittes Versuchsfahrzeug des Instituts aufgebaut. Langfristig soll damit MuCAR-3 ersetzt werden.

Basis ist hier kein handelsübliches Serienfahrzeug [Audi AG, 2018], wie man es bei jedem Händler kaufen kann, sondern eine Kleinserie, die Audi für eigene Entwicklungs- und Erprobungszwecke im Bereich des Hochautomatisierten Fahrens (HAF) entworfen und gebaut hat. Diese HAF-Kleinserie umfasst bereits zahlreiche Modifikationen, vor allem im Bereich der Energieversorgung sowie in der Möglichkeit, die Kommunikation zu nahezu allen Steuergeräten beeinflussen zu können, aber auch für das Institut eher irrelevante Ausstattungen zur Fahrerbeobachtung. Allein mit diesen Einbauten ist der Kofferraum bereits gut gefüllt, wie in Abb. 3.11 zu erkennen.

Von der Audi-Kleinserie weichen unsere Erweiterungen im Bereich der Messtechnik und Sensorik ab, die zum Großteil an das in den anderen Fahrzeugen verwendete und am Institut erprobte Setup angelehnt sind. Im Rahmen des Umbaus bei Audi wurden nach unserer Spezifikation zahlreiche weitere Leitungen zur späteren Anbindung von Sensorik und Messtechnik im Fahrzeug verlegt, sodass bspw. für den Anschluss von Sensoren auf einem Dachträger kein erneutes Entfernen des Fahrzeughimmels und anderer Verkleidungsteile notwendig ist, sondern bereits alle notwendigen wetterfesten Steckverbinder vorhanden sind.

Die folgenden Abschnitte betrachten die Unterschiede zu MuCAR-3.

3.4.1 Energieversorgung und Kühlung

Der Audi Q8 ist ein MHEV-Fahrzeug (Mild Hybrid Electric Vehicle) mit 48 VDC-Hauptbordnetz, einer Lithium-Ionen-Batterie sowie einem Riemen-Starter-Generator. Dieser kann mit einer Spitzenleistung von 12 kW rekuperieren, aber auch im Fahrbetrieb dauerhaft eine Leistung von mehreren kW bereitstellen. Im Serienfahrzeug wird aus dem 48 VDC-Hauptbordnetz über einen leistungsstarken DC-DC-Konverter das 12 VDC-Bordnetz mit separater Pufferbatterie versorgt, aus diesem werden alle Steuergeräte betrieben.

Für die Messtechnik-Einbauten wurde das System durch Audi um eine Bank aus vier Lithium-Ionen-Batterien erweitert, die im Fahrbetrieb aus dem 48 VDC-Hauptbordnetz geladen werden. Das 48 VDC-Messtechnik-Netz dient in der Messtechnik primär als Zwischenkreis und nicht zum direkten Betrieb von Verbrauchern. Diese werden stattdessen aus einem separaten 12 VDC-Messtechnik-Netz mit weiterer Pufferbatterie versorgt, welches über einen DC-DC-Konverter gespeist wird, oder aus dem 230 VAC-Messtechnik-Netz. Letzteres wird über einen Victron Energy MultiPlus 48/3000/35 [Victron Energy B.V., 2023] bereitgestellt, der sowohl als Wechselrichter als auch als Ladegerät für das 48 VDC-Messtechnik-Netz aus dem Landstrom dient – ist letzterer angeschlossen, wird das 230 VAC-Messtechnik-Netz direkt durchgeschaltet.

Zum Batterie- und Lademanagement wurde durch Audi ein komplexes System aus zahlreichen Relais und Spannungsüberwachungen integriert, welches zum aktuellen Zeitpunkt noch fehleranfällig ist.

Die Kühlung der Messtechnik-Einbauten erfolgt mithilfe eines zusätzlichen Verdampfers mit Ventilator im Kofferraum, dessen Kältemittelleitungen an den Kompressor der serienmäßigen Klimaanlage angebunden wurden.

3.4.2 Aktorik

Der Audi Q8 verfügt für die serienmäßigen ADAS-Funktionen (Fahrerassistenzsystem, engl. *Advanced Driver Assistance System*) wie ACC oder LKAS (Spurhalteassistenzsystem, engl. *Lane Keeping Assist System*) bereits über die Möglichkeit zur Ansteuerung einer Beschleunigungs- und Geschwindigkeitsschnittstelle sowie der Lenkung. Damit entfällt die Nachrüstung separater Aktorik wie in den älteren Fahrzeugen.

Der Audi Q8 verfügt über eine Allradlenkung, d. h. neben der üblichen Vorderachslenkung können auch die Räder der Hinterachse mittels Spindeltrieb und Spurstangen um ca. $\pm 5^\circ$ gelenkt werden [Audi AG, 2015, S. 27ff.]. Dies dient einerseits bei gegensinniger Ansteuerung zur Verkleinerung des Wendekreises und andererseits bei gleichsinniger Ansteuerung zur Erhöhung der Fahrstabilität bei höheren Geschwindigkeiten. Im Serienfahrzeug ist die Allradlenkung standardmäßig aktiv, kann jedoch – bspw. um im autonomen Betrieb vereinfachte Modelle nutzen zu können – abgeschaltet werden.

Im Gegensatz zu Touareg und Tiguan besteht beim Audi Q8 keine mechanische Kopplung zwischen Getriebe und Fahrstufenwählhebel. Letzterer ist nur noch ein elektronisches Eingabegerät, welches CAN-Botschaften versendet. Damit entfällt auch hier die separate Aktorik.

Zur Manipulation der Nachrichten wurden seitens Audi bereits Gateways installiert, die in die Kommunikation in diversen CAN- und Flexray-Bussen eingreifen können. Hier ist allerdings anzumerken, dass derzeit noch Abstimmungen mit Audi für die Inbetriebnahme der Schnittstellen laufen und noch keine Kommunikation möglich ist.

3.4.3 Sensorik

Die Sensorausstattung von MuCAR-5 ist im Bereich der Umfeld- und Lokalisierungssensorik an die in Abschnitt 3.2.3 vorgestellte Sensorausstattung von MuCAR-3 angelehnt.

3.4.3.1 Fahrdynamiksensorik

Wie in Abschnitt 3.4.2 erwähnt, ist noch keine Kommunikation zwischen Low-Level-Rechner und Basisfahrzeug möglich. Nach [Audi AG, 2018] ist allerdings davon auszugehen, dass mindestens die gleichen Messdaten wie in MuCAR-3 zur Verfügung stehen, also Einzelradgeschwindigkeiten, Lenkwinkel und -geschwindigkeit, Einfederungen sowie Beschleunigungs- und Drehratensensoren.

3.4.3.2 Lokalisierungssensorik

INS In MuCAR-5 wurde ein INS vom Typ OxTS RT3000 v3 integriert, welches auf dem Kardantunnel in der Nähe des Fahrzeugschwerpunktes montiert ist.

3.4.3.3 Umfeldsensorik

Kameras Das Fahrzeug wurde mit einem am Institut entwickelten 360°-Kamerasystem auf dem Fahrzeugdach ausgestattet, welches mit dem LiDAR-Sensor vom Typ Velodyne VLS-128 eine Einheit bildet. Das Kamerasystem besteht aus sechs RGB-Industriekameras nach dem GigE Vision-Standard mit einer Auflösung von 5 Mpix, die mit hochwertigen Objektiven jeweils ein HFOV von ca. 80° abdecken. Ein in das System integrierter Mikrocontroller sorgt für eine mit dem LiDAR synchronisierte Ansteuerung der Kameras.

Geplant ist des Weiteren die Integration weiterer Kameras auf einer beweglichen Kameraplattform hinter der Windschutzscheibe (angelehnt an MarVEye-8). Diese befindet sich derzeit in der Fertigung.

LiDAR Der Audi Q8 verfügt serienmäßig über einen LiDAR-Sensor vom Typ Valeo Scala, der in vier Ebenen misst und mit seinem integrierten Objekttracking primär für einen Stauassistenten genutzt wird. Aufgrund nicht offengelegter Schnittstellen sind die Rohdaten nicht verfügbar.

Analog zu MuCAR-3 wurden drei LiDAR-Sensoren auf dem Fahrzeugdach montiert: der zentrale Fernbereichssensor vom Typ Velodyne VLS-128 bildet hier eine Einheit mit dem 360°-Kamerasystem, zusätzlich wurden zwei seitlich gekippte Nahbereichssensoren vom Typ Ouster OS0-32 integriert. Aus EMV-Gründen mussten die Ouster OS0-32 in möglichst großer Entfernung zu den GNSS-Antennen montiert werden, da eine Abschirmung hier nicht möglich ist.

Radar Der Audi Q8 ist serienmäßig mit mehreren Radar-Sensoren ausgestattet, einerseits im Kühlergrill für Anwendungen wie ACC und andererseits an den Fahrzeugecken für den Totwinkelassistenten. Ohne separate Verträge mit dem Sensorhersteller sind hier keine Rohdaten verfügbar.

Daher ist für MuCAR-5 geplant, ein zu MuCAR-3 identisches Setup aus Nah- und Fernbereichssensoren der Herstellers smart microwave sensors GmbH zu ergänzen. Die notwendige Verkabelung wurde bereits eingezogen.

3.4.4 Rechnerarchitektur, Kommunikation

Die Rechnerarchitektur von MuCAR-5 orientiert sich ebenfalls an den in Abschnitt 3.1 aufgezeigten Gemeinsamkeiten und ist in Low-Level- und High-Level-Rechner unterteilt.

Low-Level-Rechner Hier wurde eine dSPACE MicroAutoBox II 1401/1513/1514 integriert, die nicht nur über CAN-Bus-Schnittstellen verfügt, sondern auch über FlexRay und Ethernet. Die Anbindung an das Fahrzeug erfolgt über Gateways, die Inbetriebnahme steht allerdings noch aus.

High-Level-Rechner Der High-Level-Rechner ist weitgehend identisch zu MuCAR-3 ausgelegt (vgl. Tabelle 3.2), einziger Unterschied ist die Energieversorgung. Diese erfolgt hier wie im Büro aus dem 230 VAC-Messtechnik-Netz und nicht mit einem speziellen 12 VDC-ATX-Netzteil.

3.4.5 Sicherheitseinrichtungen

Der Audi Q8 verfügt in der HAF-Kleinserie über ein mehrstufiges Not-Aus-Konzept.

Für die Kontrollübernahme durch den Sicherheitsfahrer dient ein Not-Aus-Schalter, der nur die Ansteuerung der Aktorik bzw. die Modifikation der Fahrzeugbusse

abschaltet. Ein weiterer Not-Aus-Schalter trennt die komplette Messtechnik auf beiden Spannungsschienen von der Versorgung.

Aufgrund modifizierter Software in den Motor- und Lenkungssteuergeräten sind hier zwei weitere Not-Aus-Schalter angebracht, mit denen diese Teilsysteme im Fehlerfall abgeschaltet werden können. Hier ist zu beachten, dass bei Auslösung weder Bremskraftverstärker noch Lenkunterstützung verfügbar sind, sodass der Sicherheitsfahrer das Fahrzeug nur unter erhöhtem Krafteinsatz in einen sicheren Zustand bringen kann.

Weitere softwarebasierte Abschaltungen, bspw. bei Pedalbetätigung oder Lenkeingriff, sind innerhalb des Low-Level-Rechners vorgesehen, aber bislang aufgrund der Schnittstellenthematik noch nicht integriert.



(a) Fahrzeug mit montiertem Leitstand



(b) Komponenten-Überblick der Sensorik

Abbildung 3.12:

Technologieträger Unbemanntes Landfahrzeug (TULF) in den ersten Jahren der Studie.

3.5 Technologieträger Unbemanntes Landfahrzeug

Das Projekt Technologieträger Unbemanntes Landfahrzeug (TULF) ist eine vom Bundesamt für Ausrüstung, Informationstechnik und Nutzung der Bundeswehr (BAAINBw) initiierte F&T-Studie (Forschung und Technologie) der Stufe 2⁶. Ziel ist die Erprobung von Teleoperation und vor allem von Autonomiefunktionen in Transportszenarien aus dem militärischen Umfeld, wobei die einzelnen Funktionen von verschiedenen Projektpartnern sowohl aus dem universitären als auch aus dem industriellen Bereich beigesteuert werden. Zusätzlich zu vollautonomen Betriebsmodi wie „Wegpunktfolgen“ oder „Konvoi“ dienen die autonomen Fähigkeiten des Fahrzeugs auch zur Entlastung des Bedieners in der Teleoperation und als Rückfallebene bei Ausfall der Funkfernsteuerung.

Aufgrund der gewählten Szenarien ist der Einsatz einer geländetauglichen Plattform erforderlich, hier kommt ein Lastkraftwagen (LKW) der Firma Rheinmetall MAN Military Vehicles GmbH (RMMV) vom Typ HX58 mit 6x6-Antrieb zum Einsatz. Der Umbau der Plattform für den DbW-Betrieb erfolgte unter Federführung von Rheinmetall Landsysteme GmbH (RLS) in Kooperation mit der PARAVAN GmbH (PARAVAN), die anfangs das DbW-System PARAVAN SpaceDrive integrierte und später auf PARAVAN SpaceDrive II umrüstete. Die Ausrüstung mit Sensorik sowie den zentralen Rechnern für die Sensordatenverarbeitung erfolgte durch Diehl BGT Defence GmbH & Co. KG (DBD). Abbildung 3.12 zeigt das Fahrzeug in den ersten Jahren der Studie.

Im Zeitraum von 2011 bis 2020 wurde eine Reihe von Studien auf dem gleichnamigen Versuchsfahrzeug TULF durchgeführt und dieses dafür regelmäßig in Hard- und Software weiterentwickelt.

⁶Vor der Realisierung bzw. Beschaffung neuer Systeme in der Bundeswehr erfolgen im Regelfall Studien, um die notwendige Technologie zu entwickeln und zu erproben. Stufe 1 umfasst die angewandte Grundlagenforschung, darauf folgt in Stufe 2 die anwendungsnahe wehrtechnische Forschung und Technologie. Abschließend erfolgen in Stufe 3 system- und lösungsorientierte Untersuchungen, bevor eine Realisierungsphase startet.

3.5.1 Energieversorgung

Wie im LKW-Bereich üblich, verfügt der TULF über ein 24 VDC-Bordnetz mit einer leistungsstarken Lichtmaschine. Dieses ist ohne Modifikationen in der Lage, die komplette Messtechnik in zu betreiben.

3.5.2 Aktorik

Lenkung Für die Ansteuerung der Lenkung kommt das redundant ausgelegte Lenksystem des PARAVAN SpaceDrive II zum Einsatz, dessen Elektromotoren über eine Magnetkupplung mechanisch an die Lenksäule gekoppelt sind. Der Motor ist ohne explizites Lösen der Magnetkupplung nicht übersteuerbar. Die Ansteuerung einer Lenkwinkelschnittstelle erfolgt vom Low-Level-Rechner per CAN-Bus. Hier ist zu beachten, dass eine nicht zu vernachlässigende Totzeit $\gg 100$ ms besteht und die Lenkung über eine stark nichtlineare Kennlinie zwischen Lenkradwinkel und Winkel am Rad verfügt. Innerhalb des Low-Level-Systems erfolgt eine Linearisierung.

Bremse und Gas Für die wechselweise Betätigung von Brems- und Gaspedal kommt ein gemeinsamer, redundant ausgelegter Elektromotor des PARAVAN SpaceDrive II zum Einsatz. Konstruktiv wird die gleichzeitige Betätigung verhindert, resultiert jedoch dadurch in einer Totzeit zwischen vollständig gelöster Bremse und Betätigung des Gaspedals. Insgesamt ist diese Ansteuerung eher träge.

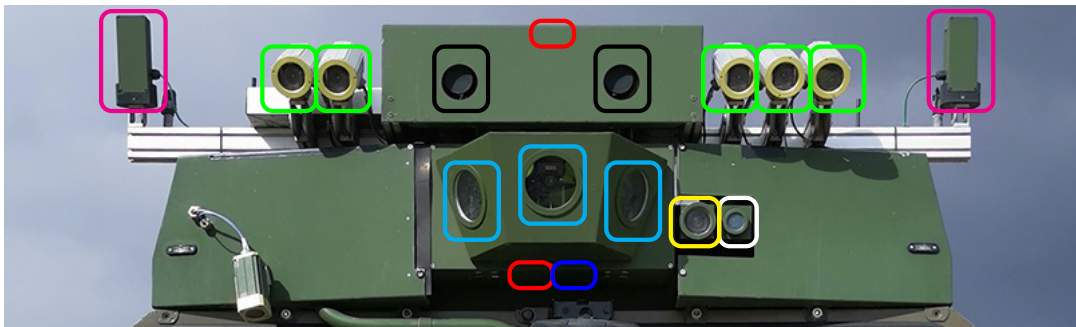
Gangschaltung Der HX58 verfügt über ein automatisiertes Schaltgetriebe. Das Einlegen der gewünschten Fahrstufe (D/R/N für den Normalbetrieb sowie Dx/Rx für den Geländebetrieb) erfolgt im manuellen Betrieb mit einem elektronischen Drehschalter. Im automatisierten Betrieb werden dessen Signale vom Low-Level-System simuliert.

Feststellbremse Die Feststellbremse wird durch das Öffnen eines Entlüftungsventils im Bremskreis angelegt, die Ansteuerung des Ventils erfolgt vom Low-Level-Rechner. Parallel ist die manuelle Bedienung der Feststellbremse weiter möglich. Ist die Bremse händisch eingelegt (also entlüftet), kann diese vom Rechner nicht gelöst werden.

3.5.3 Sensorik

3.5.3.1 Fahrdynamiksensorik

Geschwindigkeit Zu Projektbeginn wurde die Geschwindigkeit aus den Raddrehzahlsensoren nur im autonomen bzw. teleoperierten Betrieb vom Low-Level-System



(a) Dachaufbau mit hochauflösender Fisheye-Farbkamera (gelb), drei Farbkameras (cyan), Wärmebildkamera Xenics Gobi (weiß), Wärmebildkamera-Stereo ESW Variocam (schwarz), Hyperpektralkameras (grün), UWB-Sensoren (magenta), INS XSens MTi-G (rot, in der Sensorbay) mit GNSS-Antenne (rot, oben), 1-Achs-Kreisel KVH DSP-3000 (blau).



(b) Front



(c) linke Seite



(d) Heck



(e) rechte Seite

Abbildung 3.13:

Sensorik des TULF gegen Ende der Studie. LiDAR-Sensoren: Velodyne HDL-32E (gelb), Velodyne VLP-16 Puck (weiß). Radar-Sensoren: Smart Micro (rot), Continental ARS 308 (cyan), Continental ARS 408 (grün). INS: OxTS RT3003 v2 (magenta) mit Novatel GNSS-Antennen (magenta), Übergrund-Geschwindigkeitssensor (blau).

an das High-Level-System kommuniziert. Hier handelte es sich um die mittlere Geschwindigkeit der Vorderachse in grober zeitlicher Auflösung von 10 Hz.

Im weiteren Projektverlauf konnten im Rahmen dieser Arbeit Erkenntnisse gewonnen werden, welche weiteren Signale auf den CAN-Bussen des Fahrzeugs verfügbar sind (im Nutzfahrzeugsbereich ist eine standardisierte Kommunikation gemäß SAE J1939-71 [SAE International, 2011] zwischen Basisfahrzeug und Zusatzsystemen üblich), und diese zusammen mit RLS dekodiert und in Testfahrten verifiziert werden. Das Low-Level-System konnte so angepasst werden, dass in jedem Betriebsmodus vier Einzelrad-Geschwindigkeiten in besserer Auflösung von $\frac{1}{256}$ km/h und mit 50 Hz bereitgestellt werden, die Tandem-Hinterachse wird dabei auf eine virtuelle Achse reduziert. Im Gegensatz zu MuCAR-3 ist hier kein direkter Abgriff der Sensoren möglich gewesen, sodass keine ungefilterten Impulse des Radencoders zur Verfügung stehen. Zusätzlich wird die Geschwindigkeit des TULF mit einem Übergrund-Radar-Sensor vom Typ Speed Wedge des Herstellers MSO Meßtechnik und Ortung GmbH gemessen. Dieser ist speziell für die Verwendung an „Off-Highway-Fahrzeugen“ vorgesehen.

Der Sensor sendet schwache Mikrowellen (Radar) aus und mißt die Frequenzverschiebung bei den empfangenen Reflexionen (Doppler-Effekt). Aus dieser Frequenzverschiebung wird die relative Geschwindigkeit gegenüber dem Boden ermittelt. (MSO Meßtechnik und Ortung GmbH [2011])

Der Sensor liefert laut Datenblatt ein geschwindigkeitsabhängiges Rechteck-Puls-Signal mit 130 Pulsen/m. Zusätzlich wird das Signal auf einer seriellen RS232-Schnittstelle mit einer Frequenz von 5 Hz bereitgestellt. Durch die direkte Messung der Geschwindigkeit gegenüber dem Boden wird die Messung nicht durch Radschlupf oder Einsinken des Rades im Boden beeinflusst, beides Störungen, die eine Messung mit Raddrehzahlsensoren wie im PKW-Bereich (Personenkraftwagen) üblich verfälschen würden.

Im Laufe der Erprobung wurde festgestellt, dass dieser Sensor Probleme bei nasser ebener Fahrbahn oder bei Pfützen hat, dort ist keine Geschwindigkeitsmessung möglich, und der Sensor liefert Falschmessungen von 0 km/h.

Lenkwinkel Ein in der Lenkaktorik des PARAVAN-DbW-Systems integriertes Potentiometer kommt als absoluter Lenkradwinkelsensor zum Einsatz. Innerhalb des Low-Level-Systems erfolgt eine Umrechnung auf den Lenkwinkel an der Vorderachse bzw. des fiktiven mittleren Vorderrades. Im Gegensatz zu den institutseigenen Versuchsfahrzeugen MuCAR-3 und MuCAR-4 ist die Lenkgeometrie beim TULF jedoch nur rudimentär bekannt, da die Parameter aus einzelnen händischen Winkelmessungen sowie stationären Kreisfahrten bestimmt wurden. Als großer Nachteil ist zu nennen, dass der Lenkwinkel in der Anfangszeit nur bei aktiver Aktorik und geschlossener Magnetkupplung gemessen werden konnte, d. h. während manueller Fahrten standen keine Lenkwinkelinformationen zur Verfügung. Mit einer Überarbeitung des DbW-Systems konnte dieses Problem beseitigt werden.

3.5.3.2 Lokalisierungssensorik

IMU und GPS In die Sensorbay auf dem Dach der Fahrerkabine wurde ein INS vom Typ XSens MTi-G Xsens Technologies B.V. [2010a] integriert. Die Xsens wird im TULF nicht als INS betrieben, sondern als separierte IMU und GPS-Empfänger. Die IMU besteht intern aus drei MEMS-Kreisels, drei MEMS-Beschleunigungssensoren sowie einem 3-Achsen-Magnetometer. Letzteres ist in einem Metallgehäuse mit vielen Elektronikkomponenten mit starken Störungen behaftet, und auch die MEMS-Sensoren sind mit einem Bias behaftet. Die Messungen werden auf einer RS232-Schnittstelle ausgegeben, über einen Messdatenkonverter erfasst und per Ethernet an den High-Level-Rechner übertragen.

1-Achs-Kreisel Aufgrund von Herausforderungen in den DBD-Lokalisierungsalgorithmen mit den Bias-behafteten Messdaten der XSens MTi-G wurde im Projektverlauf für die Messung der Gierrate ein weiterer Sensor ergänzt. Mit dem KVH DSP-3000, einem faseroptischer 1-Achs-Kreisel, können Gierratenmessungen mit sehr geringem Rauschen und zu vernachlässigendem Bias gewonnen werden. Auch diese Messungen werden über einen Messdatenkonverter erfasst und per Ethernet an den High-Level-Rechner übertragen.

INS Für Ground-Truth-Anwendungen wurde 2016 im Rahmen dieser Arbeit eine INS vom Typ OxTS xNAV 550 temporär in den TULF-integriert, um einerseits die Eigenbewegungsschätzung und andererseits die Ergebnisse der Kollegen aus dem Fahrzeugtracking zu evaluieren. Im Vergleich zur Sensorik in MuCAR-3 ist diese unter schlechten GNSS-Bedingungen weniger präzise, unter guten Bedingungen jedoch vergleichbar.

Seitens DBD erfolgte im weiteren Projektverlauf die permanente Integration eines zum OxTS RT3003 v2 baugleichen Sensors, um damit auf alle firmeneigenen Algorithmen zur Eigenbewegungsschätzung auf Basis günstiger Sensorik verzichten zu können.

3.5.3.3 Umfeldsensorik

Abbildung 3.13 gibt einen grafischen Überblick über die am TULF vorhandene Sensorik im Jahr 2020.

Kameras Für das Fahrzeugtracking wurde von TAS die sog. Fisheye-Kamera spezifiziert und integriert, um eine Konvoi-Funktionalität mit einer einzelnen feststehenden Kamera realisieren zu können. Für die Fisheye-Kamera kommt eine Standard-Industriekamera nach dem GigE Vision-Standard vom Typ Basler ace acA2040-25gc (Auflösung 4 Mpix) mit einem Ultraweitwinkelobjektiv (Typ Kowa LM6HC, $f = 6 \text{ mm}$) zum Einsatz, damit ergeben sich ein HFOV und VFOV von je ca. 90° .

Zusätzlich kam zu Projektbeginn eine Wärmebildkamera vom Typ Xenics Gobi 640 mit VGA-Auflösung zum Einsatz, die später durch eine ESW Variocam mit XGA-Auflösung abgelöst wurde.

Der TULF ist mit einer Vielzahl weiterer Kameras ausgestattet, die meisten dienen zur Teleoperation und werden hier nicht weiter betrachtet.

LiDAR An den vorderen Fahrzeugecken sind zwei LiDAR-Sensoren vom Typ Velodyne HDL-32E [Velodyne LiDAR, Inc., 2017] integriert, deren HFOV aufgrund von Verdeckung auf je ca. 270° limitiert ist. Im weiteren Projektverlauf wurden zwei Velodyne VLP-16 Puck [Velodyne LiDAR, Inc., 2018] am Fahrzeugheck integriert, deren Sichtfeld ebenfalls auf ca. 270° limitiert ist.

Radar Das Fahrzeug verfügt an Front und Heck über jeweils einen Radar-Sensor vom Typ Continental ARS 308. Später wurde der modernere Continental ARS 408 an der Fahrzeugfront ergänzt. Beide Sensortypen liefern entweder Detektionen oder Objektlisten, aber nicht gleichzeitig, und können zwischen einem Nahbereich mit breiterem HFOV und einem schmalen Fernbereich umschalten.

Das Rundum-Radar-System der smart microwave sensors GmbH besteht aus neun Einzelsensoren und kam ursprünglich für einen Totwinkelassistenten zum Einsatz. Aufgrund von Hardwareausfällen kommt es nicht mehr zum Einsatz.

3.5.4 Rechnerarchitektur, Kommunikation

Low-Level-Rechner Hier wurde von RLS ein proprietäres System integriert, welches die Hauptkomponente des sog. B-Kit darstellt. Dieser Rechner aggregiert Daten von den verschiedenen CAN-Bussen des HX58, des DbW-Systems PARAVAN Space-Drive II sowie von analogen und digitalen Eingängen und liefert diese Messdaten auf einem separaten CAN-Bus an den High-Level-Rechner. Dabei findet eine Signalkonditionierung statt, bspw. eine Kompensation der nichtlinearen Lenkkennlinie. Der Low-Level-Rechner kann auch untere Reglerschichten ausführen, z. B. eine Geschwindigkeitsregelung.

High-Level-Rechner Im Vergleich zu MuCAR-3 ist die High-Level-Rechnerarchitektur beim TULF komplizierter aufgebaut. Zu Projektbeginn kamen mehrere kleine PCs mit Ubuntu 14.04, später 20.04. zum Einsatz, die für einzelne Aufgaben zuständig waren und über verschiedene Ethernet-Netze und die Middleware ROS kommuniziert haben. Da sie über keine Möglichkeiten zur Visualisierung verfügten, kam für die komplette TAS-Software ein kompaktes Notebook (Intel Core i7-3520M, 2.90 GHz) zum Einsatz, auf dem der komplette Softwarestack für die Konvoi-Funktionalität ausgeführt wurde. Diese besteht u. a. aus den Sensoreinzügen, dem Fahrzeugtracking, der Spurgenerierung, einer Eigenbewegungsschätzung, der Regelung sowie einer Middleware-Schnittstelle (siehe Abschnitt 4.1.4).

Später wurde ein leistungsfähiger Server ergänzt, der nahezu alle vorherigen Rechner ersetzte. Zusätzlich wurden – u. a. auf Forderung von TAS – zwei Monitore oberhalb der Windschutzscheibe integriert, um eine direkte Visualisierung des Systemzustands und der wahrgenommenen Umgebung für den Operator und Sicherheitsfahrer zu ermöglichen.

Die Kommunikation zwischen High-Level- und Low-Level-Rechner erfolgt – mit dem Umweg über einen Ethernet-CAN-Konverter – per CAN-Bus. Ebenso sind zahlreiche Sensoren über solche Konverter angebunden. Dies erhöht die Komplexität und Latenz und ist zusätzlich eine Quelle für Jitter.

3.5.5 Sicherheitseinrichtungen

Der Sicherheitsfahrer des TULF kann nur nach Betätigung des Schalters für die Magnetkupplung der Lenkaktorik die Kontrolle übernehmen, er hat sonst keine Möglichkeit zur Übersteuerung. Wird dieser Eingriff detektiert, so schaltet das DbW-System alle Eingriffe ab und übergibt die Kontrolle.

Mehrere Not-Stopp-Taster am Fahrersitz, im Leitstandscontainer, optional auch außen am Fahrzeug oder per Funkfernbedienung, lösen direkt über das DbW-System einen Nothalt aus.



- (a) Am Dach der Fahrerkabine befindet sich die „Sensorbay“ (magenta) mit integrierten Farbkamera- und Inertialsensoren, daneben ist die Wärmebildkamera montiert (gelb). Dahinter befinden sich die Antennen für die V2VKommunikation (rot). Der GPS-Empfänger befindet sich oberhalb der Sensorbay. In der Stoßstange unterhalb des ausklappbaren Trittbügels befinden sich die LiDAR- (cyan) und Radar-Sensoren (grün).
- (b) Fahrzeugheck mit Einbauposition des Übergrund-Geschwindigkeitssensors Dickey-john RADAR III hinter der Tandem-Hinterachse (magenta).

Abbildung 3.14: Versuchsfahrzeug StrAsRob mit montierter Sensorik.

3.6 StrAsRob

Das Projekt StrAsRob (Straßentransport mit Assistenzfunktionen von Robotern) ist eine vom BAAINBw initiierte F&T-Studie der Stufe 3⁷, die von 2012 bis 2015 bearbeitet wurde. Im Gegensatz zum TULF, bei dem der Schwerpunkt auf der Autonomie und dem prinzipiell möglichen unbemannten Fahren lag, sollte hier die *Folgefunktion* erweitert und in der Robustheit gesteigert werden, um als Assistenzsystem und primär zur Entlastung des MKF (Militärkraftfahrer) zu dienen.

Als Plattform kommt mit dem RMMV HX58 derselbe Fahrzeugtyp zum Einsatz, auch die Grundumbauten mit DbW-System und Low-Level-System erfolgt in Anlehnung an den TULF. Als entscheidender Unterschied ist der Übergang zu seriennäherer Sensorik zu nennen, um in Hinblick auf Stückzahlen größer eins eine Kostensenkung zu erzielen und zusätzlich diese Sensorik besser in das Chassis des HX58 zu integrieren. Dieses Fahrzeug wurde im Anschluss in anderen Studien weiterverwendet, u. a. in Kombination mit dem TULF.

⁷vgl. Fußnote 6 auf Seite 72.

3.6.1 Sensorik

3.6.1.1 Fahrdynamik- und Lokalisierungssensorik

Lenkung und Geschwindigkeit Neben den Informationen aus dem Basisfahrzeug und dem DbW-System (Lenkwinkel, Einzelradgeschwindigkeiten) stehen auch Messwerte eines Übergrund-Geschwindigkeitssensors zur Verfügung. Aufgrund der Herausforderungen mit nassen Untergründen beim im TULF verwendeten Sensor kommt hier ein Alternativprodukt zum Einsatz: der DICKEY-john RADAR III [DICKEY-john, 2020]. Der Messbereich beginnt bei $\approx 0,5$ km/h, damit kann der Sensor extrem langsame Bewegungen im Bereich des Stillstands nicht erfassen. Die Messgenauigkeit ist für kleine Geschwindigkeiten ($v \leq 3,2$ km/h) mit $\pm 3\%$ angegeben, darüber mit $\pm 1\%$. Da der Sensor nur ein geschwindigkeitsproportionales Frequenzsignal erzeugt, ist ein separater Rechner zur Signalumsetzung erforderlich.

IMU und GPS In die Sensorbay auf dem Dach der Fahrerkabine wurde die bereits bekannte INS vom Typ XSens MTi-G integriert und liefert unabhängige Messungen aus IMU und GPS-Empfänger. Zusätzlich kommt auch hier der faseroptische 1-Achs-Kreisel vom Typ KVH DSP-3000 zum Einsatz.

3.6.1.2 Umfeldsensorik

Kameras In der Sensorbay am Dach der Fahrerkabine befinden sich drei RGB-Farbkameras, die gemeinsam ein HFOV von 170° abdecken. Ursprünglich wurden hier drei identische Kameras vom Typ Basler ace acA1300-30gc mit einer Auflösung von 1,25 Mpix und günstigen Objektiven mit einem HFOV von 58° verwendet. Die mittige Kamera wurde im Laufe des Projekts gegen eine höherwertige ausgetauscht, da dieses der wichtigste Sichtbereich in der Folgefunktion ist: eine Basler ace acA2000-50gc mit einer Auflösung von 2 Mpix und einem hochwertigen Objektiv deckt einen HFOV von 68° ab. Die gewählte Überlappung ermöglicht einen sanften Übergang der Bildverarbeitungsalgorithmen zwischen den Bildern.

Ergänzend kommt eine nach vorn gerichtete Wärmebildkamera vom Typ Xenics Gobi 640 zum Einsatz. Diese basiert auf einem ungekühlten Mikrobolometer-Sensor und liefert unkalibrierte, d. h. relative Temperaturinformationen.

LiDAR Statt mehrerer teurer LiDAR-Sensoren vom Typ Velodyne HDL-32E und Velodyne VLP-16 Puck außerhalb der Fahrzeugkontur wurde ein seriennaher 8-Ebenen-LiDAR-Sensor vom Typ Ibeo Lux 8L [Ibeo Automotive Systems GmbH, 2012] ausgewählt und in die Frontstoßstange integriert. Das VFOV beträgt $6,4^\circ$, das HFOV umfasst den Bereich von -60 bis 50° . Abwechselnd werden die oberen 4 Ebenen für ein integriertes Objekttracking und die unteren 4 Ebenen für eine mögliche Hinderniserkennung abtastet. Dieser Sensor liefert die Messdaten, bestehend aus

Objektliste und Punktwolke, über eine Ethernet-Schnittstelle an den High-Level-Rechner. Für das integrierte Tracking ist eine Kenntnis der Fahrzeugbewegung notwendig, diese wird dem Sensor über eine CAN-Schnittstelle bereitgestellt.

Radar Ebenfalls in die Stoßstange integriert wurde ein Radar-Sensor vom Typ Continental ARS 308. Dieser liefert umschaltbar entweder die stärksten Detektionen oder bereits intern getrackte Objekte. Analog zum Ibeo Lux 8L benötigt auch der Radar eine Kenntnis der Fahrzeugbewegung, um intern Objekte tracken zu können.

Durch die niedrige Montageposition beider Sensoren erscheint von vorausfahrenden Fahrzeugen mit großer Bodenfreiheit oftmals nicht das Heck, sondern erst die Hinterachse in den Sensordaten.

3.6.2 Rechnerarchitektur, Kommunikation

Die Architektur des High-Level-Systems wurde analog zum TULF ausgeführt: mehrere kleine PCs⁸ sind jeweils für einzelne Aufgaben zuständig und kommunizieren über verschiedene Ethernet-Netze und die Middleware ROS. Auch die TAS-Software wurde modularisiert, um in der verteilten Architektur laufen zu können, allerdings unter dem Nachteil der schlechteren Wartbarkeit.

Neben der fahrzeuginternen Kommunikation wurde der StrAsRob mit einer Variante der V2V-Kommunikation (Fahrzeug-zu-Fahrzeug, engl. *Vehicle-to-Vehicle*) ausgestattet. Dies ist in der sog. *Leader-Tracking-Box* begründet, die zum Gesamtsystem StrAsRob gehört und in wechselnde Führungsfahrzeuge eingerüstet werden kann: eine Kombination aus IMU, GPS-Empfänger, Rechner, Funkmodule und einem Akku in einer portablen Box. Auf dieser Hardware werden Algorithmen zur Selbstlokalisierung ausgeführt und das Ergebnis per Funk übertragen, um als Stützung der Folgefunktion bei Sichtverlust zu dienen. Die notwendigen Kommunikationsmodule für den Schmalbandfunk sind auch im StrAsRob selbst integriert, die Daten werden – mit dem Umweg über einen Ethernet-RS232-Konverter – an den High-Level-Rechner übergeben.

3.6.3 Sicherheitseinrichtungen

Neben den aus dem TULF bekannten Sicherheitseinrichtungen wurde im StrAsRob von einem Projektpartner ein Radar-basierter Notbremsassistent entwickelt und integriert. Dieser sollte im Idealfall die Kollision mit einem stehenden Hindernis vermeiden. In Feldtests mit ungefährlichen aber gut in den Sensordaten sichtbaren Hindernissen kam es jedoch mehrfach zu Kollisionen, wenn auch mit reduzierter Aufprallenergie. Zusätzlich gab es eine Reihe an Falschauslösungen, sodass der Notbremsassistent für alle weiteren Testfahrten deaktiviert wurde.

⁸Im Gegensatz zu TULF wurde der StrAsRob nicht mehr auf einen zentralen schnellen Rechner umgerüstet, da diese Studie schon zuvor beendet war.



(a) *Eva* bei der automatisierten Auslieferung eines Pakets. (b) Autonome Begegnung von *Adam* (li.) und *Eva* (re.).

Abbildung 3.15: StreetScooter-Versuchsfahrzeuge in einem Industrieprojekt.

3.7 StreetScooter B14 Work

Im Rahmen eines mehrjährigen Industrieprojekts, welches das Institut TAS zusammen mit dem An-Institut ITIS GmbH für die Firma StreetScooter GmbH (StS) durchgeführt hat, wurden ab Ende 2014 zwei Fahrzeuge für die automatisierte Auslieferung von Paketen aufgebaut und mit Software zur Autonomie ausgestattet, *Eva* und *Adam*.

Hier kommt als Basisfahrzeug das BEV (Battery Electric Vehicle) *StreetScooter B14 Work* zum Einsatz, allerdings als Teil einer frühen Kleinserie. Dieser Fahrzeugtyp sowie seine Nachfolger sind primär bei den Zustellern der Deutschen Post bzw. DHL anzutreffen.

Da StS keine Vorerfahrung in Bereich autonomer Fahrzeuge besaß, sollten die Fahrzeuge auf den Erfahrungen des Institut basieren und für den speziellen Anwendungsfall weiterentwickelt werden. Die mechanische oder elektrische Umsetzung war im Regelfall Aufgabe von StS, von TAS-Seite erfolgten regelmäßige Reviews und Verbesserungsvorschläge sowie die komplette softwaretechnische und algorithmische Entwicklung.

Eine ausführliche Beschreibung der in diesem Projekt realisierten Funktionen ist in [Heinrich et al., 2018] zu finden. Anzumerken ist an dieser Stelle, dass zwischen „Erstkontakt zum Fahrzeug“ und „erster autonomer Fahrt mit Auslieferung“ nur 12 Monate lagen.

3.7.1 Energieversorgung

Für die Messtechnik wurden DC-DC-Konverter integriert, die das 12 VDC-Messtechnik-Netz mit eigener Pufferbatterie aus der Hochvoltbatterie versorgen. Zusätzlich kann dieses Netz mit Landstrom geladen werden. Alle dauerhaften Verbraucher werden aus dem 12 VDC-Messtechnik-Netz versorgt. Über einen Sinus-Wechselrichter wird ein 230 VAC-Messtechnik-Netz bereitgestellt, es dient vor allem für Entwicklernotebooks und andere temporäre Geräte.

3.7.2 Aktorik

Für das erste Fahrzeug „Eva“ wurde ein komplettes DbW-System vom Typ PARAVAN SpaceDrive II integriert, dessen Aktoren Bremspedal und Lenkung bedienen. Dieses System wird in Abschnitt 3.5.2 detaillierter vorgestellt.

Aufgrund der nicht immer positiven Erfahrungen mit dem Einsatz von PARAVAN SpaceDrive II im Robotik-Einsatz wurde das zweite Fahrzeug „Adam“ abweichend aufgebaut: es erhielt in Anlehnung an MuCAR-4 eine Bremsaktorik der Firma Bever, für die Lenkung kam eine elektromechanische Servolenkung eines Zulieferers zum Einsatz.

In diesen frühen Kleinserien-Fahrzeugen wurde von Seiten des Antriebsstrang-Zulieferers die Drehmoment-Schnittstelle zur E-Maschine (Elektromotor und -generator) selbst StS gegenüber nicht offengelegt, daher musste hier als Rückfallebene die elektronische Simulation des Gaspedals mithilfe des Low-Level-Systems zum Einsatz kommen. Diese Lösung ist nur suboptimal, da nun weitere Steuergeräte involviert sind, deren Verhalten sorgsam identifiziert und modelliert werden musste.

Das BEV verfügt über ein Ein-Gang-Getriebe, daher entfallen alle mechanischen Schaltvorgänge, es ändert sich nur die Drehrichtung der E-Maschine. Beim serienmäßigen Hebel zur Fahrtrichtungswahl handelt es sich nur um einen Schalter, dessen Spannungssignal im autonomen Betrieb vom Low-Level-Rechner simuliert wird.

Die Ansteuerung der Sekundärfunktionen wie Blinker, Licht und Hupe wurde auf dem Low-Level-Rechner realisiert.

Zusätzlich verfügt *Eva* zur Übergabe von Paketen über einen seitlichen Roboterarm mit zwei Freiheitsgraden (siehe Abb. 3.15a). Die Realisierung der Mechanik und Ansteuerung der Antriebe erfolgten durch StS und einen externen Partner, die Einbindung in das Gesamtsystem lag auf Seiten von TAS. Im Rahmen dieser Arbeit wurde der Arm modelliert, um seine erreichbaren Positionen und benötigten Freiräume in der Routen- und Bahnplanung für das Fahrzeug korrekt berücksichtigen zu können.

3.7.3 Sensorik

Fahrdynamiksensorik Hier stehen auf dem CAN-Bus Antriebsmoment und Drehzahl der E-Maschine bereit sowie der Lenkradwinkel aus dem DbW. Die nichtlineare Umrechnung auf den Lenkwinkel am virtuellen mittleren Vorderrad wurde mithilfe von Kreisfahrten und den Beziehungen des kinematischen Einspurmodells bestimmt.

Die Geschwindigkeitssignale der vier Räder werden einerseits über den CAN-Bus übermittelt, zusätzlich werden sie latenzfrei direkt am Sensor abgegriffen, mit einer Anlogschaltung aufbereitet und als Digitalsignale an das Low-Level-System übergeben – ähnlich wie in MuCAR-3. Die Auflösung beträgt hier ca. 22 Pulse/m und ist im Vergleich zu MuCAR-3 ungenauer. Für die im Projekt geforderte Positioniergenauigkeit von 1 bis 2 cm in Längsrichtung und einem Winkelfehler von

1° – und dementsprechend für langsamstes Fahren mit schleifender Bremse wie in [Heinrich, 2023] erläutert – genügte die Auflösung der Raddrehzahlsensoren nicht. Mithilfe der festen Getriebeübersetzung von der E-Maschine bis zum Rad und der hochfrequent und hochauflösend gemessenen Motordrehzahl konnte ein sehr präzises Ersatz-Geschwindigkeitssignal berechnet werden.

Lokalisierungssensorik Hier kamen in beiden Fahrzeugen INS vom Typ OxTS RT3003 v2 zum Einsatz, die auf dem Testgelände von StS das Korrektursignal einer eigenen RTK-Referenzstation nutzen und damit eine Lokalisierungsgenauigkeit von ca. 2 cm bereitstellen konnten.

Umfeldsensorik Zur Umfeldwahrnehmung (Hinderniserkennung, Objekttracking, bei Eva zusätzlich Arbeitsraumüberwachung für einen Roboterarm) kamen an den vorderen Ecken angebrachte LiDAR-Sensoren vom Typ Velodyne HDL-32E zum Einsatz, zusätzlich ein Velodyne VLP-16 Puck hinten rechts. Die Auswahl und Platzierung der Sensorik erfolgte für den Einsatzbereich im suburbanen Raum und Geschwindigkeiten bis 30 km/h, mit dem Fokus auf Hinderniserkennung, möglichst keinem toten Winkel sowie der Möglichkeit zum Objekttracking in mittleren Entfernungsbereich.

Zur Positionierung relativ zu Postkästen und Packstationen wurde eine Kamera (identisch zur Fisheye-Kamera des TULF) am Dach integriert, um mit dieser auf Basis von Markern und Kanten die Lage des Postkastens präziser als mit dem INS bestimmen zu können [Jaspers et al., 2016].

3.7.4 Rechnerarchitektur, Kommunikation

Die Rechnerarchitektur wurde entsprechend der am Institut bewährten Trennung in zwei Schichten gewählt.

Low-Level-Rechner Als Low-Level-Rechner wurde eine dSPACE MicroAutoBox II 1401/1513 integriert, die über zahlreiche CAN-Bus-Schnittstellen mit Basisfahrzeug und Aktorik kommuniziert und teilweise als CAN-Gateway fungiert. Die Kommunikation mit dem High-Level-Rechner erfolgt mit einer an MuCAR-4 angelehnten Schnittstelle.

High-Level-Rechner Für beide Fahrzeuge wurde ein zu MuCAR-3 (2015, siehe Tabelle 3.2) identischer Fahrzeugrechner spezifiziert, der aus dem 12 VDC-Messtechnik-Netz versorgt wird, und unterhalb des Beifahrersitzes integriert. Der zugehörige Entwickler-Arbeitsplatz befindet sich am Armaturenbrett.

Die Anbindung aller Sensoren und der weiteren Rechner erfolgt über mehrere 1 Gbit-Netze.

3.7.5 Sicherheitseinrichtungen

Der Sicherheitsfahrer ist auch in diesem Fahrzeug der Kern des Sicherheitskonzepts. Die Übernahme von der Autonomie in den manuellen Modus erfolgt hier mittels eines Schalters im Armaturenbrett oder über die Berührung des modifiziertes Bremspedals. Bei PARAVAN SpaceDrive II ist eine Übersteuerung des Lenkrads nicht einfach möglich, hier muss zuvor die Magnetkupplung gelöst werden. Dies erfolgt entweder durch den Sicherheitsfahrer über einen Schalter oder softwareseitig aus dem Low-Level-Rechner. Die Übersteuerung der Bremse ist immer möglich.

Ein Not-Stopp-Taster im Armaturenbrett und ein optionaler Funk-Not-Stopp lösen dagegen eine kontrollierte Notbremsung bis in den Stand aus, wobei die Querführung weiter aktiv bleibt.



(a) StreetScooter mit Sensorik

(b) Messtechnik im Compartment

Abbildung 3.16:

Versuchsfahrzeug MORE-Gütertransport, ein StreetScooter B16 Work, mit Sensorik: LiDAR Velodyne VLS-128 (gelb), LiDAR Ouster OS0-32 (gelb gestr.), LiDAR Velodyne VLP-16 (grün), Radar (rot), RGB-Kameras (magenta), GNSS-Antennen (blau), INS (blau gestr.) Der Low-Level-Rechner (orange gestr.) ist im Fahrzeughimmel integriert. Im Compartment befinden sich Energieversorgung (weiß), High-Level-Rechner (orange).

3.8 StreetScooter B16 – MORE Gütertransport

Für das Projekt MORE (Munich Mobility Research Campus), welches aus dem dtec.bw (Zentrum für Digitalisierungs- und Technologieforschung der Bundeswehr) gefördert wird, wurde das in Abb. 3.16 gezeigte Versuchsfahrzeug als Demonstrator für den automatisierten Gütertransport aufgebaut.

Als Basis dient hier ein StreetScooter B16 Work – die Nachfolgeneration zum B14 aus Abschnitt 3.7, welcher ursprünglich bei StS für eigene Entwicklungsprojekte im Bereich des autonomen Fahrens unter Verwendung der Erkenntnisse aus dem gemeinsamen Vorgängerprojekt aufgebaut wurde. Dieses Fahrzeug wurde teilumgerüstet von StS übernommen, um Zeit bei der Implementierung eines DbW-Systems zu sparen. Alle bereits vorhandenen Komponenten bis auf die Sensorik befinden sich innerhalb der Fahrerkabine (dort größtenteils im Himmel) oder im Motorraum.

In der institutseigenen Werkstatt wurde das Fahrzeug den Anforderungen an Sensorik und Rechentechnik im Projekt angepasst, die Erweiterungen befinden sich im als „Compartment“ bezeichneten Laderaum oder an der Außenhaut des Fahrzeugs.

3.8.1 Energieversorgung und Kühlung

Die Energieversorgung für die Messtechnik erfolgt aus dem 12VDC-Bordnetz und nicht aus der Hochvoltbatterie des BEVs – anders als bei den Vorgängerfahrzeugen. Im B16 ist ein Antriebsstrang eines anderen Zulieferers integriert, bei dem eine zusätzliche Energieentnahme aus der Hochvoltbatterie im Monitoring auffallen und zu Fehlermeldungen führen würde.

Der serienmäßige DC-DC-Konverter versorgt dabei das 12 VDC-Bordnetz, welches mit zwei weiteren Batterien verstärkt wurde. Zusätzlich kann dieses Netz mit Landstrom geladen werden. Über einen Sinus-Wechselrichter wird das 230 VAC-Messtechniknetz versorgt.

Erste Testfahrten bei sommerlichem Wetter zeigten den Bedarf nach mindestens einer Lüftung für die Messtechnik im Compartment auf, da das Fahrzeug über keinerlei Klimatisierung verfügt.

3.8.2 Aktorik

Alle Komponenten des DbW-Systems wurden bereits vom Fahrzeughersteller StS zusammen mit Partnerfirmen eingerüstet, parametrisiert und an den Low-Level-Rechner angebunden.

Antrieb Im Gegensatz zu den bisher genannten Fahrzeugen entfällt bei diesem BEV die Ansteuerung über das Gaspedal. Stattdessen kann über den CAN-Bus ein positives oder negatives Antriebsmoment für die E-Maschine vorgegeben werden, da diese Schnittstelle für diesen Umbau offengelegt wurde. Im normalen Fahrbetrieb reicht die über die E-Maschine erzielte Verzögerung durch Rekuperation aus. Im Fall einer vollständig geladenen Hochvoltbatterie kann allerdings keine Rekuperation erfolgen, sodass die klassische Bremsanlage benötigt wird.

Bremse In die Bremsanlage wurde ein Prototypen-Steuergerät eines Automobil-Zulieferers integriert, welches vom Low-Level-Rechner angesteuert werden kann, um einen Bremsdruck zu regeln. Die Bremse kommt immer dann zum Einsatz, wenn die Verzögerung durch Rekuperation nicht ausreicht.

Lenkung Die originale Servolenkung des StS besitzt keinerlei Möglichkeit zur Ansteuerung. Daher wurde eine elektromechanische Servolenkung eines Zulieferers integriert, die eigentlich für einen anderen Fahrzeugtyp vorgesehen war. Zur Adaption der Mechanik musste die Vorderachse um einige Zentimeter nach vorn verschoben werden – dies hatte eine veränderte Lenkgeometrie und erhöhten Reifenverschleiß zur Folge. Diese Lenkung lässt sich per CAN-Bus vom Low-Level-Rechner über eine Lenkwinkelschnittstelle ansteuern.

Sekundärfunktionen Ein über den CAN-Bus ansteuerbares Relaisboard dient zur Betätigung der Sekundärfunktionen, zu nennen sind hier Blinker, Licht und Hupe. Zusätzlich wird das Bremslicht angesteuert, wenn im autonomen Betrieb mittels Rekuperation verzögert wird.

3.8.3 Sensorik

3.8.3.1 Fahrdynamiksensorik

Hier stehen auf dem CAN-Bus der Lenkradwinkel aus der Lenkactorik sowie Antriebsmoment und Drehzahl der E-Maschine bereit. Die Kennlinie für die Übersetzung von Lenkradwinkel zu Lenkwinkel am Rad wurde von StS mitgeliefert. Die Geschwindigkeitssignale werden identisch zu den Vorgängerfahrzeugen aus CAN und digitalen Pulsen gewonnen.

3.8.3.2 Lokalisierungssensorik

INS Hier wurde bereits von StS ein OxTS RT3003v2 an der Rückwand der Fahrerkabine montiert. Die GNSS-Antennen waren ursprünglich im Himmel unter dem Kunststoffdach der Fahrerkabine montiert. Die teilweise Abschattung durch das höhere Compartment führte hier zu unnötiger Performancereduktion. Im Rahmen der Sensorik-Erweiterung wurden sie durch neue Antennen auf dem Dach des Compartments ersetzt.

3.8.3.3 Umfeldsensorik

Während bis hierhin viele Komponenten mehr oder weniger von StS übernommen werden konnten, wurde das Setup der Umfeldsensorik weitgehend verworfen [Kusenbach und Lüttel, 2021]. Stattdessen wurden die Erkenntnisse aus MuCAR-3 genutzt, um hier eine für die bestehenden Wahrnehmungsalgorithmen möglichst kompatible Ausstattung zu erzielen [Kusenbach, 2022].

Im Vergleich zu den Vorgängerfahrzeugen aus Abschnitt 3.7 ist die Herausforderung, die Sensoren zum Großteil innerhalb der Fahrzeugkontur zu platzieren und trotzdem tote Winkel möglichst zu vermeiden. Zudem soll auf „Hörner“ wie bei den vorderen Eck-LiDAR-Sensoren verzichtet werden, die v. a. beim Betrieb auf nicht-abgesperrtem Gelände eine Gefährdung umstehender Personen darstellen können.

Kameras Die Anordnung der Kameras auf dem Fahrzeugdach ähnelt dem am Institut entwickelten 360°-Kamerasystem von MuCAR-5, nur werden hier Einzelkomponenten statt ein gemeinsames Gehäuse verwendet. Das Kamerasystem besteht aus drei RGB-Industriekameras nach dem GigE Vision-Standard mit einer Auflösung von 5 Mpix, die mit hochwertigen Objektiven jeweils ein HFOV von ca. 80° abdecken.

Eine weitere Kamera wurde zunächst statisch hinter der Windschutzscheibe montiert. Mittelfristig soll auch hier eine bewegliche Kameraplattform zum Einsatz kommen.



Abbildung 3.17:

Kalibrierte 3D-Punktwolken aus dem Velodyne VLS-128 (rot), drei Ouster OS0-32 (weiß) und drei Velodyne VLP-16.

LiDAR Die Auswahl der LiDAR-Sensoren orientiert sich an MuCAR-3 und MuCAR-5. Der zentrale Fernbereichssensor vom Typ Velodyne VLS-128 befindet sich am Compartmentdach oberhalb der Fahrerkabine. In der hinteren Halbebene sind aufgrund von Selbstverdeckung nur die oberen Laser verwendbar, die vordere Halbebene wird nicht beeinträchtigt.

Drei um 45° gekippte Nahbereichssensoren vom Typ Ouster OS0-32 an der linken, rechten und hinteren Seite des Compartmentdachs dienen zur Überwachung der toten Winkel und zur Vervollständigung der 3D-Punktwolke bspw. für Objekttrackingverfahren.

Zusätzlich verfügt das Fahrzeug über mehrere LiDAR-Sensoren vom Typ Velodyne VLP-16 Puck [Velodyne LiDAR, Inc., 2018], die meisten wurden von StS übernommen. Der VFOV von $\pm 15^\circ$ resultiert in einem großen vertikalen Strahlabstand von 2° , daher ist dieser Sensortyp trotz der nominellen Reichweite vom 100 m eher zur Hinderniserkennung im Nahbereich als für das Objekttracking in der Entfernung geeignet. Die zwei Sensoren an den hinteren Fahrzeugecken sowie der neu in der vorderen Stoßstange integrierte Sensor dienen primär dem unterlagerten Nahfeldüberwachungssystem NeFiMMore (Near Field Monitoring System MORE) [Pan und Sotiriadis, 2022, Pan et al., 2023]. NeFiMMore verarbeitet die LiDAR-Daten auf separaten Mikrocontrollern, prüft die vom Autonomiesystem geplanten Trajektorien auf Hindernisfreiheit und kann ggf. einen Nothalt auslösen. Die Sensoren auf dem Dach der Fahrerkabine werden derzeit nicht verwendet.

Abbildung 3.17 zeigt die 3D-Punktwolken dieser Sensoren.

Radar Für dieses Fahrzeug wurde ein von MuCAR-3 inspiriertes Setup aus fünf Nahbereichs-Radar-Sensoren des Herstellers smart microwave sensors GmbH integriert. Aufgrund des auf den Uni-Campus beschränkten Einsatzbereichs kann überall von niedrigen Geschwindigkeiten ausgegangen, daher wurde auf einen Fernbereichs-Radar-Sensor verzichtet.

Tabelle 3.4: High-Level-Rechner im StreetScooter MORE-Gütertransport

Jahr	CPU-Typ	#CPUs x #Kerne	Frequenz (GHz)	RAM (GB)	GPU
2021	AMD Ryzen 7 Pro 4750U	1x 8 + HT	1,7–4,1	32	-
2023	AMD Ryzen 9 5900X	1x 12 + HT	3,7–4,8	64	RTX A5000

3.8.4 Rechnerarchitektur, Kommunikation

Low-Level-Rechner Hier kommt eine dSPACE MicroAutoBox II 1401/1513 zum Einsatz, die über zahlreiche CAN-Bus-Schnittstellen mit Basisfahrzeug und Aktorik kommuniziert und teilweise als CAN-Gateway fungiert. Die von StS mitgelieferte Software baut auf den Vorarbeiten des Instituts (vgl. Abschnitt 3.7) auf. Die Kommunikation mit dem High-Level-Rechner wurde modifiziert und erfolgt mit einer zu MuCAR-4 identischen Schnittstelle.

High-Level-Rechner Der aktuelle Fahrzeugrechner aus Tabelle 3.4 ist im Vergleich zu den anderen Fahrzeugen etwas schwächer ausgelegt, um die elektrische Leistungsaufnahme zu reduzieren. Nach Vergleichsmessungen fiel die Entscheidung für eine Versorgung über ein handelsübliches 230 VAC-ATX-Netzteil und gegen ein spezielles 12 VDC-ATX-Netzteil. Der kombinierte Wirkungsgrad von Wechselrichter und Standardnetzteil ist vergleichbar oder besser als beim 12 VDC-ATX-Netzteil, zudem gewinnt man an Flexibilität.

Die Anbindung aller Sensoren und der weiteren Rechner erfolgt über 10 Gbit-Ethernet. Für die Radar-Sensoren kommt ein Medienkonverter von Automotive Ethernet 100BASE-T1 auf 1000BASE-T zum Einsatz. Weitere Komponenten werden im Himmel über einen Ethernet-Switch aggregiert. Zur logischen Trennung der unterschiedlichen Komponenten kommen Virtuelle lokale Netzwerke, engl. *Virtual Local Area Networks* (VLANs) zum Einsatz.

3.8.5 Sicherheitseinrichtungen

Der Sicherheitsfahrer ist auch in diesem Fahrzeug der Kern des Sicherheitskonzepts. Die Übernahme von der Autonomie in den manuellen Modus erfolgt hier mittels eines Schalters im Himmel. Die Übersteuerung von Lenkrad und Bremse ist immer möglich.

Ein Not-Stopp-Taster im Armaturenbrett und ein optionaler Funk-Notaus lösen dagegen eine Notbremsung bis in den Stand aus.

Zusätzlich wird das oben erwähnte Nahfeldüberwachungssystem NeFiMMore integriert, das mit separater Sensorik und Embedded-Rechnern den Bereich vor dem Fahrzeug überwacht und eine Notbremsung auslöst, wenn Fehlfunktionen im Hauptsystem zu übersehenen Hindernissen führen.

3.9 Fazit

Gesamtfahrzeug Der Eigenbau von Versuchsfahrzeugen auf Basis von Serienfahrzeugen – wie bei MuCAR-3 und MuCAR-4 – bietet die höchste Flexibilität, kostet aber viel Zeit und Aufwand auf Institutsseite und erfordert Erfahrung und Know-How. Dafür erlangt man das höchste Systemverständnis, was bei der Abschätzung der Leistungsfähigkeit und bei eventuellen späteren Fehlersuchen immer hilfreich ist.

Ist man von Beginn an in den Aufbau von Fremdfahrzeugen involviert und begleitet ihn hochfrequent, siehe die StS-Fahrzeuge *Adam* und *Eva*, so besteht ein ähnlich hohes Systemverständnis, welches den Betrieb erleichtert.

Fremdfahrzeuge wie TULF und StrAsRob, die zum Großteil durch die Industrie aufgebaut bzw. umgerüstet wurden, sind aufgrund der Anzahl der involvierten Partner wesentlich schwieriger zu warten. Oftmals werden diese von Personal aufgebaut, welches selbst nicht besonders tief in die Themen des autonomen Fahrens und die Anforderungen an Sensorik und DbW-System eingearbeitet ist, sodass viele Themen Iterationszyklen benötigen – z. B. die Auflösung und Frequenz der Geschwindigkeitsmessungen in den LKWs oder die Kalibrierung der Lenk-Kennlinie. Im Nachgang lassen sich auch bei Rechnern, Sensorik oder Aktorik Änderungen herbeiführen, aber nur unter hohem Aufwand.

Zwischen diesen beiden Gruppen ist der MORE-Gütertransport aus Abschnitt 3.8 einzuordnen. Hier wurden durch den Hersteller StS und Zulieferer viele Modifikationen vorgenommen, die zum Teil noch einige Nacharbeit auf Uni-Seite erforderten. Zudem konnte die vorhandene Sensorik nicht wie erhofft zum Einsatz kommen. Für das Low-Level-System wurde der Quellcode mitgeliefert, sodass dieses leicht an unsere Anforderungen angepasst werden konnte.

Schlussendlich hat sich die gewählte Aktorik und Sensorik bei allen Fahrzeugen als geeignet herausgestellt. Für MuCAR-5 kann aufgrund von fehlender Schnittstellendokumentation und ausstehenden Inbetriebnahmen noch keine Aussage getroffen werden.

Sensorik Retrofit-Sensorik bietet starke Vorteile, sie ist meist moderner und leistungsfähiger als Seriensensorik, allerdings auch teurer. Seriensensorik verfügt meistens nur über geschlossene bzw. geheime Schnittstellen, welche im Regelfall weder vom OEM noch vom Zulieferer für ein Uni-Institut offengelegt werden.

Eine flexible Nachrüstung neuer Sensoren oder Rechner ist bei den Fahrzeugen je nach Forschungsauftrag bzw. Studie möglich gewesen, damit sind die Fahrzeuge auch langfristig nutzbar. MuCAR-3 ist bereits seit 2006 im Dienst und immer noch das Haupt-Versuchsfahrzeug des Instituts.

Insgesamt lässt sich feststellen, dass das KISS-Prinzip (Halte es einfach, Du Dummkopf!, engl. *Keep it simple, stupid!*) auch hier seine Berechtigung hat: ein nachgerüstetes Fahrzeug wie MuCAR-3 mit separater Aktorik und an der Außenhaut

nachgerüsteten Sensoren lässt sich im Forschungsumfeld leichter warten und erweitern als ein komplexes monolithisches System wie ein moderner PKW mit ab Werk integrierter Sensorik und Aktorik – hier ist als Beispiel leider noch MuCAR-5 zu nennen.

Wichtig sind einheitliche Schnittstellen zwischen High-Level-Rechner und Low-Level-Rechner. Diese sollten mindestens auf Middlewareebene gleich gehalten werden, auch wenn je nach verwendetem Bus-System unterschiedliche Module zur Kommunikation mit dem Low-Level-Rechner zum Einsatz kommen: MuCAR-3 über CAN; MuCAR-4, MuCAR-5 und der StreetScooter über UDP; TULF und StrAsRob über ROS-Konverter.

Damit kann die Forschungsfrage 1 „Wie kann ein Versuchsfahrzeug mit seiner Sensorik-, Aktorik- und Rechner-Ausstattung gestaltet werden?“ als beantwortet gelten: Beim Aufbau der Fahrzeugplattformen wurden unterschiedliche Wege verfolgt, die alle zum Ziel führten – aber das KISS-Prinzip ist für die Nutzung an einem kleinen Uni-Institut aufgrund der reduzierten Abhängigkeiten aus Sicht des Autors am besten geeignet.

Sicherheitskonzept Im Rahmen der Fahrzeugbetriebs hat sich immer wieder bestätigt, dass der Betrieb von autonomen Versuchsfahrzeugen im universitären Umfeld nur mit einem Sicherheitsfahrer an Bord guten Gewissens durchführbar ist, da dieser zusammen mit dem Operator Software und Umfeld überwacht und jederzeit die manuelle Kontrolle über das Fahrzeug übernehmen kann, bevor Schaden entsteht. Hierzu ist zwingend eine gute Visualisierung des Fahrzeugzustands und der wahrgenommenen Umgebung erforderlich, da diese dem Menschen bei der Beurteilung des Systemzustands und der Vorhersage von „Unterstützungsbedarf“ hilft. Der Sicherheitsfahrer ist ebenso für den möglichen Ausfall von Aktorik oder Sensorik eine geeignete Rückfallebene. Not-Stopp-Taster wurden dagegen, auch wenn sie im Fahrzeug vorhanden waren, selten genutzt.

Ein vollkommen unbemannter Betrieb im Straßenverkehr ist mit einem kleinen Team nicht darstellbar, da der Fokus bei der Forschung und Entwicklung nicht auf unbedingter Stabilität, Robustheit und Zulassungsfähigkeit liegt.

Damit kann aus Sicht des Autors die Forschungsfrage 2 „Wie sieht ein für die Entwicklungsphase geeignetes Sicherheitskonzept aus?“ ebenso als beantwortet gelten: Mit einem Sicherheitsfahrer, der jederzeit die volle Kontrolle über das System übernehmen kann, kann ein autonomes Fahrzeug in der Entwicklungsphase im universitären Umfeld betrieben werden.

Lessons Learned An dieser Stelle werden noch einmal Themen zusammengefasst, die bei mehreren Fahrzeugen aufgetreten sind und die beim Entwurf solcher Fahrzeuge berücksichtigt werden sollten.

Eine Energieversorgung kann nie stark genug ausgelegt sein, da im Laufe der Betriebsjahre durch Umbauten weitere und meist leistungshungrigere Komponenten

hinzukommen. Hier ist auf ausreichende Querschnitte in allen Kabeln zu achten, um unnötige Spannungsabfälle und damit Verschwendung der knappen Energie zu vermeiden. Auch jede Energieverteilung ist mit Reserven zu spezifizieren.

Dieselbe Taktik empfiehlt sich für Datenleitungen: es sollte nicht nur die gerade benötigte Anzahl an Leitungen, z. B. Ethernetbuchsen im Fahrzeugdach, verlegt werden, sondern immer mindestens Faktor zwei, um nachträglichen Aufwand zu sparen. Werden hochwertige Leitungen und Stecker verwendet, ist auch eine nachträgliche Geschwindigkeitssteigerung auf 10 Gbit/s möglich.

Aus den Testfahrten am TULF folgt, dass vor einer finalen Integration von Sensoren prüfen sollten, ob die laut Datenblatt möglichen Einbaulagen tatsächlich funktionieren. Beim INS XSens MTi-G kann zwar eine Einbaulage im Sensor konfiguriert werden, saubere Ergebnisse liefert allerdings nur eine einzige. Anzuraten ist auch, zuerst alle zur Verfügung stehenden CAN-Signale zu betrachten und sich danach zu entscheiden, und nicht wie beim TULF für die erstbeste, aber niedrig aufgelöste Geschwindigkeitsnachricht.

Nicht unerwartet ist, dass Versuchsfahrzeuge genauso wie privat genutzte Fahrzeuge altern – allerdings im Zeitraffer, da die Belastung aller Komponenten, von Reifen über Fahrwerk bis zur Energieversorgung, im Versuchsbetrieb und vor allem im Offroad-Einsatz wesentlich höher ist. Darauf sind auch SUVs nicht ausgelegt, auch wenn die Werbung anderes verspricht – eine alte Mercedes-Benz G-Klasse ist hier sicher um Größenordnungen robuster, aber als Versuchsfahrzeug aufgrund kaum vorhandener bordeigener Sensorik für die hier verfolgten Zwecke weniger geeignet. Zusätzlich altern aber auch alle eigenen Modifikationen, sodass beim langjährigen Betrieb neue Fehler auftreten, z. B. aufgrund gealterter Elektronikbauteile.

Immer wieder ein Grund für Probleme sind elektromagnetische Störungen, entweder durch schlecht geschirmte Kabel oder aber durch abstrahlende Sensorik. So ist bei der Kombination aus Ouster-LiDAR-Sensoren und GNSS-Antennen auf ausreichenden Abstand zu achten, da ansonsten eine Spitze im abgestrahlten Spektrum genau das L1-Band des GPS auslöscht.

4 Softwarearchitektur und Synchronisierung

In diesem Kapitel werden einige grundlegende Themen erläutert, die nicht nur für die weiteren Kapitel dieser Arbeit von Bedeutung sind, sondern allgemein für alle Nutzer von autonomen Versuchsfahrzeugen.

Abschnitt 4.1 betrachtet die im Rahmen der Arbeit verwendete Softwarearchitektur und Middleware zur Inter-Prozess-Kommunikation. Abschnitt 4.2 beschäftigt sich mit der Zeitstempelung von Messdaten und ihrer Synchronisation. Abschnitt 4.3 gibt einen kurzen Hinweis zur Kalibrierung.

4.1 Softwarearchitektur und Middleware

Bei der Entwicklung und dem Betrieb eines autonomen Versuchsfahrzeugs sind eine Vielzahl an Entwicklern und Wissenschaftlern beteiligt, jeder ist Spezialist in seinem Bereich. Um zielführend Software und Algorithmen für gemeinsam genutzte Versuchsfahrzeuge entwickeln und testen zu können, ist eine geeignete Softwarearchitektur zu wählen.

4.1.1 Anforderungen an die Softwarearchitektur

Für den hier betrachteten Anwendungsfall bestehen folgende Anforderungen:

- Test- und Vergleichbarkeit von neu entwickelten oder verbesserten Algorithmen
- Robustheit im Betrieb mit prototypischer Software
- Wiederverwertbarkeit von Software auf Fahrzeugplattformen unterschiedlicher Größe
- Nutzung sowohl im Versuchsfahrzeug als auch auf Entwickler-PCs im Büro und Labor
- Möglichkeit zur (selektiven) Aufzeichnung aller Sensordaten sowie Ausgaben der Einzelmodule während des Live-Betriebs; Wiedergabemöglichkeit zur Algorithmenentwicklung und zum Debugging im Büro bzw. Labor
- Echtzeitfähigkeit

Eine monolithische Software, also ein Anwendungsprogramm, welches alle Algorithmen aller Entwickler enthält, ist kaum langfristig wartbar. Stattdessen wird ein modulares System benötigt, in dem jedes Modul eine klar abgegrenzte Funktion erfüllt und über definierte Schnittstellen mit anderen Modulen kommuniziert.

Für die Kommunikation zwischen den Modulen, die sog. Interprozesskommunikation, engl. *Inter-Process Communication* (IPC), wird eine geeignete *Middleware* benötigt. Der Begriff *Middleware* ist in der Informatik breit besetzt:

Der Begriff Middleware bezeichnet Software, mit deren Hilfe sich die Lücken zwischen anderen Anwendungen, Tools und Datenbanken schließen lassen, sodass Anwender von nahtlosen Services profitieren können. Oft wird Middleware als eine Art Kleber bezeichnet, der die verschiedenen Softwareplattformen und Geräte miteinander verbindet. [...] Robotische Middleware: Robotische Middleware wird eingesetzt, um komplexe Abläufe beim Bau von Robotern zu vereinfachen, wie z. B. bei der Entwicklung der Steuerung oder bei Simulationen. (Talend Germany GmbH [2022])

Eine Middleware ermöglicht den Datenaustausch über definierte Schnittstellen, einige werden oft vom Hersteller der Middleware mitgeliefert, andere können selbst definiert werden, z. B. Kamerabilder inklusive der Kalibrierungsparameter, die Selbstlokalisierung des Fahrzeugs oder Objektlisten aus einem Trackingverfahren. Dabei ist grundsätzlich Wert auf eine langzeitstabile Schnittstelle zu legen, um unnötige Anpassungen an den zahlreichen Modulen zu vermeiden. Einheitliche Schnittstellen ermöglichen zudem den einfachen Austausch von Modulen bei einer Neuentwicklung, oder die einfache Einbindung eines weiteren Moduls, welches bspw. Kamerabilder benötigt.

Prototypische Software im universitären Umfeld ist im Regelfall nicht frei von Fehlern und kann abstürzen. Durch die lose Kopplung der Module über die Middleware verursacht ein Absturz von Komponente A keinen Komplettabsturz der Software, wie es bei einem monolithischen System wäre, sondern sorgt nur für den Ausfall eines Teilsystems. Werden die Module während des Betriebs über eine Software zentral gestartet und überwacht, kann diese Zentrale abgestürzte Module neu starten und damit die Auswirkungen des Fehlers minimieren.

Wenn die entwickelte Software auf Fahrzeugen unterschiedlicher Größenklassen zum Einsatz kommen soll, muss diese skalierbar sein. Mit einer monolithischen Software wäre dies schwer zu erzielen, der modulare Ansatz bietet auch hier Vorteile: bei einer kleinen Plattform mit wenig Sensoren und geringer Rechenleistung werden meist nicht alle Funktionalitäten benötigt, sodass nur einzelne Module gestartet werden müssen. Ebenso ist die Diagnose einfacher, welches Modul welche Rechenleistung benötigt, sodass zielgerichtet optimiert werden kann.

Sollen im Büro und Labor zur Weiterentwicklung der eigenen Algorithmen und zur Auswertung von Messdaten dieselben Softwaremodule wie auf dem Versuchsfahrzeug genutzt werden können, so wird eine Funktionalität zur zeitsynchronen Datenaufzeichnung und -wiedergabe benötigt, möglichst mit einer Filtermöglichkeit während Aufnahme und Wiedergabe. Im Laborbetrieb entfallen daher die Sensortreiber oder sog. Dateneinzüge, stattdessen dient ein sog. Player zur Wiedergabe der mit dem Fahrzeug aufgenommenen Daten. Alle weiteren Module insbesondere aus dem Bereich der maschinellen Wahrnehmung können problemlos auf jedem PC genutzt werden – dieses jedoch nur Open-Loop ohne Rückkopplung in die Steuerung.

Für Closed-Loop-Tests, also Untersuchungen im geschlossenen Regelkreis, kommt entweder das reale Versuchsfahrzeug oder vorab eine Simulation zum Einsatz. Dies ist besonders für die erste Reglerauslegung oder Verhaltensentscheidungen essentiell.

Sobald das reale Versuchsfahrzeug im Closed-Loop-Betrieb genutzt wird, ist die Echtzeitfähigkeit des Gesamtsystems von extremer Bedeutung. Unter Echtzeitfähigkeit wird an dieser Stelle nicht die „harte“ Echtzeitfähigkeit mit garantierten Reaktionszeiten verstanden, sondern eine sog. „Schnell-genug-Echtzeit“:

- Die Sensoreinzüge sind in der Lage, jede mit der Frequenz f_{Sensor} von externen Schnittstellen eintreffende Messung zu verarbeiten und für „konsumierende“ Module bereitzustellen, bspw. Kamerabilder.
- Die konsumierenden Module können von eintreffenden Daten getriggert werden, um ohne Zeitverzug (bspw. durch Polling) mit der Verarbeitung starten zu können, und verarbeiten die Daten innerhalb eines Abtastschrittes, sodass sie ihre Ausgangsdaten mit $f_{\text{Modul}} = f_{\text{Sensor}}$ bereitstellen.
- Für alle kommunizierten Daten werden in den Metadaten der Aufnahmezeitstempel bzw. bei weiterverarbeiteten Daten der Bezugszeitstempel mitgeliefert, damit ist das Alter der Daten für die nachverarbeitenden Module bekannt und kann entsprechend kompensiert werden.
- Die verwendete Middleware sollte möglichst wenig Latenz im Datentransport und wenig Rechenlast verursachen.

Im Folgenden werden zwei Middleware-Varianten vorgestellt, die diese Anforderungen erfüllen: die KogMo-RTDB und ROS.

4.1.2 KogMo-RTDB

Die KogMo-RTDB (Echtzeitdatenbank für kognitive Automobile, engl. *KogniMobil Real Time Data Base*) wurde im Rahmen des DFG-finanzierten Sonderforschungsbereichs KogniMobil (Sonderforschungsbereich Transregio 28 – „Kognitive Automobile“) an der TU München entwickelt [Goebel und Färber, 2007]. Über viele Jahre hat sie sich zur IPC in den autonomen Fahrzeugen diverser Projektpartner in München und Karlsruhe sowie zum Datenaustausch zwischen Projektpartnern bewährt.

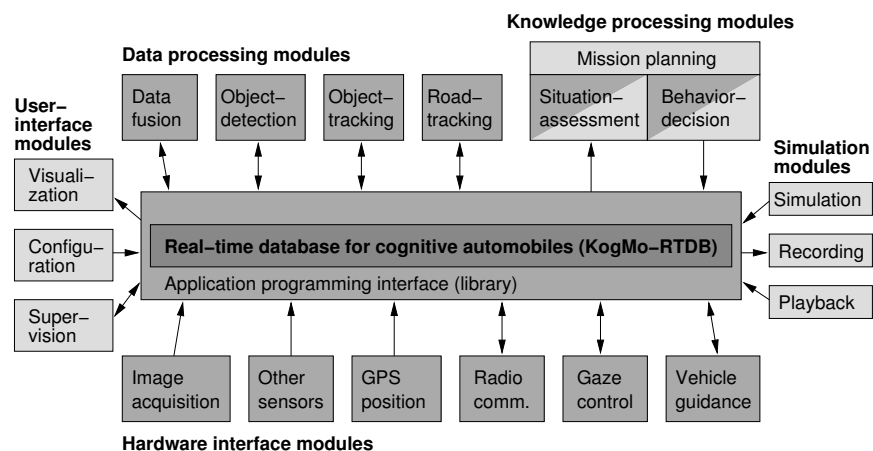


Abbildung 4.1: Architektur der KogMo-RTDB. Bildquelle: [Goebel und Färber, 2007].

Die KogMo-RTDB basiert auf einer Shared-Memory-Architektur, wodurch sie derzeit auf einen Rechner limitiert ist. Diese Technik erlaubt aber einen sehr hohen lokalen Datendurchsatz, sodass alle Datentypen über eine einheitliche Middleware-Schnittstelle ausgetauscht werden können. Für jede Nachricht steht ein Ringpuffer mit einstellbarer Größe bereit, sodass über die API (Application Programming Interface) mithilfe eines Zeitstempels auf historische Daten zugegriffen werden kann.

Die zeitsynchronisierte Speicherung aller Rohdaten – inkl. der Kamerabilder – sowie zwischenverarbeiteter Daten von den einzelnen Applikationen wie z. B. Missionsplanung, Objekterkennung oder Eigenbewegungsschätzung, kann im laufenden Betrieb in sog. RTDB-Streams erfolgen. Die maximale Datenrate bei der Aufzeichnung ist primär durch die Schreibgeschwindigkeit des Datenträgers begrenzt und seit der Umstellung von Festplatten auf SSDs kein Problem mehr. In den RTDB-Streams steht das gesamte Informationsmaterial, welches zur Analyse und zum Offline-Testen von Modulen im Labor notwendig ist, zur Verfügung.

Aufgrund der positiven Erfahrungen aus dem SFB KogniMobil wurde die KogMo-RTDB am Institut weiter betrieben und hat sich zum langjährigen Kern des Softwareframeworks entwickelt und sich über Jahre in Benchmarks ggü. dem später entstandenen ROS (Robot Operating System) behauptet [Schneider und Wünsche, 2013].

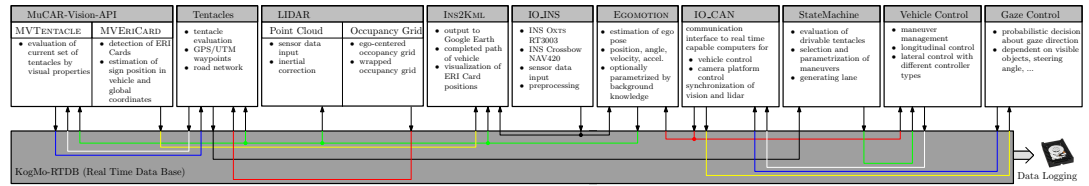
Rund um die KogMo-RTDB sind im Laufe der Jahre viele Tools entstanden, bspw. zum Datenexport oder zur Datenanalyse. Abbildung 4.1 zeigt die generelle Architektur der KogMo-RTDB, in Abbildung 4.2 werden drei Beispiele für um die KogMo-RTDB herum entwickelte Softwarearchitekturen präsentiert.

4.1.3 ROS

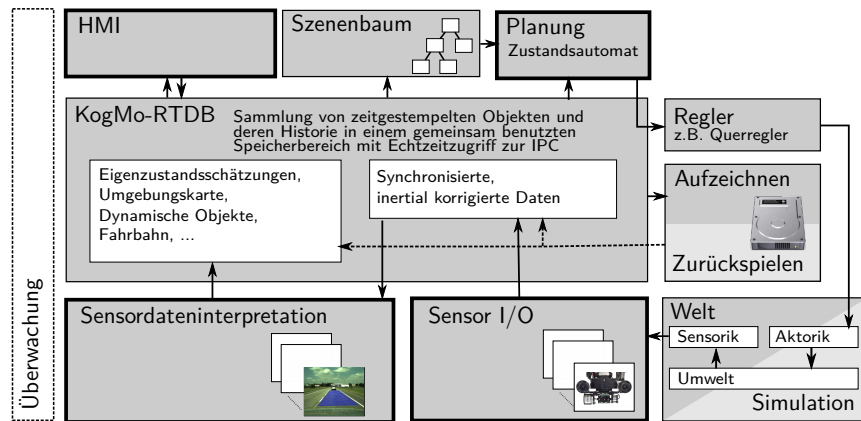
Das Robot Operating System (ROS) geht auf Arbeiten am Stanford Artificial Intelligence Laboratory zurück und wurde seit 2007 als Open-Source-Projekt durch die Firma Willow Garage als Framework für persönliche Roboter und Industrieroboter vorangetrieben [Ackerman und Guizzo, 2017]. Seit 2012 wird ROS von der gemeinnützigen Open Source Robotics Foundation (OSRF) unterstützt und weiterentwickelt.

ROS findet jedoch auch Anwendung in der Forschung und Vorentwicklung bei Automobilherstellern wie BMW [Gerkey, 2015] und in vielen Universitäten. Die Firma Apex.AI entwickelt auf Basis von ROS einen kommerziellen Ableger, der die Stärken von ROS übernimmt, und fokussiert sich auf Verlässlichkeit, Stabilität und Sicherheit für Anwendungen im autonomen Fahren [Ackerman, 2019].

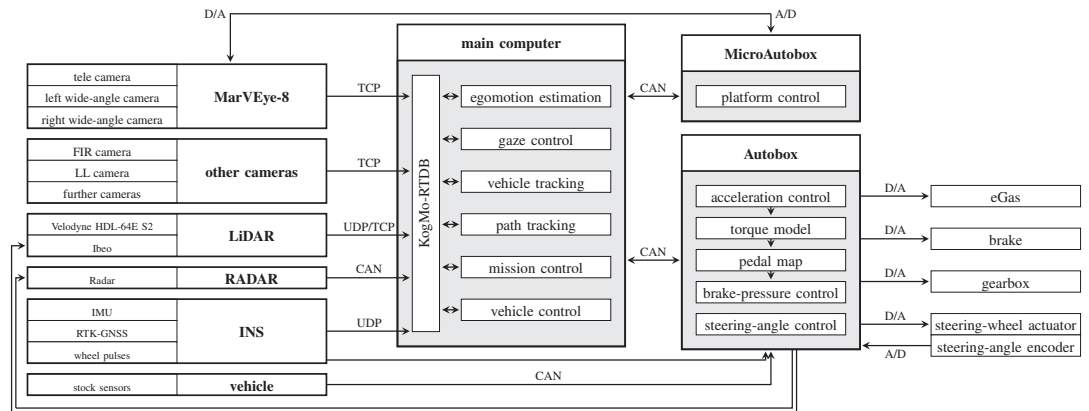
Der zentrale *ROS-Core* dient zur „Vermittlung“ der Kommunikation zwischen verschiedenen Programmen, den sog. *Nodes*, der Datentransfer selbst läuft von Punkt zu Punkt. ROS ist im Gegensatz zur KogMo-RTDB Multi-Rechner-fähig, da der Datenaustausch über TCP (Transmission Control Protocol) oder UDP (User Datagram Protocol) abgewickelt wird. Dies sorgt bei im Netzwerk verteilten Nodes und großen Messages, bspw. bei Kamerabildern, für eine hohe Auslastung und Latenzen.



(a) Softwarearchitektur des Systems für die C-ELROB 2009. Bildquelle [Himmelsbach et al., 2009b].



(b) Softwarearchitektur 2012. Bildquelle: [Manz, 2013].



(c) Softwarearchitektur für die M-ELROB 2018. Die KogMo-RTDB ist trotz zahlreicher weiterer Sensoren und wachsendem Datendurchsatz weiterhin problemlos geeignet. Bildquelle: [Ebert et al., 2018]

Abbildung 4.2:

Weiterentwicklung der Softwarearchitektur von MuCAR-3 mit der Middleware KogMo-RTDB.

Neben der reinen Kommunikation bringt ROS eine Reihe an Tools zur Visualisierung von 2D-, 3D- und Zeitreihendaten mit, die leicht erweiterbar sind. Die Schnittstellen sind durch eine Vielzahl an mitgelieferten Nachrichten, die sog. *Messages*, gut spezifiziert und flexibel erweiterbar. Die starke Verbreitung von ROS hat zu einem starken Ökosystem geführt, viele Sensorhersteller liefern bereits ROS-Treiber mit, und auch zu vielen wissenschaftlichen Veröffentlichungen wird eine Referenzimplementierung für ROS bereitgestellt.

ROS-Bags dienen zur Datenaufzeichnung im Livebetrieb, ähnlich wie die RTDB-Streams enthalten sie eine zeitsynchronisierte Speicherung einer ausgewählten Menge an Nachrichten. Allerdings erzeugt die Aufnahme der ROS-Bags in allen Daten bereitstellenden Nodes zusätzliche Rechenlast, da die Daten ein weiteres Mal verschickt werden müssen, sodass negative Rückwirkungen durch die Aufzeichnung nicht auszuschließen sind. Die Nutzung der ROS-Bags ist flexibler als bei RTDB-Streams, es stehen viele Tools auch zur nachträglichen Bearbeitung und der Integration von Daten aus dem Postprocessing zur Verfügung.

In den letzten Jahren wurde die Migration auf ROS forciert, da einige Drittmittelpartner auf einer nativen Implementierung für das ROS-Framework bestehen und die Kopplung mit Softwaremodulen von Projektpartnern vereinfacht wird.

Derzeit laufen Untersuchungen zur Portierung auf ROS 2, das langfristig mit Updates versorgt werden soll und Performanceverbesserungen in der Inter-Prozess-Kommunikation verspricht. ROS 2 ist allerdings nicht direkt kompatibel mit dem am Institut verwendeten ROS 1 und vielen aus der Robotik-Community genutzten Softwaremodulen, daher wird die Umstellung noch herausgezögert.

4.1.4 Hybrider Betrieb mit mehreren Middlewares

In den Studien TULF (Technologieträger Unbemanntes Landfahrzeug) und StrAsRob (Straßentransport mit Assistenzfunktionen von Robotern) sowie den gleichnamigen Versuchsfahrzeugen kommt ROS als zentrale Middleware zum Einsatz. Über diese Middleware wird der Großteil des Datenaustausches zwischen Sensoreinzügen, Schnittstellenmodulen und den Algorithmen abgewickelt. Sämtliche Kamerabilder sowie andere größere Nachrichten sind bisher von der Verteilung über ROS ausgeschlossen, um die Netzwerk-Last im Gigabit-Ethernet-Netz zu reduzieren. Kamerabilder werden stattdessen über Multicast-Transfer direkt vom Sensor über ein separates Netzwerk an unterschiedliche Rechner verteilt. Des Weiteren kommen proprietäre UDP-Unicast-Verbindungen zum Einsatz.

Es ist anzumerken, dass die in diesen Studien verwendeten Nachrichten nicht den Standard-ROS-Nachrichtendefinitionen entsprechen, sondern proprietär sind. Grund hierfür ist, dass die Nutzdaten der Nachrichten an DBD-interne (Diehl BGT Defence GmbH & Co. KG) Datenstrukturen aus einer früher verwendeten Middleware angelehnt sind. Dies führt dazu, dass kaum Tools aus dem ROS-Ökosystem genutzt werden können.

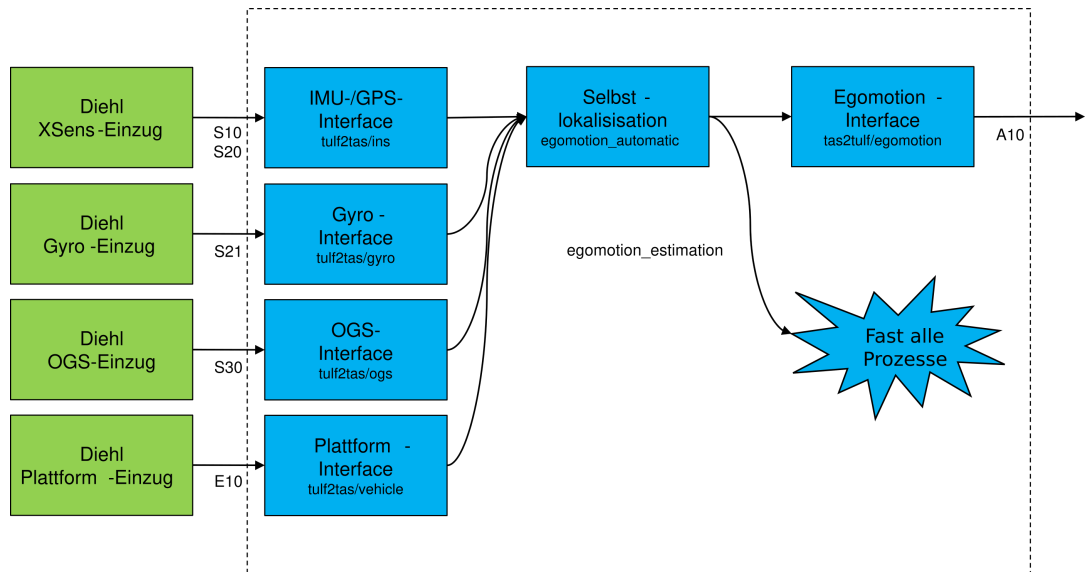


Abbildung 4.3:

Beispiel für die hybride Nutzung der beiden Middlewares ROS und KogMo-RTDB anhand der zur Eigenbewegungsschätzung und Selbstlokalisierung verwendeten Softwaremodule, der verwendeten Low-Level-Daten sowie der kommunizierten Daten. Alle Module im gestrichelten Kasten kommunizieren über die KogMo-RTDB. Abkürzungen an den Pfeilen geben einen Hinweis auf die verwendeten ROS-Messages, die vom Interface-Module *tulf2tas* (jetzt *ros2tas*) gelesen bzw. von *tas2tulf* (jetzt *tas2ros*) gesendet werden. In diesem Beispiel handelt es sich größtenteils um proprietäre, von DBD spezifizierte Nachrichten. Die mit *S*- und *E*- bezeichneten ROS-Messages enthalten Sensordaten (*S10* GNSS, *S20* 3-Achs-IMU, *S21* 1-Achs-Kreisel, *S30* Übergrund-Geschwindigkeitssensor, *E10* Fahrzeugplattform und DbW), mit *A*- bezeichnete Nachrichten enthalten Ergebnisse von Algorithmen (*A10* Eigenbewegung und Lokalisierung). Der Urheber der Softwaremodule ist jeweils farblich kodiert: TAS (blau), DBD (grün). Bildquelle: [Lüttel et al., 2015]

Um die in anderen BAAINBw-finanzierten (Bundesamt für Ausrüstung, Informationstechnik und Nutzung der Bundeswehr) Studien bei TAS entwickelten Softwaremodule möglichst schnell auf die Lastkraftwagen (LKWs) zu portieren, wurde auch aufgrund der Performanceuntersuchungen in [Schneider und Wünsche, 2013] auf eine native ROS-Portierung verzichtet. Stattdessen kommen modular aufgebaute Schnittstellenmodule *tas2ros* und *ros2tas* zwischen den beiden Middlewares KogMo-RTDB und ROS zum Einsatz, die den Betrieb der vorhandenen Softwaremodule ermöglichen. Ein Beispiel wird in Abb. 4.3 gezeigt. Dieser hybride Betrieb hat sich über mehrere Jahre bewährt.

4.2 Zeitstempel

Ein Zeitstempel ist ein Datum, im Linux-Umfeld üblicherweise die verstrichene Zeit seit dem 01.01.1970 in Nanosekunden-Auflösung, welches die Gültigkeit der Daten beschreibt, z. B. den Zeitpunkt der Messdatenaufnahme. Dieser Zeitstempel wird in der Kette der Algorithmen entsprechend weitergereicht, sodass z. B. eine erkannte

Objektposition im Kamerabild mit dem Zeitstempel des Bildes gekennzeichnet wird, und auch die durch einen anderen Algorithmus geschätzte Position des Führungsfahrzeugs diesen Zeitstempel trägt. Je weiter man in der Prozesskette voranschreitet und die Algorithmen dabei Rechenzeit „verbrauchen“, desto größer wird die Abweichung von Datenzeitstempel und aktueller Zeit – und desto größer wird die Bedeutung einer korrekten Handhabung.

Die im vorherigen Abschnitt 4.1 beschriebenen Middlewares KogMo-RTDB und ROS können die Zeitstempel in den Metadaten übermitteln und stellen geeignete Methoden in der API bereit, um auf die zu einem exakt bestimmten, dem nächstgelegenen, dem nächstjüngeren oder dem nächstälteren Zeitstempel korrespondierenden Daten zuzugreifen. Diese Methoden ermöglichen es den Algorithmen, synchronisierte Eingangsdaten unterschiedlicher Sensoren zu empfangen.

4.2.1 Rechner-Uhren

Allgemein gilt, dass bei Verwendung mehrerer Rechner im Netzwerk (im ROS-Umfeld oft üblich), die Daten untereinander austauschen, deren Uhren synchronisiert werden sollten, um sowohl den Uhrenfehler untereinander als auch die Abweichung zur globalen Zeit zu minimieren. Hierzu kommt das weit verbreitete Network Time Protocol (NTP) zum Einsatz, mit dem sich in einer vermaschten Struktur unterschiedliche Clients zu Zeitservern synchronisieren. In einem lokalen Netzwerk erreicht NTP bestenfalls eine Uhrengenauigkeit von 200 μ s, über das Internet etwa 10 ms.

Eine Alternative ist das modernere Precision Time Protocol (PTP), welches bei geeigneter Hardware die Uhren auf wenige Nanosekunden genau synchronisieren kann und welches von einer steigenden Anzahl an Sensoren direkt unterstützt wird. Zu beachten ist, dass es mehrere inkompatible Implementierungen von PTP gibt, sodass nicht jeder Sensor mit jedem Master kompatibel ist.

Für einen Alleinbetrieb ist eine globale Uhrensynchronisation nicht zwingend erforderlich, hier reicht eine fahrzeuginterne Synchronisierung aller Rechner und Sensoren. Wenn das Versuchsfahrzeug live Daten mit anderen Versuchsfahrzeugen oder stationärer Infrastruktur austauscht oder solche Daten im Postprocessing offline kombiniert werden, ist eine präzise Absolutzeit essentiell.

Die Versuchsfahrzeuge verfügen meist über keine stabile und latenzarme Internetverbindung, um sich zu einer globalen Zeitquelle zu synchronisieren. Stattdessen kann ein High-Level-Rechner als lokaler Zeitserver agieren und seine Zeit bspw. mithilfe eines GNSS-Empfängers (Globales Navigationssatellitensystem, engl. *Global Navigation Satellite System*) zur globalen Zeit synchronisieren.

Die Uhren der unterschiedlichen Rechner laufen aufgrund von Ungenauigkeiten und Temperaturabhängigkeiten der integrierten Quarze nach einer gewissen Zeit auseinander. Im laufendem Betrieb wird die Uhrzeit des Fahrzeugrechners wie bei jedem PC vom Betriebssystem anhand des CPU-Taktes ((Haupt-)Prozessor eines Computers, engl. *Central processing unit*) bestimmt, welches den Fehler klein hält. Bei abgeschaltetem Rechner übernimmt eine auf dem Mainboard integrierte und über eine

Pufferbatterie versorgte Echtzeituhr das Hochzählen der Zeit, was bei über Wochen geparkten und komplett spannungsfreien Versuchsfahrzeugen zu Uhrenfehlern in der Größenordnung von Minuten führen kann [Büchler, 2020].

Sowohl der initiale Uhrenfehler beim Einschalten des PC als auch kleine Differenzen im Betrieb können mittels der Synchronisierung zu einer externen Zeitquelle wie einem GNSS-Empfänger kompensiert werden. Allgemein kann jedoch nicht von einer permanent global synchronisierten Rechneruhr ausgegangen werden, da bspw. in langen Tunneln oder in Tiefgaragen die Synchronisierung mittels GNSS nicht möglich ist. Hilfreich ist allerdings, dass die Uhrenstabilität der GNSS-Empfänger selbst bei vollständiger Abschattung oft wesentlich besser ist als die der Rechneruhr und demnach zur Stützung beiträgt.

4.2.2 Zeitstempelvergabe

Sensoren mit synchronisierter Uhr

Einige Sensoren bieten Möglichkeiten zur Synchronisierung zur globalen Zeit und liefern passende Zeitstempel zu jeder Messung:

Velodyne LiDAR-Sensoren Aktuelle LiDAR-Sensoren (Light Detection And Ranging) des Herstellers Velodyne verfügen über eine interne Uhr, die sich mithilfe eines GNSS-Empfängers auf die globale Zeit synchronisieren lässt. Als Zeitquelle wird das PPS-Rechtecksignal (Puls pro Sekunde) zusammen mit der auf einer seriellen Schnittstelle im NMEA-Protokoll (National Marine Electronics Association) übertragenen Uhrzeit verwendet. Einige moderne LiDAR-Sensoren können sich alternativ auf einen PTP-Master synchronisieren, wodurch die separate Verkabelung für die Synchronisierung entfallen kann.

Auf Basis der synchronisierten internen Uhr werden alle einzelnen Entfernungsmessungen bereits im Sensor mit einem Zeitstempel versehen, der in der weiteren Verarbeitung berücksichtigt werden kann.

Als Nebeneffekt kann nur bei synchronisierten Sensoren eine synchrone Drehzahl über mehrere Sensoren hinweg erzielt werden. Dies ist die Grundlage, um einen konstanten, einstellbaren Phasenabstand zu gewährleisten und gegenseitige Blendungen benachbarten Sensoren zu verhindern. Der Phasenabstand bezeichnet hier den Versatz zwischen den Rotationswinkeln.

INS Inertiale Navigationssysteme, engl. *Inertial Navigation Systems* (INS) wie das in MuCAR-3 (Munich Cognitive Autonomous Robot Car 3rd Generation) verwendete OxTS RT3003 v1 oder das im TULF genutzte Xsens MTi-G stempeln intern ihre Messdaten mit der GNSS-Zeit. Diese Zeit beginnt am 06.01.1980 und berücksichtigt keine Schaltsekunden, daher besteht eine Abweichung zur Koordinierten Weltzeit

(UTC). Dieser Offset muss im Sensortreiber korrigiert werden, damit die Daten einen gültigen Linux-Zeitstempel erhalten.

Solange die Rechneruhr global synchronisiert ist, haben die Zeitstempel von Sensoren intern gestempelten Messdaten (z. B. INS) dieselbe Basis wie die erst im Rechner gestempelten Messdaten. Ist die Rechneruhr dagegen nicht global synchronisiert, so sind die globalen Zeitstempel in einzelnen Messdaten nicht hilfreich. Stattdessen sollte dann auch für das INS ersatzweise auf die weiter unten beschriebenen Verfahren zurückgegriffen werden, entweder analog zu den *Sensoren mit asynchroner Uhr* oder zu den *Sensoren ohne eigene Uhr*.

Die meisten INS können – analog zu GNSS-Empfängern – das PPS-Signal sowie Nachrichten im NMEA-Protokoll ausgeben und damit als Zeitquelle für weitere Sensoren verwendet werden. Moderne INS wie das OxTS RT3000 v3 können im lokalen Netzwerk als PTP-Master eingesetzt werden, um weitere Sensoren oder Rechner hiermit zu synchronisieren.

Kameraplattform MarVEye-8 Die Kameraplattform MarVEye-8 verfügt über einen separaten Echtzeitrechner vom Typ dSPACE MicroAutoBox, welcher mit einer Taktfrequenz von 1000 Hz die Plattformmotoren ansteuert und die Winkelsensoren ausliest. Da die dSPACE MicroAutoBox über keine interne Uhr verfügt, wurde im High-Level-Rechner anfangs mit einem konstanten Alter der Daten gerechnet (vgl. *Sensoren ohne eigene Uhr* weiter unten).

Im Rahmen dieser Arbeit wurde hier eine eigene Zeitsynchronisierung auf Basis von PPS und NMEA entwickelt, die ähnlich einer PLL (Phasenregelschleife, engl. *Phase-Locked Loop*) eine interne Uhr nachbildet.

Sensoren mit asynchroner Uhr

Weitere Sensoren haben eine interne Uhr, die jedoch nicht synchron zur Rechneruhr läuft:

Farbkameras Die meisten verwendeten Farbkameras vom Typ Basler ace sind mit einer internen Uhr ausgestattet, die nach jedem Hardwarereset der Kamera bei „Null“ im Jahre 1970 startet und nicht über den Kameratreiber bzw. das zugrunde liegende SDK (Softwareentwicklungssystem, engl. *Software Development Kit*) eingestellt werden kann. Der Kameratreiber führt daher ein internes Mapping zwischen Kamera- und Rechneruhr durch:

$$t_{\text{Rechner}}^{\text{AcquisitionStart}} = t_{\text{Kamera}}^{\text{AcquisitionStart}} + \Delta t_{\text{Offset}}, \quad (4.1)$$

wobei er den zyklisch bestimmten Offset zwischen beiden Uhren berücksichtigt:

$$\Delta t_{\text{Offset}} = t_{\text{Rechner}}^{\text{Sync}} - t_{\text{Kamera}}^{\text{Sync}}. \quad (4.2)$$

Für jedes übermittelte Bild lässt sich nun aus dem berechneten Zeitpunkt „Belichtungsstart“ und der Belichtungsdauer der zugehörige Zeitstempel errechnen:

$$t_{\text{Rechner}}^{\text{Bild}} = t_{\text{Rechner}}^{\text{AcquisitionStart}} + 0,5 \Delta t_{\text{Kamera}}^{\text{Exposure}} . \quad (4.3)$$

Einige moderne Kameras können ihre interne Uhr analog zu den LiDAR-Sensoren auf einen PTP-Master synchronisieren.

Sensoren ohne eigene Uhr

Andere Sensoren haben keine interne Uhr oder lassen sich aus anderen Gründen nicht synchronisieren. Hier bleibt nur, die Daten beim Eintreffen möglichst früh in der Prozesskette mit einem Zeitstempel zu versehen. Die einfachste Lösung ist die Verwendung der aktuellen Rechneruhrzeit, dabei wird allerdings jegliche Latenz vernachlässigt. Wird zusätzlich ein sensor-spezifischer Offset berücksichtigt, der die als konstant angenommene, empirisch ermittelte Latenz modelliert, so lässt sich der Fehler stark reduzieren.

IMU Die Xsens MTi-G stempelt intern ihre Messdaten mit der GPS-Zeit (Global Positioning System), dagegen verfügt die reine IMU (Inertiale Messeinheit, engl. *Inertial Measurement Unit*) Xsens MTi über keine Zeitstempel. Beim Eintreffen der Messdaten am Rechner werden diese mit der aktuelle Rechneruhrzeit, korrigiert um eine empirisch bestimmte Latenz, als Zeitstempel versehen. Diese Latenz setzt sich aus mehreren Komponenten zusammen:

$$\Delta t_{\text{IMU}} = \Delta t_{\text{IMU}}^{\text{internal}} + \Delta t_{\text{IMU}}^{\text{Transfer}} + \Delta t_{\text{Converter}}^{\text{internal}} + \Delta t_{\text{IMU}}^{\text{empirisch}} . \quad (4.4)$$

Die Sensor-interne Verarbeitungsdauer $\Delta t_{\text{IMU}}^{\text{internal}}$ kann meist anhand des Datenblatts bestimmt werden. Aus der Nutzdatengröße und der Busgeschwindigkeit kann die minimale Transferzeit $\Delta t_{\text{IMU}}^{\text{Transfer}}$ berechnet werden, allerdings ist diese bei mit anderen Komponenten geteilten Bussen wie Ethernet oder einem stark ausgelasteten CAN-Bus (Controller Area Network) mit einem Jitter, d. h. nicht-deterministischer Verzögerungen, behaftet. Falls zusätzliche Signal- und Buskonverter involviert sind, muss $\Delta t_{\text{Converter}}^{\text{internal}}$ bestimmt werden – dies ist im TULF oftmals der Fall.

In Vergleichsfahrten mit einer Referenzsensorik kann evaluiert werden, ob die aus den drei ersten Komponenten bestimmte Latenz korrekt ist, oder ob eine zusätzlich empirisch bestimmte Latenz $\Delta t_{\text{IMU}}^{\text{empirisch}}$ addiert werden muss. Für die Latenzevaluierung einer IMU kann das Fahrzeug im Stand zu Roll- oder Nickbewegungen angeregt und die Messwerte über der Zeit aufgetragen werden. Aus dem Phasenversatz ergibt sich $\Delta t_{\text{IMU}}^{\text{empirisch}}$.

Wärmebildkameras Die Wärmebildkameras sowohl in MuCAR-3 als auch im TULF basieren auf ungekühlten Mikrobolometer-Detektoren, für die eine Detektor-Zeitkonstante aus dem Datenblatt bekannt ist, aber nicht die bei Farbkameras

übliche helligkeitsabhängige Belichtungszeit. Zudem kann hier die Sensoruhr über den verwendeten Kameratreiber weder gestellt noch ausgelesen werden. Der Zeitstempel der Bilder berechnet sich daher aus aktueller Rechneruhrzeit bei Eintreffen der Daten zuzüglich eines empirisch bestimmten Offsets, der aus einem visuellen Abgleich mit dem Farbkamerabild bestimmt werden kann.

Fahrdynamikensensoren in Basisfahrzeug & DbW Die Messungen der Fahrdynamikensensoren wie z. B. Radgeschwindigkeiten, Lenkwinkel, Antriebsmoment oder Bremsdruck, werden von unterschiedlichen Steuergeräten des Basisfahrzeugs erfasst, intern auf nicht dokumentierte Art verarbeitet und anschließend mit unterschiedlichen Frequenzen, je nach Steuergerät und Signal, bereitstellt. Die verwendeten Low-Level-Rechner erfassen diese asynchron über den CAN-Bus eintreffenden Messwerte sowie analoge Signale und geben sie in festen Zyklen synchronisiert an das High-Level-System weiter.

Für MuCAR-3 werden Zeitstempel und Latenz ähnlich wie bei der IMU bestimmt. Bei TULF und StrAsRob wird die aktuelle Rechneruhrzeit beim Eintreffen der Daten als Zeitstempel verwendet.

Weitere Fahrdynamikensensoren in TULF und StrAsRob Die beiden LKWs sind mit zusätzlichen Sensoren zur Messungen der Fahrzeugbewegung ausgestattet (vgl. Abschnitt 3.5.3): Der Übergrund-Geschwindigkeitssensor DICKKEY-john RADAR III wird über ein Frequenz-Interface von einem separaten Rechner eingelesen und an das High-Level-System kommuniziert, nach aktueller Kenntnis wird seitens DBD die aktuelle Rechneruhrzeit beim Eintreffen der Daten als Zeitstempel verwendet. Der 1-Achs-Kreisel KVH DSP-3000 kommuniziert über sein RS232-Interface keinen Zeitstempel, dieser wird erst – nach der Verarbeitung im RS232-Ethernet-Signalkonverter mit unbekanntem Latenzen – beim Eintreffen im High-Level-System vergeben. Nach aktueller Kenntnis wird auch hier die aktuelle Rechneruhrzeit beim Eintreffen der Daten als Zeitstempel verwendet.

4.2.3 Fazit zur Zeitstempelung

Eine saubere Zeitstempelung aller Daten hat sich als äußerst wichtig herausgestellt. Alle Daten sollten so früh wie möglich in der Prozesskette mit einem möglichst globalen Zeitstempel versehen werden, dies schließt auch unterlagerte Bestandteile wie das Low-Level-System mit ein, und dann durch die komplette Verarbeitungskette als Metadaten kommuniziert werden. Nur durch früh eingeführte Zeitstempel lassen sich variable Latenzen in der Kommunikationskette in nachgelagerten Algorithmen sauber berücksichtigen.

Die Uhrensynchronisation der beteiligten Rechner über NTP – inklusive temporär im Netz befindlicher Diagnose- und Entwickler-Notebooks wie im TULF – hat sich als unabdingbar erwiesen, kann jedoch gelegentlich fehlschlagen. Insgesamt ist ein

Multi-Rechner-System wie im TULF oder StrAsRob jedoch anfälliger für solche Synchronisierungsprobleme als ein Ein-Rechner-System wie in MuCAR-3.

Eine Herausforderung im realen Versuchsfahrzeug sind jedoch Teilsysteme, die nicht vollständig unter eigener Kontrolle stehen, wie die Low-Level-Systeme in TULF und StrAsRob. Bestehen diese aus mehreren verketteten Teilsystemen, die asynchron mit festen, aber unterschiedlichen Zykluszeiten betrieben werden, so entsteht eine stark schwankende und kaum beobachtbare Latenz. Der hierdurch verursachte Fehler in nachgelagerten Algorithmen kann beispielsweise durch eine Annahme stärker verrauschter Messdaten modelliert werden.

Die genaue Bestimmung der Latenz als Grundlage für die Zeitstempel ist insbesondere bei Messgrößen mit hoher Dynamik essentiell, wie sie bei Roll- oder Nickbewegungen auftreten, und weniger bei eher trägen Vorgängen wie der Geschwindigkeitsentwicklung.

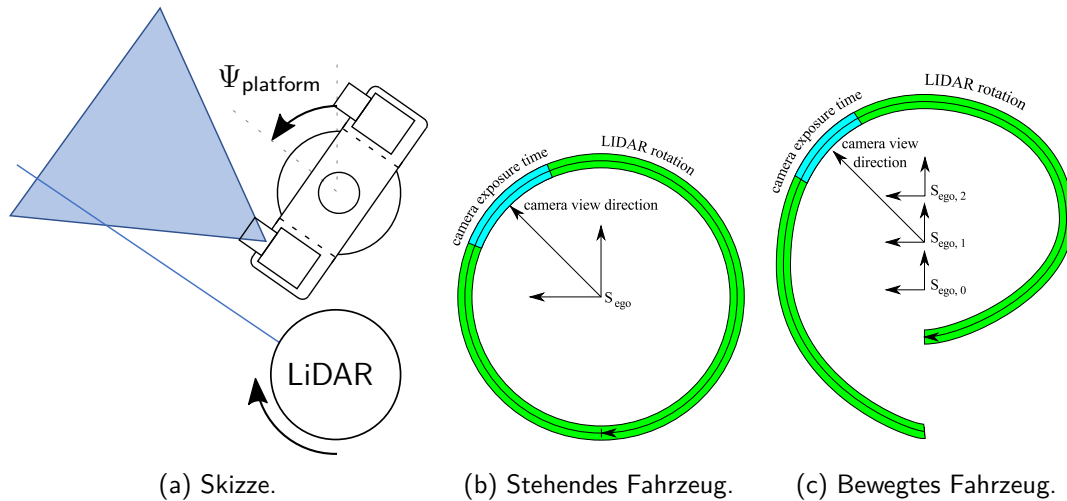
Latenzen treten jedoch nicht nur bei den Sensordaten für die Wahrnehmungsmodule auf, sondern auch in der Aktorik und durch Einflüsse des Fahrzeugs an sich als dynamisches System. Sie können damit leicht zum Aufschwingen der Regelkreise führen. Kritisch sind insbesondere nicht modellierte und kompensierte Totzeiten in der Lenkaktorik – mittels genauer Zeitstempel auf Stell- und Messgrößen können diese aber trotz mehrerer unterlagerter Systeme erfasst werden.

4.2.4 Synchronisierung

Die Kenntnis des exakten Zeitstempels allein genügt nicht für jeden Anwendungsfall in der Sensordatenfusion, sondern es muss eine zeitlich-örtliche Überlappung sichergestellt werden.

Um eine 3D-Punktwolke des LiDAR-Sensors korrekt mit Farbinformationen aus einer Kamera einfärben zu können, wäre eine im Sensorkopf integrierte rotierende Zeilenkamera ideal, sodass die Blickrichtungen der Sensoren dauerhaft kongruent sind – solch ein Sensor ist jedoch nicht verfügbar. Während der LiDAR-Sensor während der Drehung hochfrequent Punkte erfasst, machen Flächenkameras einen Snapshot über eine gewisse Belichtungszeit. Mit steigender Belichtungsdauer wächst der Winkelbereich, den der LiDAR-Sensor während der Belichtung erfasst, linear. Ohne eine Synchronisierung oder Triggerung der Sensoren ist daher die geforderte zeitlich-örtliche Überlappung nicht gewährleistet.

Zur Erzeugung der eingefärbten Punktwolke kommt in MuCAR-3 eine auf der rotierbaren Kameraplattform MarVEye-8 (Multifokales aktives / reaktives Fahrzeug-Auge der 8. Generation, engl. *Multifocal active / reactive Vehicle Eye 8th Generation*) vorhandene Farb-Flächenkamera zum Einsatz, die nun entsprechend ihrer aktuellen Blickrichtung und Belichtungszeit so von der dSPACE MicroAutoBox getriggert wird, dass der LiDAR-Sensor zur Mitte der Belichtungszeit parallel zur Kamerablickrichtung ausgerichtet ist. Abbildung 4.4 zeigt eine grafische Darstellung der Synchronisierung und Datenaufnahme. Die Umsetzung [Schneider et al., 2010] ist gemeinsam mit einem Kollegen erfolgt, der dieses in [Schneider, 2023] im Detail erläutert.

**Abbildung 4.4:**

Synchronisierung von Kamera und LiDAR: Die Kamera muss so ausgelöst werden, dass sie das Bild aufnimmt, wenn die Laserstrahlen das Sichtfeld der Kamera passieren (cyan). Bildquelle (b,c): [Schneider et al., 2010].

Ergänzend sei hier erwähnt, dass diese explizite Lösung nur für eine Kamera (oder ein Set an parallel ausgerichteten Kameras) sowie einen einzelnen LiDAR-Sensor anzuwenden ist. Für Kameras mit unterschiedlichen Blickrichtungen und vor allem für mehrere LiDAR-Sensoren mit nicht-parallelen Rotationsachsen ist eine Erweiterung erforderlich, bspw. eine sequentielle Triggerung der Einzelkameras einer 360°-Surround-Kamera wie auf dem Versuchsfahrzeug MuCAR-5 (Munich Cognitive Autonomous Robot Car 5th Generation).

4.3 Kalibrierung

Für die Sensordatenfusion ist nicht nur die zeitliche Lage der Messungen zueinander relevant, sondern auch die örtliche Lage der Sensoren, die sog. extrinsische Kalibrierung, sowie die oft nichtlinearen Abbildungseigenschaften, die mit der intrinsischen Kalibrierung bestimmt werden.

Die komplexe Frage der Kalibrierung aller Sensoren ist für die Sensordatenfusion insbesondere in den Wahrnehmungsprozessen essentiell.

Die *intrinsische Kalibrierung*, bspw. die Bestimmung der Abbildung der 3D-Umwelt durch ein verzerrendes Objektiv auf einen 2D-Kamerachip, erfolgt meist offline, für die Versuchsfahrzeuge des Instituts bspw. auf der Basis von visuellen Markern [Kallwies et al., 2020].

Für die *extrinsische Kalibrierung*, d. h. die Lage der Sensoren zueinander bzw. zu einem gemeinsamen Koordinatensystem¹, gibt es unterschiedliche Möglichkeiten,

¹Übergänge zwischen zwei kartesischen Koordinatensystemen werden mit Homogenen Transformationsmatrizen beschrieben. Eine Erläuterung hierzu ist in Anhang A zu finden.

um sie sowohl online [Horn et al., 2021] [Schneider et al., 2013], d. h. während der Fahrt und allein auf Basis der Sensordaten, als auch offline, d. h. außerhalb des Normalbetriebs und ggf. unter Zuhilfenahme von Kalibrierkörpern oder externer Messtechnik, zu bestimmen. Für die Versuchsfahrzeuge des Instituts finden sich Details in [Berthold et al., 2021].

Für Sensoren wie GNSS-Empfänger, IMUs oder Raddrehzahlsensoren, die im folgenden Kapitel Verwendung finden, genügt dagegen oft ein „händisches“ Ausmessen für die extrinsische Kalibrierung, und die intrinsische Kalibrierung bspw. eines Lenkwinkelsensors erfolgt wie in Kapitel 3 erläutert durch Messfahrten.

Daher wird das Thema Kalibrierung hier nicht weiter vertieft.

5 Eigenbewegungsschätzung und Lokalisierung

Für ein autonomes Fahrzeug ist eine präzise Kenntnis der eigenen Bewegung essentiell. Die Verwendung erstreckt sich von der Beobachtung nicht direkt messbarer Parameter für die Fahrzeugregelung über die räumlich-zeitliche Korrektur von Messdaten der Umfeldsensorik bis hin zu Anwendungen im Objekttracking. Neben der Bewegung wird eine Schätzung der globalen Lokalisierung benötigt, bspw. für die Routenplanung mithilfe von Karten. Abschnitt 5.1 geht auf die einzelnen Anforderungen ein.

In Abschnitt 5.2 werden die unterschiedlichen Koordinatensysteme für die Repräsentation der Umwelt, des eigenen Fahrzeugs sowie von Sensormessungen eingeführt.

Abschnitt 5.3 stellt unterschiedliche Methoden vor, die im Bereich der Eigenbewegungsschätzung und Lokalisierung genutzt werden.

Die im Rahmen dieser Arbeit verwendeten Varianten der rekursiven Schätzverfahren werden in Abschnitt 5.4 vorgestellt. Die dazugehörigen Modelle zur Beschreibung der Fahrzeugbewegung im Raum, die sog. Prozessmodelle, werden in Abschnitt 5.5 betrachtet.

Die folgenden Abschnitte 5.6 bis 5.8 stellen die im Rahmen dieser Arbeit entwickelten Architekturen zur Eigenbewegungsschätzung und Lokalisierung auf Basis eines flexiblen Sensorsetups und den zugehörigen Messmodellen vor. Eine Berücksichtigung von umgebungsabhängigen Besonderheiten wird in Abschnitt 5.9 vorgestellt, bevor Abschnitt 5.10 Ergebnisse der beiden Architekturen zeigt. Abschließend zieht Abschnitt 5.11 ein Fazit und wirft einen Blick auf mögliche Erweiterungen.

5.1 Motivation

Nahezu alle Module in einem autonomen Versuchsfahrzeug benötigen Informationen aus der Eigenbewegungsschätzung und Lokalisierung. Die *Eigenbewegungsschätzung* nutzt diverse am Fahrzeug vorhandene Sensoren für die Schätzung der Fahrzeugdynamik an einem einheitlichen Fahrzeugreferenzpunkt, statt in den Algorithmen je nach Sensorposition und -art einzelne fehlerbehaftete Daten zu nutzen. Zu den bereitgestellten Daten gehören Beschleunigungen, Geschwindigkeiten, Winkelgeschwindigkeiten und die Orientierung ggü. der Ebene (Rollwinkel Φ und Nickwinkel Θ), aber auch beobachtete Sensorfehler.

Abnehmer ist hier vor allem die (überlagerte) Fahrzeugregelung, die zur Führung des Fahrzeugs entlang einer Bahn dient. Dafür benötigt sie Informationen aus der Eigenbewegungsschätzung zur internen Prädiktion der Bewegung und zur Totzeitkompensation, bspw. Geschwindigkeit, Lenkwinkel und nicht direkt messbare Größen wie den Schwimmwinkel. Mitgeschätzte Sensorfehler, z. B. ein Lenkwinkeloffset, können zur Verbesserung der Stellgrößen dienen.

Die *Lokalisierung* baut integral auf der Eigenbewegungsschätzung auf und stellt die Pose (bestehend aus Position und Orientierung) des bewegten Versuchsfahrzeugs ggü.

einem ortsfesten Koordinatensystem bereit, wobei je nach Anwendungszweck zwei verschiedene Systeme zur Anwendung kommen: einerseits das sog. Dead-Reckoning bzw. die Koppelnavigation, bei dem eine Pose sprungfrei aber driftbehaftet aufintegriert wird, und andererseits eine global referenzierte Pose in einem festen Koordinatensystem, die bspw. durch GNSS-basierte (Globales Navigationssatellitensystem, engl. *Global Navigation Satellite System*) Positionsmessungen gestützt wird und springen kann.

Für die Lokalisierung sind die Abnehmer vielfältiger:

Inertiale Korrektur von Sensordaten Ein Beispiel sind Aufnahmen von rotierenden LiDAR-Sensoren (Light Detection And Ranging), aus denen während der Fahrt aus einer oder mehreren Umdrehungen eine konsistente Punktwolke generiert werden soll. Ohne Korrektur würde ein statisches Objekt in unterschiedlichen Entfernungen in der Punktwolke auftreten, ebenso in allen nachfolgenden Verarbeitungsschritten wie den Hinderniskarten. Wenn für jede Messung die aktuelle Position des Fahrzeugs bzw. eine Relativbewegung seit der letzten Messung bekannt ist, kann jede Messung zur Akkumulation in ein ortsfestes Koordinatensystem transformiert werden. Abb. 4.4c zeigt exemplarisch, wie ein Strahl während einer Umdrehung den Boden trifft. Hier gilt „je hochfrequenter die Eigenbewegungsschätzung, desto besser“ – da die Messfrequenz der INS (Inertiales Navigationssystem, engl. *Inertial Navigation System*) (100 Hz) mit der Interpolation ihrer Messdaten ausreicht, sollte dies auch die Mindestfrequenz für die Eigenbewegungsschätzung sein. Für diese Anwendung ist keine globale Referenzierung notwendig, die Daten müssen nur für einige Sekunden bzw. ca. 100 m konsistent sein, daher kommt das driftbehaftete Dead-Reckoning zum Einsatz.

Tracking-Anwendungen Für sensorbasierte Wahrnehmungsalgorithmen wie das Objekttracking oder die Fahrspurerkennung spielt die globale Referenz ebenfalls keine Rolle, im Vordergrund steht die Relativlage aus Sicht des Ego-Fahrzeugs. Für die Prädiktion der eigenen Bewegung und der Relativlage der anderen Objekte kommen die Daten der Eigenbewegungsschätzung wie Roll- und Nickwinkel und die Lokalisierung aus dem Dead-Reckoning zum Einsatz.

Kartennutzung Die global referenzierte Lokalisierung findet dagegen in allen Anwendungen, in denen Kartenwissen zum Einsatz kommt, Verwendung. Neben einer globalen Routenplanung sind dies die im weiteren Verlauf der Arbeit verwendeten Rückfallebenen zur GNSS- und kartenbasierten Spur- und Kreuzungsgenerierung. Wenn die Rückfallebene direkt zur Fahrzeugführung genutzt werden soll, ist eine Lokalisierungsgenauigkeit im Zentimeter-Bereich erforderlich, d. h. die Verwendung von RTK-GNSS. Dienen die Kartendaten nur als Hypothese für sensorgestützte Verfahren, genügt eine Meter-Genauigkeit.

Für die meisten Anwendungen wie beim TULF genügt eine 5-DOF-Pose (Freiheitsgrade, engl. *Degrees of Freedom*): eine x - y -Position in der Welt und drei Rotationswinkel.

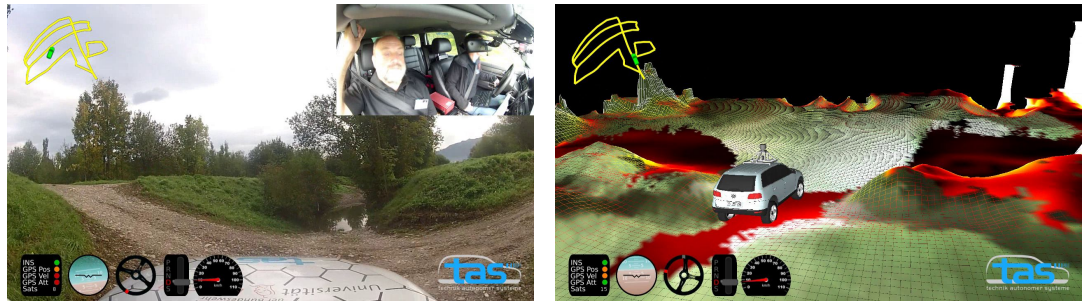


Abbildung 5.1:

Die Topographie des Wettbewerbsgeländes auf der M-ELROB 2012 erfordert eine vollständige 6-DOF-Eigenbewegungsschätzung und Lokalisierung.

Erst sehr welliges Gelände wie in Abb. 5.1 auf der M-ELROB (Military European Land Robot Trial) 2012 erfordert zwingend eine vollständige 6-DOF-Pose, um eine fehlerhafte Hinderniskartierung zu vermeiden.

Einschränkungen von INS

Die meisten der genannten Versuchsfahrzeuge verfügen bereits über hochwertige INS – da stellt sich natürlich die Frage, warum diese nicht als alleinige Datenquelle für Eigenbewegung und Lokalisierung verwendet werden.

Ein Grund ist in Abb. 5.2 dargestellt und erläutert: Ein Loosely-Coupled INS wie das verwendete OxTS RT3003 v1 erhält fertige Positionslösungen aus einem integrierten GNSS-Empfänger (hier nur GPS und GLONASS), der dazu mindestens vier Satelliten erkennen und anhand des PDOP-Wertes (Positional DOP) für geeignet befinden muss. Weichen die Positionslösungen wie im Bild gezeigt aufgrund von Störungen über lange Strecken ab, so können sie anfangs vom Filter im INS entweder über den PDOP-Wert oder über die Innovationskovarianz aussortiert werden, während die Position auf Basis der IMU (Inertiale Messeinheit, engl. *Inertial Measurement Unit*) und rudimentärer Odometrie (Radpulse eines einzelnen Rads) weiter propagiert wird. Je hochwertiger die integrierte IMU (bspw. Faseroptische Kreisel statt MEMS-Kreisel (Mikrosystem, engl. *Micro-Electro-Mechanical System*)) und je hochauflösender die Odometrie, desto langsamer wächst die Schätzfehlerkovarianz.

Ab einem gewissen Zeitpunkt ist sie aber so weit angewachsen, dass auch die fehlerhaften Messungen innerhalb der Schätzfehlerkovarianz liegen und – trotz möglicher Abweichung zum bisherigen Schätzwert – eingefiltert werden und damit das Ergebnis verschlechtern.

Ein eigener Algorithmus zur Eigenbewegungsschätzung und Lokalisierung kann ein zur Anwendung passendes Prozessmodell nutzen, welches die Bewegungsmöglichkeiten bspw. eines vorderachsgelenkten Fahrzeugs berücksichtigt, und weitere Sensorik wie Einzelradodometrie oder den Lenkwinkel mit einbeziehen. Durch beide Maßnahmen

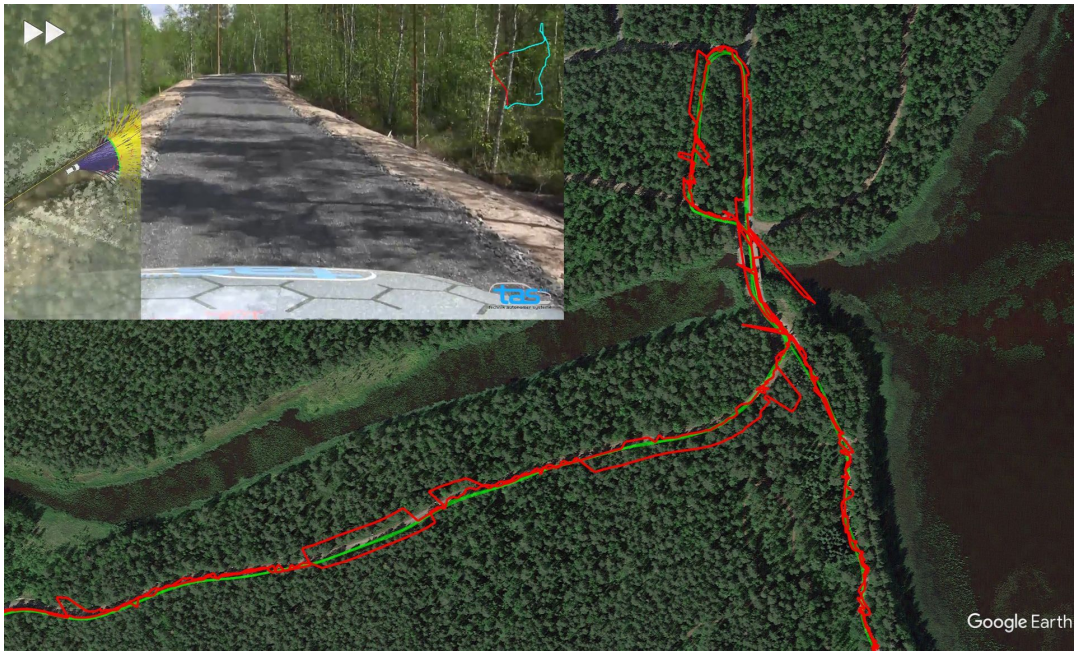


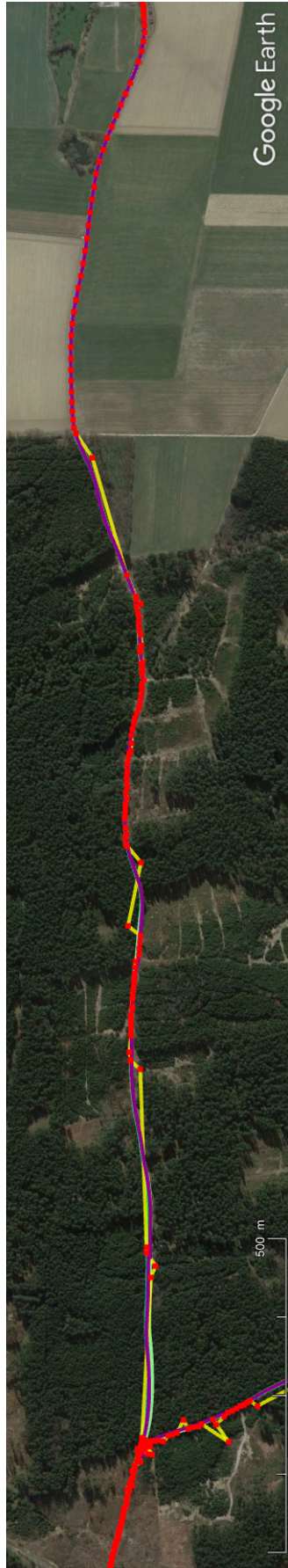
Abbildung 5.2:

GNSS-Positionsmessungen unter suboptimalen Bedingungen auf der C-ELROB 2009: Während längerer Fahrt entlang eines schmalen Waldwegs mit stark eingeschränkter Sicht zum Himmel liefert der GNSS-Empfänger noch durchgängig eine Position —, diese ist aber teilweise über 100 m Strecke um viele Meter parallel versetzt. Grund hierfür ist eine am hohen PDOP-Wert (bestenfalls 3, streckenweise >8 , im Bereich des Parallelversatzes >20) erkennbare schlechte Satellitenkonstellation (wenige Satelliten, ungünstige Anordnung aufgrund der Waldschneise). Über diesen Zeithorizont konnten die Messungen anhand des PDOP-Werts oder der Innovationskovarianz durch die Algorithmen im INS noch aussortiert werden, so dass dessen Positionslösung — hiervon noch nicht merklich verfälscht wird.

kann eine höhere Robustheit gegen fehlerhafte Messungen geschaffen werden. Zusätzlich können weitere Relativbewegungsmessungen und globale Positionsmessungen integriert werden, um fehlende GNSS-Messungen zu kompensieren.

Abbildung 5.3 zeigt die Ergebnisse einer Vergleichsfahrt mit unterschiedlichen INS der Preisklassen 30 k€ (iMAR iNAT-M200 —) über 50 k€ (OxTS RT3003 v2 —) bis 120 k€ (iMAR iNAT-FSLG-01 —) unter suboptimalen Bedingungen im Wald, Abb. 5.4 die Fortsetzung unter schweren Bedingungen nach längerer Zeit im Wald inklusive Rangieren und bis zu 1 km Strecke ohne gültige GNSS-Messungen. Zur Stützung kam hier für alle Sensoren ein optischer Geschwindigkeitssensor vom Typ LUXACT 1D Compact [SMG Technologie GmbH, 2018] statt der Odometripulse eines serienmäßigen ABSensors zum Einsatz.

Die Ergebnisse zeigen, dass selbst das mit faseroptischen Kreiseln ausgestattete iMAR iNAT-FSLG-01 (—) als Loosely-Coupled INS solch lange Aussetzer nicht kompensieren kann. Beim Tightly-Coupled INS wird dagegen nicht die Positionslösung des GNSS-Empfängers verwendet, sondern es werden die sog. Pseudorange-Messungen jedes einzelnen Satelliten direkt im Filter des INS berücksichtigt. Dies führt zu einer








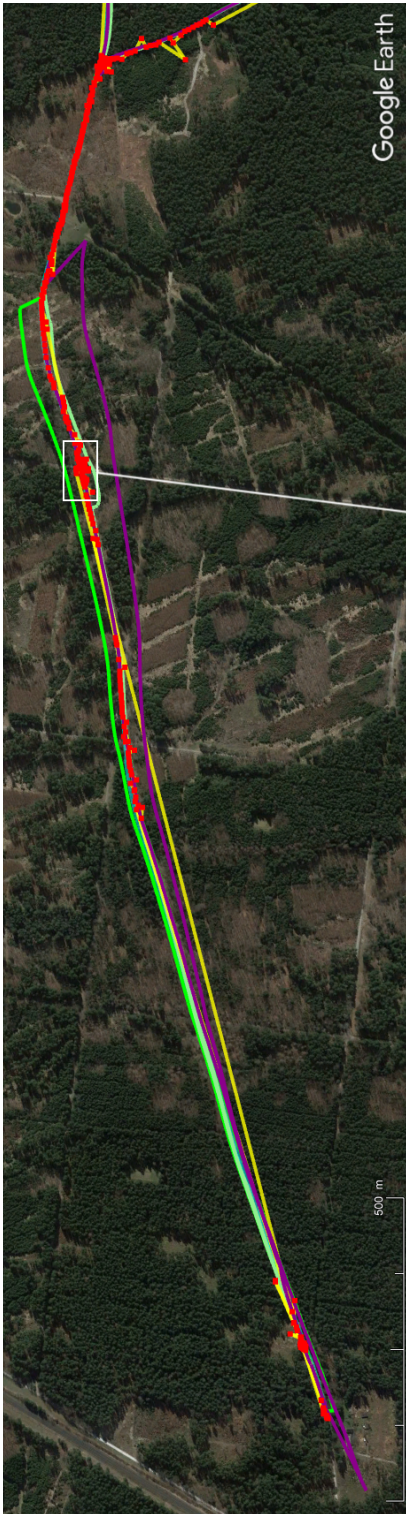
(a) Fahrt vom freien Feld (rechts) in den Wald hinein, ca. 2500 m.



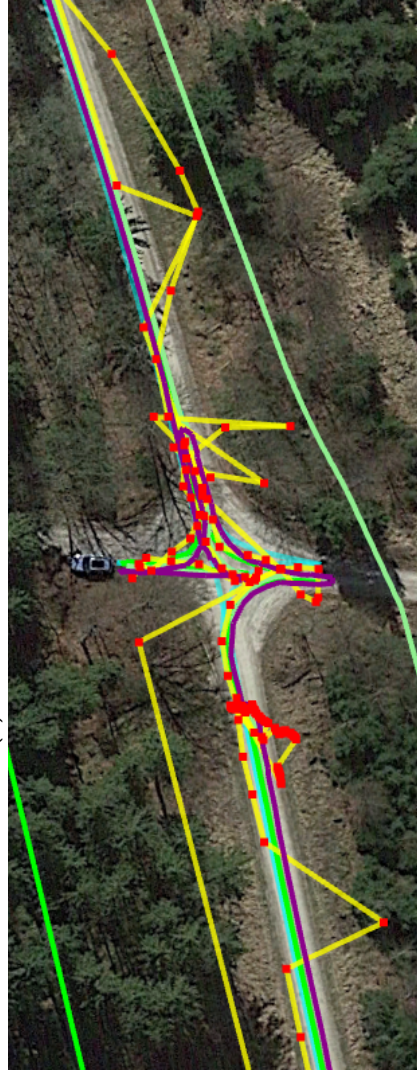
(b) Ausschnitt, Gesamtlänge ca. 500 m.

Abbildung 5.3:

GNSS-Positionsmessungen unter suboptimalen Bedingungen: Die Genauigkeit und vor allem Verfügbarkeit der GNSS-Positionsmessung (Novatel OEM6 ) nimmt ab. Zum Vergleich sind die Positionslösungen unterschiedlicher INS gezeigt, die hier noch weitgehend identisch und auf dem Weg liegen (iMAR iNAT-M200 , iMAR iNAT-FSLG-01 , OxTS RT3003 v2 [tightly-coupled] , OxTS RT3003 v2 [loosely-coupled] ). Im Zoom (b) ist zu erkennen, dass das iMAR iNAT-M200 nach der GNSS-Lücke stärker abweicht und dann mit einer Korrektur nach oben reagiert.







(a) Fahrt durch den Wald, ca. 2500 m



(b) Ausschnitt, ca. 120 m

Abbildung 5.4:

GNSS-Positionsmessungen unter schlechten Bedingungen: (a) zeigt die Walddurchfahrt von rechts nach links. An der weiß markierten Kreuzung im rechten Bildddrittel, die in (b) vergrößert gezeigt wird, erfolgt eine doppelte Dreipunktwende. Diese ist qualitativ bei allen INS gut zu erkennen, wenn auch mit unterschiedlichen Abweichungen. Anschließend führt die Fahrt weiter nach links bis zum Bildende. Nach einer kurzen Rückwärtsfahrt im linken Bereich erfolgt eine Dreipunktwende, dann wird die Fahrt nach rechts bis zur Abzweigung am Bildrand fortgesetzt. Auf diesem zweiten Teil sind auf ca. 1 km Strecke keine gültigen GNSS-Positionsmessungen (Novatel OEM6 ) vorhanden, was bei den meisten INS zu großer Drift führt (iMAR iNAT-M200 , iMAR iNAT-FSLG-01 , OxTS RT3003 v2 [loosely-coupled] ). Das OxTS RT3003 v2 nutzt ein Tight-Coupling und ist das einzige INS, dessen Positionsschätzung auf dem Rückweg weitgehend auf der Straße liegt (zu erkennen an gerader Linie über die Kreuzung).

Performancesteigerung im Wald und anderen herausfordernden GNSS-Umgebungen, da eine Stützung des Filters auch bei weniger als vier sichtbaren Satelliten stattfinden kann (OxTS RT3003 v2 mit einem Beta-Softwarestand [—](#)).

Kommt statt dieser teuren Sensorik ein günstiger GNSS-Empfänger mit einer günstigen IMU (z. B. XSens MTi-G) zum Einsatz, sehen die Ergebnisse um ein Vielfaches schlechter aus, was den möglichen Performancegewinn durch eine selbst entwickelte Eigenbewegungsschätzung und Lokalisierung verstärkt.

5.2 Koordinatensysteme

In diesem Abschnitt werden die im weiteren Verlauf des Kapitels verwendeten Koordinatensysteme und ihre Beziehung zueinander vorgestellt. Dabei wird in Anlehnung an Wendel [2011] zwischen raum- bzw. erdfesten Koordinatensystemen (vgl. Abschnitt 5.2.2) und mitbewegten Koordinatensystemen (vgl. Abschnitt 5.2.3) unterschieden.

Übergänge zwischen zwei kartesischen Koordinatensystemen werden mit Homogenen Transformationsmatrizen beschrieben. Eine Erläuterung bzw. Herleitung hierzu ist in Anhang A zu finden.

Ergänzend werden konkrete Nutzungsbeispiele für die einzelnen Koordinatensysteme vorgestellt, z. B. in der Sensorik und den Algorithmen.

Hinweis: Der Inhalt dieses Abschnitts wurde in einer frühen Fassung in [Lüttel et al., 2015, Kap. 3.1] für einen begrenzten Leserkreis bereitgestellt.

5.2.1 Erdmodelle

Erdmodelle umfassen die wichtigsten Parameter der Erdfigur, der Erdrotation und des Schwerefeldes. Diese Aspekte werden im Folgenden behandelt.

5.2.1.1 Beschreibung der Erdfigur durch Referenzellipsoide

Bereits in der Antike wurde die Figur der Erde als Kugel beschrieben, allerdings gibt es bereits seit dem 17. Jahrhundert Zweifel an dieser Figurhypothese. Zur geometrischen Beschreibung der Erdfigur kommen seitdem regelmäßig an den Polen abgeplattete Rotationsellipsoide zur Anwendung, die die Form der Erde in erster Näherung gut beschreiben. Durch die Modellierung als Rotationsellipsoid ist nur die z -Komponente eines Schwerebeschleunigungsvektors von Null verschieden.

Ein häufig verwendetes Referenzellipsoid ist das Bessel-Ellipsoid (auch Bessel 1841 genannt), welches auf großflächigen Vermessungen in Europa, Russland, Indien und Peru basiert. Aufgrund dieser Datengrundlage passt es sich dem Geoid und der mittleren Erdkrümmung in Eurasien recht gut an und wurde daher in vielen Landesvermessungen als Grundlage verwendet, unter anderem in Deutschland.

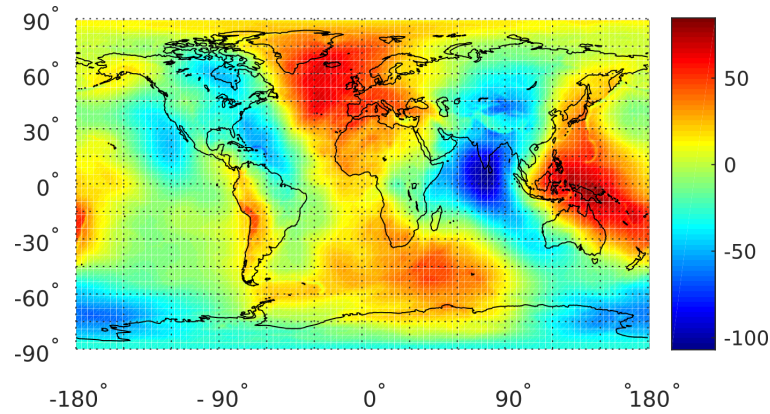


Abbildung 5.5: Weltweite Verteilung der Geoidhöhe N des EGM 96. Grafik erzeugt mit Hilfe von [MathWorks, 2023].

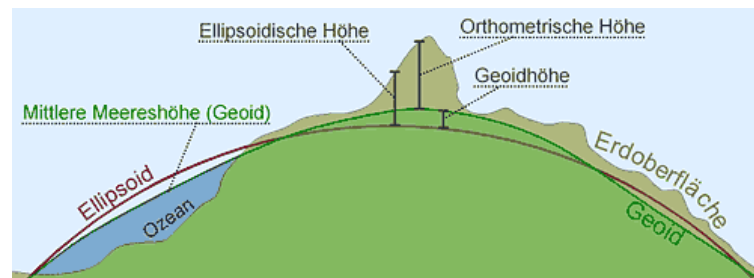


Abbildung 5.6: Vergleich unterschiedlicher Höhenangaben: Ellipsoidische Höhe h vs. Geoidhöhe N vs. orthometrische Höhe H . Bildquelle: <http://www.kowoma.de/gps/geo/mapdatum.htm>

Weltweit wird heute meist das WGS 84-Referenzellipsoid (World Geodetic System 1984) verwendet, welches auch im globalen Satellitennavigationssystem GPS (Global Positioning System)¹ Verwendung findet. Seine Parameter entsprechen in etwa denen des GRS 80-Referenzellipsoiden (Geodätisches Referenzsystem 1980). Die drei anderen GNSS-Systeme GLONASS, Galileo und Beidou orientieren sich ebenfalls am ITRF (International Terrestrial Reference Frame) und nutzen faktische identische Referenzellipsoide.

Die Referenzellipsoide werden durch die *Länge der Äquatorhalbachse* a , die *Länge der Polhalbachse* b und die sich daraus ergebende *Abplattung* $f = \frac{a-b}{a}$ definiert.

5.2.1.2 Beschreibung des Schwerfeldes durch das Geoid

Im Gegensatz zu der geometrischen Darstellung des Referenzellipsoiden ist das Geoid ein physikalisches Modell der Erdfigur, das die Oberfläche als Fläche gleichen Schwerepotentials beschreibt. Außerhalb der kontinentalen Landmassen ist das Geoid in der mittleren Höhe des Meeresspiegels sichtbar. Es gibt unterschiedliche

¹Die offizielle Bezeichnung lautet *Navigational Satellite Timing and Ranging* GPS (NAVSTAR GPS, zur Vereinfachung wird in dieser Arbeit jedoch die Kurzform GPS genutzt.

Tabelle 5.1: Referenzellipsoide der verwendeten Erdmodelle.

Name	Jahr	Äquatorhalbachse (a)	Numerus = $1/\text{Abplattung}$ ($n = \frac{1}{f}$)
Bessel 1841	1841	6 377 397,155 m	299,152 812 8
GRS 80	1980	6 378 137,0 m	298,257 222 101
WGS 84	1984	6 378 137,0 m	298,257 223 563

Darstellungen des Geoids, weit verbreitet ist das von Fell [1997] vorgestellte Earth Gravitational Model 1996 (EGM 96). Die Geoidhöhe N ist hier der Abstand des Geoids vom WGS 84-Referenzellipsoiden und wird auch als Geoid-Undulation bezeichnet. Werte außerhalb des Ellipsoiden haben ein positives Vorzeichen.

Die orthometrische Höhe H gibt die Höhe eines Punktes über dem Geoid an, sie wird längs der gekrümmten Lotlinie durch den Punkt gemessen. Höhen außerhalb des Geoids sind positiv. Umgangssprachlich wird die orthometrische Höhe auch als „Höhe über Normal-Höhen-Null“ bezeichnet. Dabei ist zu beachten, dass in verschiedenen Ländern leicht unterschiedliche Normal-Null-Meeresspiegel als Bezugspunkt verwendet werden.

Abbildung 5.5 zeigt die weltweite Verteilung der Geoidhöhe. Abbildung 5.6 visualisiert den Zusammenhang von Geoidhöhe N , ellipsoidischer Höhe h sowie orthometrischer Höhe H , damit ergibt sich:

$$h \approx H + N. \quad (5.1)$$

5.2.1.3 Verwendete Erdmodelle

Die nachfolgenden, weit verbreiteten Erdmodelle kommen in dieser Arbeit zum Einsatz:

GRS 80 ist ein Erdmodell, welches die wichtigsten Parameter der Erdfigur, der Erdrotation und des Schwerefeldes (Geoid) umfasst. Die Erdfigur wird durch ein Referenzellipsoid beschrieben, dieses Ellipsoid wird auch im ETRS89 (Europäisches Terrestrisches Referenzsystem 1989) als geometrische Rechen- und Abbildungsfläche verwendet. Als Abbildungssystem wird dabei die konforme UTM-Abbildung eingesetzt.

WGS 84 ist ein geodätisches Referenzsystem als einheitliche Grundlage für Positionsangaben auf der Erde und im erdnahen Weltraum, welches u. a. vom GPS verwendet wird. WGS 84 verwendet annähernd das GRS 80-Ellipsoid, enthält aber im Vergleich zu GRS 80 mehr Daten über das Schwerefeld und beschreibt daher die Form präziser.

In Tabelle 5.1 findet sich ein Vergleich der Parameter der zugehörigen Referenzellipsoide.

5.2.2 Raum- und Erd feste Koordinatensysteme

5.2.2.1 Inertialkoordinatensystem

Für die Form der Erde wird im Folgenden ein Rotationsellipsoid als Näherung angenommen. In dessen Mittelpunkt liegt der Ursprung des kartesischen Inertialkoordinatensystems $\mathbf{F}_{\text{inertial}}$. Die Achsen sind fest im Bezug zu den Fixsternen: Die z_{inertial} -Achse zeigt zum Nordpol und ist identisch mit der Rotationsachse der Erde. x_{inertial} und y_{inertial} spannen die Äquatorebene auf.

Das Inertialkoordinatensystem $\mathbf{F}_{\text{inertial}}$ spielt eine Rolle für die Winkelgeschwindigkeitsmessungen von IMUs.

5.2.2.2 Erdfestes Koordinatensystem WGS 84

Das erdfeste, geozentrische Koordinatensystem wird in der Literatur [Zogg, 2011, Wendel, 2011] häufig als Earth Centered, Earth Fixed (ECEF) bezeichnet, daher wird hier \mathbf{F}_{ECEF} als Bezeichnung verwendet.

\mathbf{F}_{ECEF} hat denselben Ursprung und dieselbe z_{ECEF} -Achse wie das Inertialkoordinatensystem $\mathbf{F}_{\text{inertial}}$, rotiert gegenüber diesem allerdings einmal pro Tag um diese Achse sowie einmal pro Jahr um die Sonne. Das entspricht einer Winkelgeschwindigkeit von

$$\dot{\Psi}_{\text{inertial}} \approx \frac{360^\circ}{24 \cdot 60 \cdot 60 \text{ s}} \left(1 + \frac{1}{365,25} \right) \approx 15,04^\circ/\text{h}. \quad (5.2)$$

Die x_{ECEF} -Achse ergibt sich als Schnittpunkt der Äquatorebene und der Ebene durch den Null- bzw. Greenwich-Meridian, die y_{ECEF} -Achse vervollständigt das rechtshändige Koordinatensystem; sie ist gegenüber x_{ECEF} um 90° in östlicher Richtung verdreht. Abbildung 5.7 zeigt dieses Koordinatensystem.

WGS 84 ist das wohl am häufigsten verwendete erdfeste Koordinatensystem und kommt unter anderem im GPS zum Einsatz. Zugehörig ist die Definition des weltweiten Referenzellipsoids WGS 84, das als global bestangepasstes Rotationsellipsoid die Erdoberfläche geometrisch beschreibt. Vor der Verfügbarkeit von GPS und der Einführung von WGS 84 hat jedes Land „sein eigenes“ lokal bestangepasstes Rotationsellipsoid verwendet. Vereinfachend beschrieben, wird dazu das Bessel-Ellipsoid ggü. dem Geoid so verschoben, dass die Geoid-Undulation N für den Wunschbereich minimiert wird. Damit ergab sich für Deutschland das *Datum Potsdam*, während für Österreich das *Datum Austria* am besten geeignet war.

Kartesische geozentrische Koordinaten im \mathbf{F}_{ECEF} -System werden in der Praxis selten verwendet, stattdessen kommen meist die ellipsoidischen bzw. geodätischen Koordinaten \mathbf{Y}_{LLH} (Breitengrad φ , Längengrad λ , Höhe h , engl. *Latitude, Longitude, Height*) zum Einsatz. Die geographische Länge λ_{LLH} beschreibt die Rotation um z_{LLH} mit dem Wertebereich $-180^\circ \leq \lambda_{\text{LLH}} \leq 180^\circ$. Die geographische Breite φ_{LLH} gibt den Winkel an, mit der die lokale Vertikale die Äquatorebene schneidet. Der Wertebereich

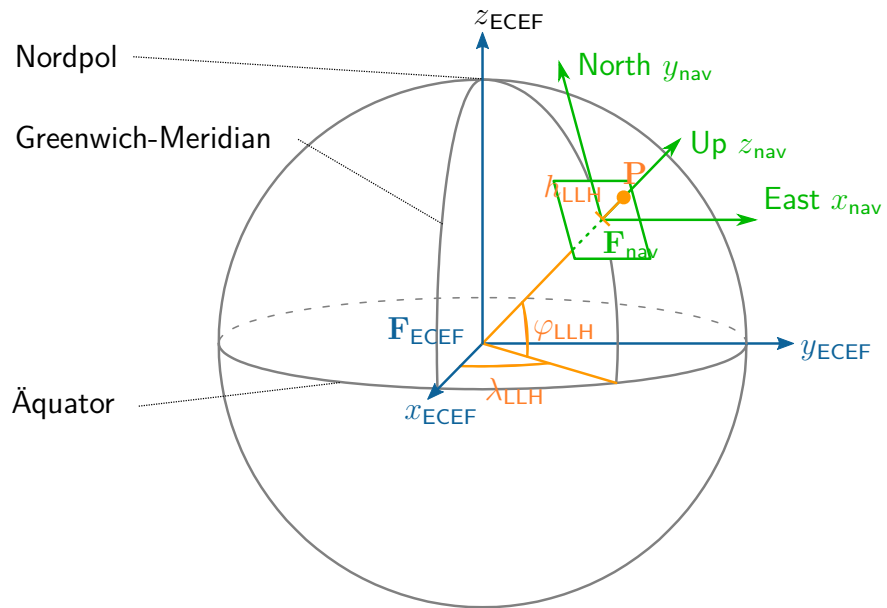


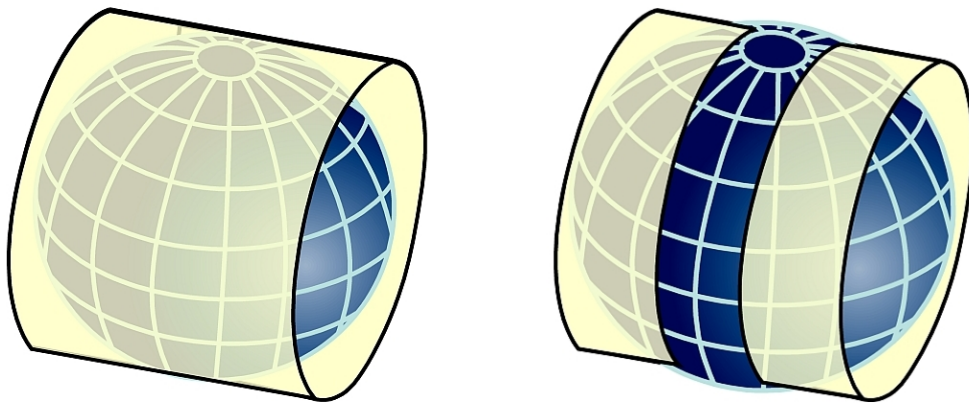
Abbildung 5.7:

Darstellung des erdfesten geozentrischen Koordinatensystems F_{ECEF} , der ellipsoidischen geographischen Koordinaten Y_{LLH} und des Navigationskoordinatensystems F_{nav} (in der ENU-Variante). Der Punkt P befindet sich in der Höhe h über dem Rotationsellipsoid. Hinweis: Die Erde ist hier vereinfachend als Kugel gezeichnet, beim eigentlichen Rotationsellipsoid schneidet z_{nav} nicht den Erdmittelpunkt.

beträgt $-90^\circ \leq \varphi_{LLH} \leq 90^\circ$. Diese Vertikale geht nicht durch den Erdmittelpunkt, da die Erde nicht als Kugel sondern als Rotationsellipsoid approximiert wird. Die ellipsoidische Höhe h_{LLH} gibt den Abstand eines Punktes vom Referenzellipsoiden an, gemessen längs der Ellipsoidnormalen. Die Transformationen zwischen F_{ECEF} und Y_{LLH} sind beispielsweise in [Wendel, 2011, Kap. 3] zu finden.

Verwendung in dieser Arbeit Die meisten GNSS-Empfänger liefern Positions-Messwerte im erdfesten Koordinatensystem Y_{LLH} , und auch die fusionierten Daten eines INS sind in Y_{LLH} gegeben. Je nach Sensor unterscheidet sich jedoch das Bezugssystem der Höhe: während das günstige INS Xsens MTi-G eine ellipsoidische Höhe h_{LLH} bereitstellt, wendet das hochpreisige INS Oxts RT3003 v1 standardmäßig intern ein Geoidmodell an und liefert eine orthometrische Höhe H_{LLH} .

Bewertung Vorteil bei Werten im erdfesten Koordinatensystem WGS 84 ist, dass drei Zahlen die Position auf der gesamten Welt stetig beschreiben. Es gibt keine Zonengrenzen wie in den im Folgenden beschriebenen ebenen Koordinatensystemen, die zu Unstetigkeiten und Sprüngen führen können. Als Nachteil ist zu erwähnen, dass die Berechnung in ellipsoidischen Koordinaten nicht so einfach verständlich ist wie im kartesischen Raum.



(a) Berührzylinder der Gauß-Krüger-Abbildung (b) Schnittzylinder der UTM-Abbildung

Abbildung 5.8:

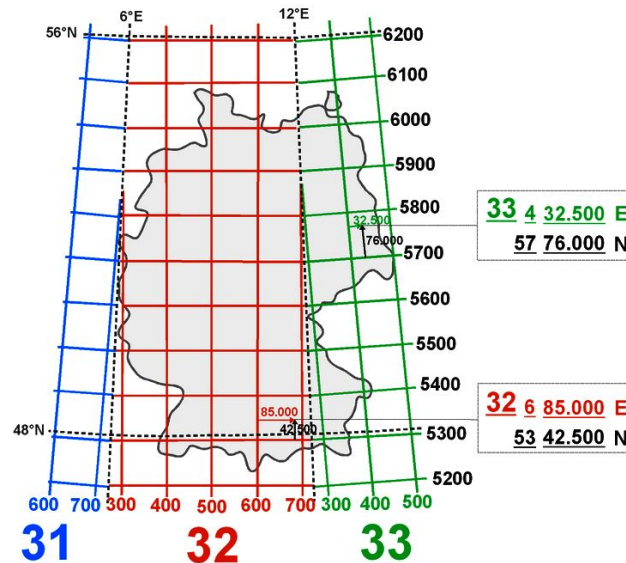
Vergleich der Zylinderprojektion von Gauß-Krüger- und UTM-Koordinaten, die Streifenbreite der Meridianstreifen ist stark überhöht dargestellt. Bildquelle: https://www.wuppertal.de/rathaus-buergerservice/karten_vermessung/kataster/ETRS89_UTM.php.

5.2.2.3 Ebene Koordinatensysteme

Ebene Koordinatensysteme erhält man durch Projektion eines geographischen Koordinatenreferenzsystems in die Ebene. Historisch bedingt gibt es hier viele unterschiedliche Systeme in den unterschiedlichen Regionen der Erde. Die für diese Arbeit wichtigsten werden im Folgenden vorgestellt.

Gauß-Krüger-Koordinaten Die bisherigen amtlichen Koordinaten des Vermessungswesens in Deutschland basierten auf dem Gauß-Krüger-Meridianstreifen-System. Dabei werden zur Kartenprojektion auf einem der Erdkugel angeschmiegtten Berührzylinder 3° breite Streifen verwendet, wobei der mittlere Meridian jeweils längengetreu abgebildet wird (vgl. Abb. 5.8a). Die Gauß-Krüger-Koordinaten werden in Rechts- und Hochwerten angegeben. Bezugssystem für das Liegenschaftskataster ist dabei das „Deutsche Hauptdreiecksnetz 1990“ (DHDN).

UTM-Koordinaten Das UTM-Koordinatensystem (Universal Transverse Mercator) F_{UTM} ist ein globales Koordinatensystem, welches das GRS 80- resp. WGS 84-Ellipsoid zur geometrischen Beschreibung verwendet. Die Erdoberfläche wird streifenförmig in 6° breite Meridianstreifen aufgeteilt, eine Visualisierung am Beispiel Deutschland ist in Abb. 5.9 zu finden. Jede Zone wird separat mit der jeweils günstigsten transversalen Mercator-Projektion verebnet und mit einem kartesischen Koordinatensystem überzogen. Um die Strecken-Verzerrungen am Rand der Streifen gering zu halten, wird ein Schnittzylinder für die Projektion in die Karte verwendet –

**Abbildung 5.9:**

UTM-Koordinaten für Deutschland. Die Nord- und Ostachsen in den einzelnen Meridianstreifen (farbige durchgezogene Linien, Achsenbeschriftung rechts und unten) sind aufgrund der Projektion nicht parallel zu den Längen- und Breitengraden (schwarze gestrichelte Linien, Achsenbeschriftung oben und links). Bildquelle: <http://gps0.de/navigation/utm.html>

im Gegensatz zum Berührzylinder bei den Gauß-Krüger-Koordinaten. Abbildung 5.8 zeigt die Unterschiede. Diese Zonen sind gültig im Bereich von 80° Süd bis 84° Nord, für die Polkappen wird abweichend die Universale Polare Stereografische Projektion (UPS) verwendet. Die Koordinaten werden dabei Ost- und Nord-Werte genannt:

$$x_{\text{UTM}} = \text{Ost / East} \quad \text{Äquator} \quad (5.3)$$

$$y_{\text{UTM}} = \text{Nord / North} \quad \text{Mittelmeridian} \quad (5.4)$$

$$z_{\text{UTM}} = \text{Höhe / Altitude} \quad \text{vervollständigt das rechtshändige System} \quad (5.5)$$

Bezugssystem in der Landesvermessung ist zukünftig das europaweit einheitliche „ETRS89“. Sowohl in Deutschland als auch in Österreich ist die Umstellung in der Vermessung auf UTM-Koordinaten weitgehend abgeschlossen, auch in der Landesvermessung anderer Staaten finden diese zunehmend Anwendung. Damit verliert das mit UTM verwandte Gauß-Krüger-Koordinatensystem langfristig an Bedeutung, kommt aber immer noch in Teilbereichen zum Einsatz.

Bewertung Als Vorteil gegenüber ellipsoidischen Koordinaten überwiegt hier für den Nutzer ganz klar das kartesische, metrische System von UTM.

Im Gegensatz zu den ellipsoidischen Koordinaten treten bei allen ebenen Koordinatensystemen Sprünge beim Überqueren von Zonengrenzen auf. Diese lassen sich – unter Inkaufnahme etwas größerer Ungenauigkeiten und leicht begrenztem Aktionsradius –

vermeiden, wenn die UTM-Zone beim Einschalten des Systems festgelegt und auch beim Überqueren der Zonengrenze daran festgehalten wird.

Nutzung In dieser Arbeit und allgemein bei TAS wird das ebene Koordinatensystem \mathbf{F}_{UTM} für alle globalen Berechnungen genutzt, von der Eigenbewegungsschätzung und Lokalisierung bis zur Missionsplanung. Die Transformation, sowohl zwischen erdfestem Koordinatensystem \mathbf{Y}_{LLH} und ebenem Koordinatensystem \mathbf{F}_{UTM} als auch zwischen \mathbf{F}_{UTM} und anderen ebenen Koordinatensystemen, erfolgt mithilfe der weit verbreiteten Open-Source-Softwarebibliothek PROJ².

Die gemessene Position aus dem GNSS-Empfänger wird von \mathbf{Y}_{LLH} nach \mathbf{F}_{UTM} transformiert, damit ergibt sich die Position ${}^{\text{UTM}}\mathbf{p}^{\text{GNSS}}$:

$${}^{\text{UTM}}\mathbf{p}^{\text{GNSS}} = \text{proj}({}^{\text{LLH}}\mathbf{p}^{\text{GNSS}}) = \text{proj}(\varphi, \lambda, h) . \quad (5.6)$$

Die Eigenbewegungsschätzung resp. Selbstlokalisierung (EME) liefert eine geschätzte 6-DOF-Pose³ des Fahrzeugs (veh), die absolut im UTM-System referenziert ist:

$${}^{\text{UTM}}\mathbf{P}_{\text{veh}}^{\text{EME}} . \quad (5.7)$$

Zusätzlich wird eine Koppelnavigation (DR, Dead-Reckoning) durchgeführt, dessen Startpose für bessere Visualisierbarkeit mit der Startpose der Eigenbewegungsschätzung initialisiert wird:

$${}^{\text{UTM}}\mathbf{P}_{\text{veh},t_0}^{\text{DR}} = {}^{\text{UTM}}\mathbf{P}_{\text{veh},t_0}^{\text{EME}} . \quad (5.8)$$

Abbildung 5.10 zeigt, dass diese beiden Posen ${}^{\text{UTM}}\mathbf{P}_{\text{veh}}^{\text{EME}}$ und ${}^{\text{UTM}}\mathbf{P}_{\text{veh}}^{\text{DR}}$ während der Fahrt auseinander „driften“ (siehe Abschnitt 5.8 zur Berechnung).

Um die Indizes in den weiteren Kapiteln kurz zu halten, wird das verwendete ebene Koordinatensystem als \mathbf{F}_0 bezeichnet, wobei gilt:

$$\mathbf{F}_0 \equiv \mathbf{F}_{\text{UTM}} . \quad (5.9)$$

5.2.3 Mitbewegte Koordinatensysteme

Körperfeste Koordinatensysteme wie das Ego-Koordinatensystem \mathbf{F}_{veh} sind fest mit dem Versuchsfahrzeug verbunden, ihr Ursprung befindet sich im Fahrzeug. Es gibt weitere mitbewegte Koordinatensysteme, die meisten von ihnen sind durch eine zeitinvariante Transformation mit dem Ego-Koordinatensystem verbunden.

²<https://proj.org/>

³Allgemein gilt hier: eine 6-DOF-Pose ${}^A\mathbf{P}_B$ beschreibt die Lage eines Koordinatensystems \mathbf{F}_B gegenüber dem Koordinatensystem \mathbf{F}_A . Liefern mehrere Algorithmen eine Pose im selben Koordinatensystem, so werden wird ein Index für die Quelle oben rechts ergänzt: ${}^A\mathbf{P}_B^{\text{Quelle}}$. Unterschiedliche Zeiten t werden unten rechts hinterlegt: ${}^A\mathbf{P}_{B,t}$. Für Details zu den 6-DOF-Posen siehe Anhang A.

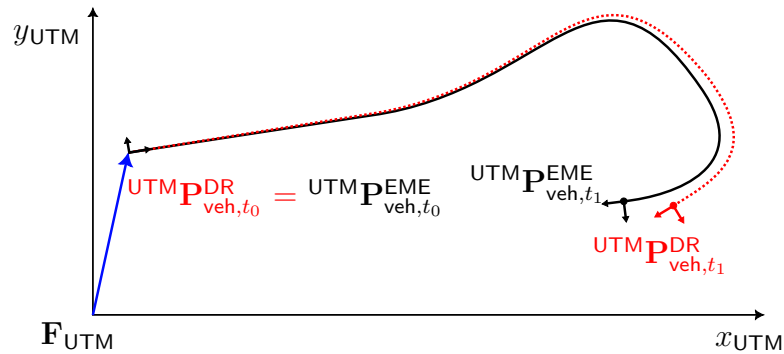


Abbildung 5.10:

Trajektorien im UTM-Koordinatensystem. Die schwarze Linie beschreibt den zurückgelegten Weg eines Fahrzeugs (veh), welcher mithilfe der Eigenbewegungsschätzung (EME) bestimmt wurde. Beginn ist zum Zeitpunkt t_0 an der Pose ${}^{\text{UTM}}\mathbf{P}_{\text{veh},t_0}^{\text{EME}}$, das Ziel wird zum Zeitpunkt t_1 an der Pose ${}^{\text{UTM}}\mathbf{P}_{\text{veh},t_1}^{\text{EME}}$ erreicht. Die rote Linie beschreibt denselben Weg, der allerdings mithilfe eines Koppelnavigationsverfahrens (DR, Dead-Reckoning) bestimmt wurde und daher über die Zeit wegdriftet. Für eine bessere Visualisierung wurde die Koppelnavigation zu Beginn initialisiert: ${}^{\text{UTM}}\mathbf{P}_{\text{veh},t_0}^{\text{DR}} = {}^{\text{UTM}}\mathbf{P}_{\text{veh},t_0}^{\text{EME}}$.

5.2.3.1 Fahrzeugkoordinatensysteme

Für die Ausrichtung von Fahrzeugkoordinatensystemen gibt es die in Abb. 5.11 gezeigten zwei gängigen Varianten, Landfahrzeug- und Luftfahrzeug-Koordinaten.

Landfahrzeug-Fahrzeugkoordinatensystem Die Verwendung von Fahrzeugkoordinaten nach DIN ISO 8855 [DIN Deutsches Institut für Normung e.V., 2013] bzw. der Vorgängernorm DIN 70000 [DIN Deutsches Institut für Normung e.V., 1994] ist in der Automobilindustrie üblich und auch in der Robotik stark vertreten, beispielsweise in Anteilen von ROS (Robot Operating System). Sie kommen daher auch für diese Arbeit und generell bei TAS zum Einsatz.

Die in Abb. 5.11a dargestellten Achsen zeigen in Fahrzeuglängsrichtung (x_{veh}), nach links (y_{veh}) sowie nach oben (z_{veh}).

Die 6-DOF-Pose ${}^{\text{UTM}}\mathbf{P}_{\text{veh}}$ beschreibt die Lage des Ego-Koordinatensystem \mathbf{F}_{veh} im UTM-Bezugssystem, welches hier als ENU-System (Osten-Norden-Oben, engl. *East-North-Up*) zu sehen ist: Der Gierwinkel ${}^{\text{UTM}}\Psi_{\text{veh}}$ des Fahrzeugs beschreibt die Rotation um die aus der Erdebene nach oben zeigende Hochachse z_{UTM} , der Winkel ${}^{\text{UTM}}\Psi_{\text{veh}} = 0$ beschreibt eine Fahrzeugausrichtung nach Osten.

Luftfahrzeug-Koordinatensystem Die in Abb. 5.11b dargestellten Achsen zeigen in Fahrzeuglängsrichtung (x_{aero}), nach rechts (y_{aero}) sowie nach unten (z_{aero}).

Die 6-DOF-Pose ${}^{\text{UTM}}\mathbf{P}_{\text{aero}}$ beschreibt wiederum die Lage des Ego-Koordinatensystem \mathbf{F}_{aero} im UTM-Bezugssystem, welches hier allerdings als NED-System (Norden-Osten-Unten, engl. *North-East-Down*) zu sehen ist: Der Gierwinkel ${}^{\text{UTM-NED}}\Psi_{\text{aero}}$ des Fahrzeugs beschreibt die Rotation um die in die Erdebene hinein zeigende Achse

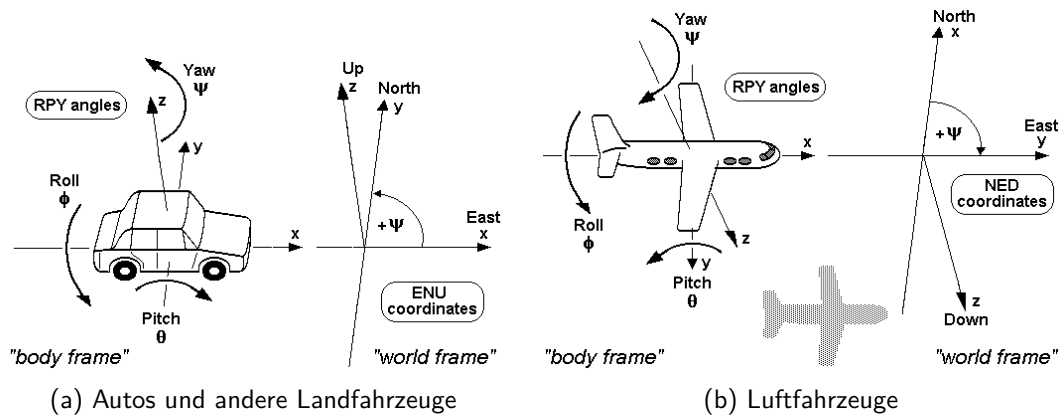


Abbildung 5.11:

Unterschiedliche Fahrzeugkoordinatensysteme. Bildquelle: <https://de-academic.com/dic.nsf/dewiki/1196041>

Tabelle 5.2:

Lage der Ego-Koordinatensysteme in den verschiedenen Versuchsfahrzeugen

Fahrzeug	Variante	Lage von \mathbf{F}_{veh}
MuCAR-3	Land	im (einmalig ermittelten) Fahrzeugschwerpunkt: in x - und y -Richtung mittig zwischen den Achsen bzw. Rädern, in z -Richtung 0,64 m über dem Boden (zzgl. Auswirkungen der Luftfederung)
MuCAR-4	Land	analog zu MuCAR-3
MuCAR-5	Land	analog zu MuCAR-3, ca. 0,25 m nach hinten verschoben (temporär am Montageort des INS)
TULF	Land/Luft	in x - und y -Richtung in der Mitte der Tandem-Hinterachse, z am Boden
StrAsRob	Land/Luft	analog zu TULF
StreetScooter	Land	in x - und y -Richtung in der Mitte der Hinterachse, z am Boden

z_{UTM} , der Winkel ${}^{UTM-NED}\Psi_{aero} = 0$ beschreibt eine Fahrzeugausrichtung nach Norden.

Ego-Koordinatensystem Das Koordinatensystem des Ego-Fahrzeugs/Roboters wird hier als Ego-Koordinatensystem \mathbf{F}_{veh} bezeichnet. Es ist ein körperfestes kartesisches Koordinatensystem mit – in Bezug zum Fahrzeug – fest ausgerichteten Achsen.

Im Regelfall wird für \mathbf{F}_{veh} ein Punkt gewählt, der sich für Fahrdynamikanwendungen gut eignet (Hinterachs-Mittelpunkt oder Schwerpunkt) oder der für Messungen gut zugänglich ist. Tabelle 5.2 stellt die Ausrichtungen und Lagen von \mathbf{F}_{veh} für

die Versuchsfahrzeuge vor. Es ist anzumerken, dass diese Wahl zum Teil historisch bedingt ist⁴.

\mathbf{F}_{veh} in der Ausrichtung *Landfahrzeug-Koordinaten* ist das für viele der in Abschnitt 7.3 vorstellten Wahrnehmungsprozesse verwendete Koordinatensystem, sowohl für die als Eingang benötigten Eigenbewegungsdaten als auch für die nach außen gelieferten Schätzergebnisse.

Für die Versuchsfahrzeuge StrAsRob und TULF wurden die an den Schnittstellen verwendeten Koordinatensysteme zu Beginn vom Projektpartner Diehl BGT Defence GmbH & Co. KG (DBD) definiert, der durch seine sonstigen Geschäftsfelder primär das Luftfahrzeug-Koordinatensystem nutzt. Die in Abschnitt 4.1.4 beschriebenen Schnittstellenmodule behandeln die notwendigen Transformationen. Zum Projektende wurde hier auch auf Landfahrzeug-Koordinaten umgestellt.

Schwerpunktkoordinatensystem Das Schwerpunktkoordinatensystem \mathbf{F}_{CoG} ist achsenparallel zu \mathbf{F}_{veh} . Der Ursprung befindet sich im ausgemessenen oder geschätzten Fahrzeugschwerpunkt: in x -Richtung zwischen den Fahrzeugachsen, in y -Richtung im Regelfall mittig zwischen linkem und rechtem Rad, und in z -Richtung oberhalb des Bodens innerhalb des Fahrzeugchassis. Die Transformation⁵ ${}^{\text{CoG}}\mathbf{H}_{\text{veh}}$ beschränkt sich damit auf eine reine Translation und wird hier vereinfachend als zeitinvariant angenommen.

Hinterachskoordinatensystem Das Hinterachskoordinatensystem \mathbf{F}_h ist ebenfalls achsenparallel zu \mathbf{F}_{veh} . Der Ursprung befindet sich im Hinterachsmittelpunkt, die Transformation ${}^h\mathbf{H}_{\text{veh}}$ beschränkt sich damit auf eine reine Translation und ist zeitinvariant.

Koordinatensysteme in der Fahrbahnebene Der zusätzliche Index $(\cdot)_g$ bei den Koordinatensystemen $\mathbf{F}_{\text{veh}g}$, $\mathbf{F}_{\text{CoG}g}$ sowie \mathbf{F}_{hg} beschreibt achsenparallele Varianten der oben erläuterten körperfesten Koordinatensysteme, bei denen $\mathbf{F}_{(\cdot),g}$ seinen Ursprung in der Fahrbahnebene senkrecht unterhalb von $\mathbf{F}_{(\cdot),g}$ hat. Die Transformation ${}^{(\cdot),g}\mathbf{H}_{(\cdot),g}$ kann bei einigen Fahrzeugen zeitvariant sein, z. B. bei einem Fahrwerk mit Niveaugulierung und der dort integrierten Sensorik. Eine durch unterschiedliche Beladung des Fahrzeugs verursachte Einfederung, die nicht durch die Sensoren erfasst wird, wird dagegen vernachlässigt.

⁴Für MuCAR-3 und MuCAR-4 wurde \mathbf{F}_{veh} in den (einmalig abgeschätzten) Schwerpunkt gelegt, da dies nach Literaturrecherche eines Kollegen am besten zu den Fahrzeugmodellen in der Regelung passen würde, und aufgrund von langfristiger Messdatenkompatibilität für diese Fahrzeuge nie verändert. Um Transformationen der INS-basierten Lokalisierung mit zu großem Hebelarm zu vermeiden, wurde dieser Sensor so nah wie möglich zu \mathbf{F}_{veh} montiert. Dass dieser Punkt „frei im Raum schwebt“ bzw. sich innerhalb des Mittelunnels befindet, hat zum Nachteil, dass extrinsische Sensorkalibrierungen nicht intuitiv verifiziert werden können – dies ist bei einem Punkt wie dem Hinterachsmittelpunkt oder einem Punkt auf der Karosserie einfacher.

⁵Übergänge zwischen zwei kartesischen Koordinatensystemen werden mit Homogenen Transformationsmatrizen beschrieben. Eine Erläuterung hierzu ist in Anhang A zu finden.

Eigenbewegungskoordinatensystem Das Eigenbewegungskoordinatensystem F_{EME} ist das interne Koordinatensystem der Eigenbewegungsschätzung. Seine Lage wird in Abhängigkeit des verwendeten Prozessmodells definiert und in den entsprechenden Abschnitten 5.5.2 und 5.5.3 erläutert. Oft ist es identisch zu F_{hg} oder F_{CoGg} . F_{EME} ist immer achsenparallel zum Ego-Koordinatensystem F_{veh} .

5.2.3.2 Sensorkoordinatensysteme

Sensorkoordinatensystem Jeder Sensor i am Fahrzeug besitzt ein spezifisches Sensorkoordinatensystem $F_{Sensor,i}$. Dieses ist entsprechend seiner Einbaulage über eine – im Regelfall zeitinvariante – Transformation mit dem Ego-Koordinatensystem F_{veh} verbunden.

Eine Ausnahme bei den institutseigenen Versuchsfahrzeugen MuCAR-3 und MuCAR-4 bilden einige Kamerasensoren, die auf der beweglichen Kameraplattform MarVEye-8 montiert sind, für diese ist die Transformation zeitvariant und aus der Positionssensorik der Kameraplattform bekannt. Bei den Versuchsfahrzeugen TULF und StrAsRob sind einige Sensoren an der Fahrerkabine montiert (Kameras, IMU, GPS in der „Sensorbay“, beim TULF zusätzlich noch die Velodyne HDL-32E-Sensoren). Die Fahrerkabine ist beweglich gegenüber dem Fahrzeugchassis gelagert, bei der Fahrt wird dieses vor allem um die y -Achse zu Schwingungen angeregt. Damit ist auch hier die Transformation zeitvariant, kann aber mangels Sensoren zur Relativlagebestimmung nicht gemessen werden und wird daher bislang vernachlässigt.

Je nach Messprinzip des Sensors kann es sich beim Sensorkoordinatensystem um ein kartesisches Koordinatensystem $F_{Sensor,i}$, ein zylindrisches Koordinatensystem $Z_{Sensor,i}$, ein Kugelkoordinatensystem $K_{Sensor,i}$ oder ein abbildendes Koordinatensystem handeln.

F_{Sensor} : Kartesische Koordinaten im Sensorursprung Die meisten Sensoren der Fahrzeugplattform und des DbW-Systems (engl. *Drive-by-Wire*) liefern ihre Daten ebenfalls in einem F_{Sensor} , z. B. Einzelradgeschwindigkeiten oder Lenkwinkel. Auch eine IMU liefert Messungen in Bezug auf ein kartesisches Koordinatensystem. Radar-Sensoren (Radio Detection And Ranging) liefern die Objektliste, d. h. die sensorintern getrackten Objekte, in kartesischen Sensorkoordinaten.

Z_{Sensor} : Polare bzw. zylindrische Koordinaten im Sensorursprung Ältere Radar-Sensoren liefern die Targetliste, d. h. die nur vorverarbeiteten aber nicht getrackten Objekte, in ebenen Polarkoordinaten.

K_{Sensor} : Kugelkoordinaten im Sensorursprung LiDAR-Sensoren wie der Velodyne HDL-64E S1 oder der Ibeo LUX 8L liefern die 3D-Punkte implizit in Kugelkoordinaten: Pro 3D-Punkt ist eine Entfernung, der horizontale Winkel des Sensorkopfs bzw. Spiegels sowie die Laser-ID gegeben. Die Punktwolke wird anschließend mithilfe der

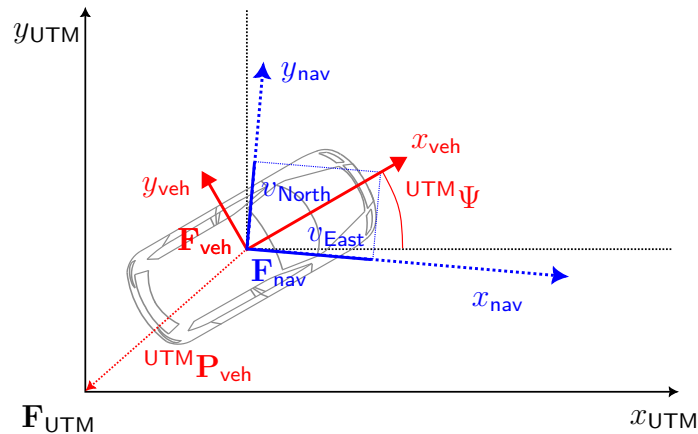


Abbildung 5.12:

Die 6-DOF-Pose ${}^{\text{UTM}}\mathbf{P}_{\text{veh}}$ beschreibt die Lage des Ego-Koordinatensystem \mathbf{F}_{veh} im UTM-Bezugssystem \mathbf{F}_{UTM} . Ein GNSS-Empfänger liefert die zugehörige Geschwindigkeitsmessung im Navigationskoordinatensystem \mathbf{F}_{nav} , welches standortabhängig ggü. \mathbf{F}_{UTM} verdreht ist.

bekanntem intrinsischen Sensorkalibrierung von $\mathbf{K}_{\text{Sensor}}$ in das $\mathbf{F}_{\text{Sensor}}$ transformiert. Neuere Radar-Sensoren können auch vertikal auflösen und liefern die Targetliste in Kugelkoordinaten.

▮: **Bildkoordinaten** Die Kamerasensoren liefern ihre Bilder in Bildkoordinaten. Für die Projektion der 3D-Welt auf die Sensorebene werden sowohl die extrinsische Kalibrierung ${}^{\text{veh}}\mathbf{P}_{\text{Sensor}}$ als auch die intrinsische Kalibrierung, die die Objektiv- und Bildsensoreigenschaften modellieren, verwendet.

5.2.4 Navigationskoordinatensystem

Das Navigationskoordinatensystem verknüpft die erdfesten Koordinatensysteme aus Abschnitt 5.2.2.2 (z. B. \mathbf{Y}_{LLH}) mit den mitbewegten Koordinatensystemen aus Abschnitt 5.2.3.

Das Navigationskoordinatensystem \mathbf{F}_{nav} hat seinen Ursprung an derselben Position wie das körperfeste Koordinatensystem \mathbf{F}_{veh} . Die x_{nav} - und y_{nav} -Achsen dieses Koordinatensystems liegen in der Tangentialebene an den Referenzellipsoid, wie in Abb. 5.7 gezeigt.

Es existieren drei häufig verwendete Varianten:

NED Die x_{nav} -Achse zeigt nach Norden, y_{nav} nach Osten und z_{nav} in Richtung des Schwerevektors nach unten.

ENU Die x_{nav} -Achse zeigt nach Osten, y_{nav} nach Norden und z_{nav} entgegen des Schwerevektors nach oben.

NWU (Norden-Westen-Oben, engl. *North-West-Up*) Die x_{nav} -Achse zeigt ebenfalls nach Norden, y_{nav} nach Westen und z_{nav} entgegen des Schwerevektors nach oben.

Verwendung GNSS-Empfänger liefern die gemessene Geschwindigkeit üblicherweise im Navigationskoordinatensystem \mathbf{F}_{nav} und daher achsenparallel zu den Längen- und Breitengraden. Hierbei ist zu beachten, dass die Achsen des kartesischen \mathbf{F}_{UTM} und des elliptischen \mathbf{Y}_{LLH} – außer in wenigen Konstellationen – trotz ähnlicher Bezeichnung „Nord“ und „Ost“ nicht parallel sind, wie in Abb. 5.9 am Winkelversatz zwischen der gestrichelten Linie $\lambda = 12^\circ$ und der roten Linie bei 700 zu erkennen ist. Daher muss der Geschwindigkeitsvektor eines GNSS-Empfängers ortsabhängig unterschiedlich stark rotiert werden, wie in Abb. 5.12 gezeigt wird.

5.2.5 Anmerkung

Eine korrekte Handhabung aller Koordinatensysteme ist äußerst wichtig für eine gute Performance der Algorithmen. Gerade bei mehreren beteiligten Projektpartnern ist auf eine saubere Definition und deren Einhaltung zu achten, unterschiedliche Bezugssysteme sollten möglichst vermieden werden.

Weiterhin lässt sich festhalten, dass Messungen möglichst im Sensorkoordinatensystem kommuniziert werden sollten, da sich die Sensoreigenschaften hier sehr gut in Form von Varianzen modellieren lassen. Am Beispiel Radar wären dies Polarkoordinaten, mit kleiner Varianz in der Entfernung und großer Varianz im Winkel. Führt man sehr früh in der Kommunikationskette eine Transformation auf kartesische Koordinaten durch, so ist danach keine einfache Modellierung mehr möglich.

5.3 Stand der Technik

Aus der DARPA Urban Challenge folgt zum Thema Lokalisierung die wesentliche Erkenntnis, die relative Eigenbewegung von der absoluten Lokalisierung strikt zu trennen [Moore et al., 2009]. Diese beiden Lösungen unterscheiden sich in ihrem Optimierungsziel: Die relative Eigenbewegung (in [Moore et al., 2009] „local frame“ genannt) beschreibt einen sprungfreien Positionsverlauf von einem beliebigen Startpunkt aus mit dem Ziel, zwischen zwei Zeitschritten möglichst exakt zu sein. Die langfristige Position kann jedoch driftbehaftet sein. Die absolute Lokalisierung (in [Moore et al., 2009] „global frame“ genannt) hat hingegen zum Ziel, in einem ortsfesten Koordinatensystem die beste Positionslösung zum aktuellen Zeitpunkt zu finden und langfristig keine Drift aufzuweisen – dafür können kurzfristige Sprünge in der Position auftreten. (Matthaei et al. [2015, Kap. 61.2.4.1])

In dieser Arbeit wird primär von *Eigenbewegungsschätzung* gesprochen, wenn nur Messungen der eigenen Fahrzeugbewegung vorliegen, aber keine globalen Positionsmessungen. Über die Integration von Sensormessungen aus Odometriesensorik und/oder einer IMU, kann eine Trajektorie ermittelt werden – dies wird als *Dead Reckoning* oder Koppelnavigation bezeichnet. Nach [Moore et al., 2009] ist dies der „local frame“. Sensorfehler wie der Bias oder Skalierungsfehler bei Drehratensensoren oder Schlupf zwischen Rad und Untergrund können hier zu starker Drift führen, also einer Formabweichung der ermittelten Trajektorie zum real gefahrenen Weg. Für die meisten Anwendungen kommen rekursive Schätzverfahren zum Einsatz, die Modellwissen über das räumlich-zeitliche Verhalten eines dynamischen Systems mit verrauschten Messungen unterschiedlicher Sensoren kombinieren. Hier besteht die Möglichkeit, die langsam veränderlichen Sensor- und Modellfehler in den Zustandsraum aufzunehmen und über die Zeit anhand der Messungen aus heterogenen Sensoren zu beobachten, sodass sie vor der Integration kompensiert werden und sich damit die Drift verringert.

Ein gutes Beispiel hierfür sind die Automobilhersteller: Sie benötigen für die Fahrdynamikregelung und ADAS-Anwendungen (Fahrerassistenzsystem, engl. *Advanced Driver Assistance System*) eine präzise Eigenbewegungsschätzung, aber meist keine globale Lokalisierung. Daher können Dead-Reckoning-Verfahren zum Einsatz kommen, die die Messungen aus der Seriensensorik (Einzelrad-Odometrie, IMU, Lenkwinkel, aktuelles Antriebsmoment, ...) mit extrem detaillierten Fahrzeugmodellen kombinieren, um die Trajektorie der eigenen Fahrzeugposes seit Fahrzeugstart zu bestimmen. Dabei werden viele Fehler wie Kreisel-Bias oder -Skalierung, veränderliche Raddurchmesser durch Abrieb oder Schlupf zwischen Reifen und Fahrbahn mitgeschätzt – damit wird eine sehr driftarme Trajektorie erzielt. Das sog. Zero Velocity Update kann verwendet werden, um einige Sensorfehler sehr schnell zu schätzen, benötigt aber eine externe Stillstandserkennung – dies ist beim Einsatz im Fahrzeug leicht über die Odometrie möglich. Die Verfahren der Automobilhersteller sind allerdings primär für die befestigte Straße entwickelt worden.

Werden zusätzlich globale Messungen integriert, bspw. aus einem GNSS-System oder einem kartenbasierten Verfahren, so handelt es sich um eine *Lokalisierung*, nach [Moore et al., 2009] der „global frame“. Wenn die Messungen in einem kartesischen System erfolgen, wie beispielsweise bei Verwendung von Karten oder Landmarken zur Lokalisierung, bietet es sich an, die 6-DOF-Pose in den Zustandsvektor des Schätzverfahrens aufzunehmen. Die Prädiktion der Pose zum Zeitpunkt der nächsten globalen Messung erfolgt hier anhand des dynamischen Modells und der geschätzten dynamischen Zustände sowie mithilfe der höherfrequenten Messungen (Odometrie, IMU).

Ein INS nutzt im Normalfall GNSS-basierte Positionsmessungen zur Stützung der globalen Navigation. Aufgrund der unterschiedlichen Einsatzbereiche ein- und desselben INS kann nur wenig Modellwissen angenommen werden (also bspw. kein Einspurmodell wie bei Fahrzeugen, sondern nur ein Punktmassmodell), dafür erfolgt eine äußerst präzise Schätzung der Sensorfehler zur Driftkompensation (Bias- und Skalierungsfehler für Kreisel und Bias für Beschleunigungssensoren). Bei INS kommt

meist ein Error-State-Space-Modell zum Einsatz [Wendel, 2011, Kap. 8.2], bei dem nur die Positions-, Geschwindigkeits-, Lage- und Sensorfehler Teil des Zustandsraums sind und über der Zeit geschätzt werden. Die Messgrößen der IMU werden als Eingangsgrößen betrachtet und zur Integration der Position, Geschwindigkeit und Lage im Koordinatensystem \mathbf{F}_{ECEF} genutzt, die globalen Positionsmessungen aus dem GNSS dienen nur zur Beobachtung der Fehler. Für die Ausgabe werden die Daten von \mathbf{F}_{ECEF} in ellipsoidische Koordinaten \mathbf{Y}_{LLH} für die Position, in das Navigationskoordinatensystem \mathbf{F}_{nav} für die Geschwindigkeiten und in das Sensorkoordinatensystem \mathbf{F}_{Sensor} für die Lagewinkel und Winkelgeschwindigkeiten transformiert. Im Detail muss bei INS noch zwischen *Loosely Coupling*, also dem Einfiltern der Positionslösung eines als Black-Box betrachteten GNSS-Empfängers, *Tightly Coupling*, bei dem die Rohmessungen des GNSS-Empfängers genutzt werden und für diesen zusätzliche Parameter mitgeschätzt werden, und *Deep Integration*, bei dem eine Rückführung aus dem Filter in den GNSS-Empfänger erfolgt, unterschieden werden [Steinhardt und Leinen, 2015, Kap. 26.3.1].

In vielen Anwendungen, vor allem in Closed-Loop-Ansätzen autonomer Fahrzeuge, wird eine dauerhafte Bereitstellung einer globalen Position benötigt, auch bei GNSS-Ausfällen. Neben Lehrbüchern wie [Groves, 2013] gibt auch [Duník et al., 2020] einen guten Überblick über unterschiedlichste Filteransätze für Navigations- und Lokalisierungszwecke je nach Umgebung und Art der stützenden Messungen.

[Dicu et al., 2018] beschreibt ein sehr einfaches System, welches unter Verwendung von Serien-Sensorik im PKW ein Dead-Reckoning durch Integration von Gierrate und Geschwindigkeit durchführt und damit bei GNSS-Ausfall weiterhin eine – mit wachsendem Fehler behaftete – Position bereitstellen kann. Hier werden keine Schätzverfahren für die Bestimmung von dynamischen oder statischen Zuständen verwendet, dementsprechend wirken sich Bias- und andere Sensorfehler stark aus. GNSS-Messungen kommen nur zur Richtungsbestimmung und zum Reset des Dead-Reckonings zum Einsatz.

Bevly und Parkinson [2007] beschäftigen sich mit Dead-Reckoning-Navigation für einen Traktor und damit Offroad-Einsatz. Sie trennen die Filter auf: einen hochfrequenten für die Fehlerschätzung von IMU und Odometrie (auf Basis eines Übergrund-Geschwindigkeitssensors), einen niederfrequenten zur Integration von GNSS-Positionsmessungen, und eine dritten für die Integration von fahrzeugabhängigen Messungen wie dem Lenkwinkel. Die Autoren modellieren ihren Schätzer in Inertialkoordinaten $\mathbf{F}_{inertial}$, schlagen aber die Untersuchung für eine Schätzung im Fahrzeugkoordinatensystem \mathbf{F}_{veh} vor.

Zhang et al. [2016] nutzen ähnliche Sensoren wie am TULF und ebenfalls separate Filter zur Schätzung von Geschwindigkeit und Heading, bevor die Ergebnisse mit den GNSS-Positionsmessungen in einem dritten Filter fusioniert werden. Für die Odometrie werden vorab drei Kalibrierungen für normale Fahrt, Beschleunigen und Verzögern erstellt, um unterschiedlichen Schlupf zu berücksichtigen.

Neben GNSS können auch weitere externe Quellen zur Stützung verwendet werden. Schütz et al. [2020] nutzen beispielsweise ein Loosely-Coupled-Verfahren zur Loka-

lisierung mittels IMU und GNSS, verwenden GNSS-Messungen aber nur bei einer durch RTK-Korrekturdaten (Echtzeitkinematik, engl. *Real Time Kinematic*) erzielten Positionsgenauigkeit im Zentimeter-Bereich. Die entscheidende Geschwindigkeits- und damit Positionsstützung erfolgt durch aus LiDAR-Messungen gewonnene Relativposen: aus detektierten Objekten wird eine grobe Relativpose bestimmt und anschließend mittels ICP verfeinert.

Je nach Anforderungsprofil kommen in der Eigenbewegungsschätzung und Lokalisierung unterschiedliche Zustandsräume zum Einsatz: Für einige Anwendungen (z. B. Flurförderfahrzeuge in einer Industriehalle) genügt die Lokalisierung in einer 2D-Ebene mithilfe einer 3-DOF-Pose (x, y, Ψ) , während für andere Anwendungen (z. B. autonome Geländefahrzeuge) im 3D-Raum eine vollständige 6-DOF-Pose $(x, y, z, \Phi, \Theta, \Psi)$ erforderlich ist. Die unterschiedlichen Verfahren zur Eigenbewegungsschätzung und Lokalisierung können im Allgemeinen für die Anwendung im 2D- und 3D-Raum angepasst werden.

5.4 Schätzverfahren

Rekursive Schätzverfahren sind mathematische Verfahren zur optimalen Schätzung eines Zustandsvektors \mathbf{x} auf der Basis von fehlerhaften, verrauschten Messungen \mathbf{y} . Zusätzlich kommt ein räumlich-zeitliches Dynamikmodell $f(\cdot)$ des Systems zum Einsatz, welches auch die Schätzung von nicht direkt messbaren Größen ermöglicht.

Das allgemeine System- oder Prozessmodell beschreibt die nichtlineare Transformation des Zustandsvektors \mathbf{x}_{k-1} , des Systemeingangs \mathbf{u}_{k-1} und des Störvektors \mathbf{v}_{k-1} mithilfe der Systemgleichung $f_{k-1}(\cdot)$:

$$\mathbf{x}_k = f_{k-1}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{v}_{k-1}) . \quad (5.10)$$

Das Messmodell beschreibt die nichtlineare Abbildung des Zustandsvektors \mathbf{x}_k , des Systemeingangs \mathbf{u}_k und des Messrauschvektors \mathbf{w} mithilfe der Messgleichung $g_k(\cdot)$:

$$\mathbf{y}_k = g_k(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k) . \quad (5.11)$$

Für die meisten Systeme existiert kein Durchgangsanteil, dann vereinfacht sich Gleichung (5.11) zu

$$\mathbf{y}_k = g_k(\mathbf{x}_k, \mathbf{w}_k) . \quad (5.12)$$

Tabelle 5.3 gibt eine Übersicht über unterschiedliche Schätzverfahren zur Datenfusion. In diesem Kapitel liegt der Schwerpunkt auf dem Kalman-Filter (KF) und seinen Derivaten, die in den Abschnitte 5.4.1 bis 5.4.3 kurz vorgestellt werden. In den im weiteren Verlauf dieser Arbeit verwendeten Wahrnehmungsmodulen (vgl. Abschnitt 7.3), die zum großen Teil von Kollegen am Institut entwickelt wurden, findet auch das Partikel-Filter (PF) Verwendung.

Tabelle 5.3:

Überblick über unterschiedliche Schätzverfahren zur Datenfusion. Bildquelle: [Steinhardt und Leinen, 2015, S. 488]

Bayes-Filter	linear parametrisch	<ul style="list-style-type: none"> – nur eingeschränkt in digitalen Systemen umsetzbar – keine kontinuierlichen Ausgabegrößen
Kalman-Filter (KF)	linear parametrisch	<ul style="list-style-type: none"> – Annahme: System und Messungen normalverteilt (Gaußsches weißes Rauschen) – Unsicherheitspropagation (Varianz und Mittelwert) – bei Gaußschem weißem Rauschen optimaler Schätzer (erreicht Erwartungstreue und minimale Varianz) – geringer Rechenaufwand – Varianzen haben quadrierte Einheit der Zustandsgrößen
Informationsfilter (IF)	linear parametrisch	<ul style="list-style-type: none"> – wie Kalman-Filter, aber Propagation der inversen Varianz-Kovarianz-Matrix – Zustand unendlicher (Anfangs-)Unsicherheit damit numerisch darstellbar – signifikant erhöhter Rechenaufwand gegenüber dem KF, wenn Varianzen der Zustände benötigt werden
Extended KF/IF	linearisiert parametrisch	<ul style="list-style-type: none"> – wie KF oder IF, jedoch: – durch Linearisierung sind auch nichtlineare Zusammenhänge modellierbar – Verlust der Optimalität (Erwartungstreue und minimale Varianz) durch Linearisierung – moderat erhöhter Rechenaufwand
(Extended) Error-State-Space-KF	linearisiert parametrisch	<ul style="list-style-type: none"> – wie (Extended) KF, jedoch: – Schätzung von Fehlerinkrementen statt vollen Zuständen – Prädiktionsschritt durch Messung eines Basissystems – Erwartungswert der Fehlerinkremente ist null, daher kleiner Linearisierungsfehler – Rechenaufwand leicht reduziert gegenüber (Extended) KF
Unscented KF	nicht-linearisiert parametrisch	<ul style="list-style-type: none"> – Unscented Transformation: nichtlineare Fortpflanzung von aus der Normalverteilung entnommenen Punkten, anschließend Mittelwert und Varianz aus Transformationsergebnis berechnet – Vorteil gegenüber EKF bei ausgeprägter Nichtlinearität – Rechenaufwand vergleichbar mit EKF
Histogramm-Filter	nicht-parametrisch	<ul style="list-style-type: none"> – keine Normalverteilung für Ein- und Ausgang für Optimalität vorausgesetzt – Zustandsraum in endlich viele Regionen zerlegt (statisch oder dynamisch) – nichtlineare Propagation der Stützstellen – hoher, für Echtzeitanwendungen kritischer Rechenaufwand
Partikelfilter	nicht-parametrisch	<ul style="list-style-type: none"> – zufällige Abtastpunkte der Eingangsverteilung nichtlinear auf Ausgang projiziert – einfacher und in Bezug auf die nichtlineare Systemgleichung flexibler Algorithmus – Anzahl, Dichte und Varianz der zufälligen Abtastpunkte nur experimentell zu bestimmen – hoher, für Echtzeitanwendungen kritischer Rechenaufwand

Auf eine Herleitung der Filter wird hier verzichtet, sondern auf die Literatur [Kalman, 1960] sowie Lehrbücher verwiesen [Maybeck, 1979, Bar-Shalom et al., 2001].

5.4.1 Kalman-Filter

Das Kalman-Filter wurde von Kalman [1960] vorgestellt, Anteil daran haben auch Bucy und Stratonovich. Bei linearen dynamischen Systemen und linearen Abbildungen ist dieses Filter geeignet, auch nicht direkt messbare Zustände optimal im Sinne der kleinsten Fehlerquadrate zu schätzen.

Für lineare Systeme $f(\cdot)$ und $g(\cdot)$ ergeben sich aus Gleichungen (5.10) und (5.12) für die mit weißem Rauschen im Störvektor \mathbf{v} und Messrauschvektor \mathbf{w} behaftete reale Strecke:

$$\mathbf{x}_k = \Phi_{k-1}\mathbf{x}_{k-1} + \mathbf{B}_{k-1}\mathbf{u}_{k-1} + \mathbf{D}_{k-1}\mathbf{v}_{k-1}, \quad (5.13)$$

$$\mathbf{y}_k = \mathbf{C}_k\mathbf{x}_k + \mathbf{w}_k. \quad (5.14)$$

Φ_{k-1} wird als Transitionsmatrix bezeichnet, \mathbf{B}_{k-1} ist die Eingangsmatrix und \mathbf{D}_{k-1} die Störgrößeneingangsmatrix. \mathbf{C}_k ist die Ausgangsmatrix oder Messmatrix.

Beim Kalman-Filter handelt sich um einen rekursiven Algorithmus, in dem abwechselnd die Schritte *Prädiktion* und *Innovation* erfolgen.

In der Prädiktion auf den Zeitpunkt der nächsten Messung k erfolgt die Vorhersage für den Zustand \mathbf{x}_k^* und die Schätzfehlerkovarianz \mathbf{P}_k^* :

$$\mathbf{x}_k^* = \Phi_{k-1}\hat{\mathbf{x}}_{k-1} + \mathbf{B}_{k-1}\mathbf{u}_{k-1}, \quad (5.15)$$

$$\mathbf{P}_k^* = \Phi_{k-1}\mathbf{P}_{k-1}\Phi_{k-1}^\top + \mathbf{D}_{k-1}\mathbf{Q}_{k-1}\mathbf{D}_{k-1}^\top, \quad (5.16)$$

$$\mathbf{K}_k = \mathbf{P}_k^* \mathbf{C}_k^\top \underbrace{(\mathbf{C}_k \mathbf{P}_k^* \mathbf{C}_k^\top + \mathbf{R}_k)^{-1}}_{\mathbf{S}_k}. \quad (5.17)$$

Die Matrix \mathbf{Q}_{k-1} beschreibt die Kovarianz des Systemrauschvektors \mathbf{v} . Die Matrix \mathbf{R}_k beschreibt die Kovarianz des Messrauschvektors \mathbf{w} .

Anschließend erfolgt die Innovation des Zustands $\hat{\mathbf{x}}_k$ und der Schätzfehlerkovarianz \mathbf{P}_k mithilfe der neuen Messung \mathbf{y}_k :

$$\hat{\mathbf{x}}_k = \mathbf{x}_k^* + \mathbf{K}_k \left(\mathbf{y}_k - \underbrace{\mathbf{C}_k \mathbf{x}_k^*}_{\mathbf{y}_k^*} \right), \quad (5.18)$$

$$\mathbf{P}_k = \mathbf{P}_k^* - \mathbf{K}_k \mathbf{C}_k \mathbf{P}_k^*. \quad (5.19)$$

Die Rückführmatrix \mathbf{K}_k wird auch als Kalman-Gain bezeichnet und gewichtet die Differenz zwischen realer Messung \mathbf{y}_k und vorhergesagter Messung \mathbf{y}_k^* . Die Innovationskovarianzmatrix \mathbf{S}_k dient zum Aussortieren von potentiell fehlerhaften Messungen,

die zu stark vom Erwartungswert abweichen. Im ersten Schritt wird der Systemzustand mit einer A-priori-Schätzung zu \mathbf{x}_0 initialisiert und dann mit der ersten Messung aktualisiert.

Das KF ist bei Systemen mit nichtlinearen System- oder Messgleichungen nicht mehr optimal.

5.4.2 Erweitertes Kalman-Filter

Eine Erweiterung des linearen KF ist das EKF (Erweitertes Kalman-Filter).

Für nichtlineare Systeme $f_{k-1}(\cdot)$ und $g_k(\cdot)$ ergeben sich aus Gleichungen (5.10) und (5.12) für die mit weißem Rauschen im Störvektor \mathbf{v}_{k-1} und Messrauschvektor \mathbf{w}_k behaftete reale Strecke:

$$\mathbf{x}_k = f_{k-1}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) + \mathbf{D}_{k-1}\mathbf{v}_{k-1}, \quad (5.20)$$

$$\mathbf{y}_k = g_k(\mathbf{x}_k) + \mathbf{w}_k. \quad (5.21)$$

Im EKF kommen für die Berechnung von prädiziertem Zustand $\mathbf{x}_k^* = f_k(\cdot)$ und prädizierter Messung $\mathbf{y}_k^* = g(\cdot)$ die exakten nichtlinearen Transformationsfunktionen zum Einsatz und ersetzen Gleichungen (5.15) und (5.18):

$$\mathbf{x}_k^* = f_{k-1}(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}), \quad (5.22)$$

$$\mathbf{P}_k^* = \mathbf{\Phi}_{k-1}\mathbf{P}_{k-1}\mathbf{\Phi}_{k-1}^\top + \mathbf{D}_{k-1}\mathbf{Q}_{k-1}\mathbf{D}_{k-1}^\top, \quad (5.23)$$

$$\mathbf{K}_k = \mathbf{P}_k^* \mathbf{C}_k^\top (\mathbf{C}_k \mathbf{P}_k^* \mathbf{C}_k^\top + \mathbf{R}_k)^{-1}, \quad (5.24)$$

$$\hat{\mathbf{x}}_k = \mathbf{x}_k^* + \mathbf{K}_k (\mathbf{y}_k - g_k(\mathbf{x}_k^*)), \quad (5.25)$$

$$\mathbf{P}_k = \mathbf{P}_k^* - \mathbf{K}_k \mathbf{C}_k \mathbf{P}_k^*. \quad (5.26)$$

Das EKF benötigt für die Berechnung von Schätzfehlerkovarianzmatrix und Kalman-Gain Ableitungen der Transformationsfunktionen $f(\cdot)$ und $g(\cdot)$ im Arbeitspunkt. Diese werden als Matrix der partiellen Ableitungen der nichtlinearen Prozess- und Messmodelle nach dem Zustandsvektor gebildet, kurz als Jacobi-Matrix bezeichnet:

$$\mathbf{\Phi}_{k-1} = \frac{\partial f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1})}{\partial \mathbf{x}_{k-1}}, \quad (5.27)$$

$$\mathbf{C}_k = \frac{\partial g(\mathbf{x}_k)}{\partial \mathbf{x}_k}. \quad (5.28)$$

Wenn eine Transformationsfunktion analytisch bekannt ist, so kann die partielle Ableitung ebenfalls analytisch bestimmt werden. Dies ist allerdings oft aufwendig und nur mit Hilfe von Tools wie Mathematica oder Matlab möglich und resultiert oft in langen Termen.

Wenn eine Transformationsfunktion analytisch nicht bekannt ist, weil in der Messung z. B. ein Funktionsaufruf aus einer Softwarebibliothek involviert ist, so kann auch die

Ableitung nicht analytisch bestimmt werden. In diesem Fall kann durch numerisches Differenzieren um den Arbeitspunkt eine Näherungslösung bestimmt werden.

5.4.3 Unscented Kalman-Filter

Eine Alternative zum EKF, bei der keine analytischen partiellen Ableitungen bekannt sein und auch keine numerischen Ableitungen berechnet werden müssen, ist das Unscented Kalman-Filter (UKF). Es handelt sich um eine Variante der Sigma-Punkt Kalman-Filter (SPKF), die in [van der Merwe, 2004] gut zusammenfasst werden.

Das UKF geht auf Julier und Uhlmann zurück, die seit Mitte der 1990er Jahre die Unscented Transform entwickelten [Julier und Uhlmann, 2004]. Mittels der Unscented Transform können Mittelwert und Varianz einer stochastischen Variable nach einer nichtlinearen Transformation approximiert werden. Hierzu werden $2n + 1$ Punkte im n -dimensionalen Zustandsraum ausgewählt und nichtlinear transformiert. Aus dem als Gauß-verteilt angenommenen Ergebnis dieser Transformation wird anschließend ein gewichteter Mittelwert und eine gewichtete Varianz berechnet.

Das UKF ist allerdings nicht in jedem Fall ein Allheilmittel: bei stark nichtlinearen Abbildungen wird die Annahme der Gauß-Verteilung nach der Transformation verletzt, sodass die Approximationen von Mittelwert und Kovarianz nicht mehr stimmen.

Sind die Voraussetzungen jedoch erfüllt, wird mit dem UKF im Vergleich zum EKF eine bessere Approximation des Zustands, vor allem aber der Kovarianz erzielt. Allerdings ist das Verfahren etwas rechenaufwändiger, hat aber noch die gleiche Komplexität $\mathcal{O}(n^3)$ wie das EKF. Der große Vorteil für den Entwickler ist der Entfall der analytischen partiellen Ableitungen – dies wird auch in der Literatur so gesehen:

Der Entwurf eines Sigma-Point-Kalman-Filters erfordert hierbei sicherlich weniger Erfahrung, was ein Vorteil ist. Mit einem an das erwartete Szenario angepassten Kalman-Filter hingegen kann in vielen Fällen mit deutlich weniger Rechenzeit die gleiche Genauigkeit und Integrität, d. h. Übereinstimmung zwischen der Kovarianzmatrix der Schätzfehler des Filters und den tatsächlich vorliegenden Schätzfehlern, erzielt werden. (Wendel [2011, S. 249])

Für diese Arbeit kam daher primär das EKF zum Einsatz.

5.4.4 Verzögerte Messungen

Kommt ein Filter in einem Verbund aus mehreren Sensoren zum Einsatz, so ist im Allgemeinen mit unterschiedlichen Zykluszeiten und Latenzen für die einzelnen Messungen zu rechnen. Im Anwendungsbeispiel der Eigenbewegungsschätzung liefern GNSS-Empfänger Messungen mit 1 bis 10 Hz und größerer Latenz, während eine IMU Messungen mit 100 Hz oder höher bei niedriger Latenz liefert. Verzögerte Messungen, engl. *Out-of-Sequence Measurements* (OOSMs) liegen immer dann vor, wenn Messungen nicht in der Reihenfolge ihrer Entstehung am Filter eintreffen, sondern

durcheinander. Diese OOSM erfordern eine besondere Behandlung im Schätzprozess, um eine Verfälschung von Mittelwert und Varianz des Schätzergebnisses zu verhindern.

An dieser Stelle sollen mehrere Ansätze kurz vorgestellt, aber nicht vertieft werden:

OOSMs verwerfen Alle verspätet eintreffenden Messungen zu verwerfen, ist im Regelfall wenig sinnvoll. Im obigen Beispiel würde dann keine GNSS-Messung genutzt, da zu jedem Zeitpunkt bereits eine aktuellere IMU-Messung genutzt wurde. Dieses Verfahren sollte nur dann zum Einsatz kommen, wenn OOSM seltenst auftreten, bspw. weil die Erzeugung einzelner Messungen länger als üblich gedauert hat oder der Transportweg die Latenz einmalig erhöht hat. In diesen Sonderfällen kann diese Messung ohne große Auswirkungen verworfen werden.

Alter der OOSM ignorieren Nur für Systeme mit sehr niedriger Dynamik geeignet ist das Verfahren, OOSMs parallel mit den aktuellen Messungen zu verarbeiten und ihr Alter zu ignorieren. Diese Art der Verarbeitung kommt jedoch oft ungewollt zum Einsatz, wenn in einem zeitgetriggerten Filter keine Latenzen berücksichtigt sondern alle Messungen ohne Unterschied eingefiltert werden. Dies führt im Regelfall zu einer gestörten Filterperformance.

Messdatenverzögerung Eine weitere Möglichkeit ist, das komplette Filter so weit in der Vergangenheit laufen zu lassen, dass jede OOSM zusammen mit den hochfrequenten Messungen normal eingefiltert werden kann. Hierzu werden alle eintreffenden Messungen erst einmal abgespeichert (das sog. Buffering). Dabei wird auf den Informationsgewinn der neueren (ungenutzten) Messungen verzichtet, auf den Jetzt-Zustand kann nur über eine ungenaue Prädiktion über einen vergleichsweise langen Zeitraum geschlossen werden. Für das Post-Processing von Daten im Labor ist dieses ein geeigneter Ansatz, im Live-Betrieb jedoch sehr kritisch zu sehen.

Brute-Force, Reprozessierung Für diesen Ansatz werden alle eintreffenden Messungen sowie die geschätzten Zustandsvektoren und Schätzfehlerkovarianzmatrizen in einem Pufferspeicher für einen definierten Zeithorizont abgelegt. Beim Eintreffen einer OOSM wird das Filter auf den Zustand vor dieser Messung zurückgesetzt, auf den Zeitpunkt der OOSM prädiziert, diese eingefiltert, und anschließend Prädiktions- und Updateschritte für alle Messungen aus dem Pufferspeicher bis zum aktuellen Zeitpunkt durchgeführt. Diese Variante ist extrem rechenaufwändig, da jede Messung viele Male verarbeitet werden muss. Sie ist jedoch eine im Sinne der Schätzfehler optimale Methode, da keine Approximationen zum Einsatz kommen.

Retrodiktion Eine andere Variante ist die sog. Retrodiktio [Bar-Shalom et al., 2002] oder Rückprojektion vom aktuellen Zustand auf den Zeitpunkt der OOSM unter Verwendung des inversen Systemmodells, das Einfiltern der OOSM und anschließend eine Ein- oder Mehr-Schritt-Prädiktio auf den aktuellen Zustand. Hier wirkt sich allerdings das Prozessrauschen gleich doppelt aus.

Vorwärts-Prädiktio mit Fusion und Dekorrelatio Ein weiterer Ansatz ist die sog. Vorwärts-Prädiktio mit Fusion und Dekorrelatio, engl. *Forward-Prediction Fusion and Decorrelation* (FPFD), die im Informationsraum arbeitet und auf eine Retrodiktio verzichtet [Rheume und Benaskeur, 2008]. Für diesen Ansatz müssen die geschätzten Zustandsvektoren und Schätzfehlerkovarianzmatrizen in einem Pufferspeicher für einen definierten Zeithorizont zwischengespeichert werden. Es erfolgen zwei Prädiktionen vom Zeitpunkt t_{OOSM} auf den aktuellen Zeitpunkt, einmal vor und einmal inklusive der Innovation mit der OOSM. Anschließend erfolgt eine Dekorrelatio der beiden prädizierten Schätzfehlerkovarianzen, um den Einfluss der OOSM zu extrahieren – dies entspricht im Informationsraum einer Subtraktion. Diese Differenz wird anschließend im Informationsraum zur aktuellen Schätzfehlerkovarianz addiert, analog dazu wird der Zustand aktualisiert. Wie die Retrodiktio ist die FPFD im Mehr-Schritt-Fall eine Approximation und nur im Ein-Schritt-Fall optimal. Eine gute Erläuterung findet sich in [Westenberger, 2014, Kapitel 3.5].

Weitere Verfahren und detaillierte Erklärungen hierzu finden sich in den Dissertationen von van der Merwe [2004, Kapitel 5.3] und Muntzinger [2011, Kapitel 4].

In dieser Arbeit kam vor allem der Brute-Force-Ansatz als optimale Methode zum Einsatz, der in diesem Einsatzzweck zur Eigenbewegungsschätzung und Lokalisierung nur vernachlässigbarem Speicher- und Rechenzeitbedarf führte.

5.5 Modellierung der Fahrzeugbewegung

Die Lenkung heutiger vorderachsgelenkter Fahrzeuge lässt sich auf den bayerischen Wagenbauer Georg Lankensperger zurückführen. Dessen Neuerung seiner Achsschenkel-Lenkung bestand darin, durch eine Lenkungskinetik das kurveninnere Rad stärker einzuschlagen als das kurvenäußere. Somit schneiden sich alle Achsverlängerungen im Kurvenmittelpunkt, dem Momentanpol MP, und alle Räder rollen gleitfrei ab. Lankensperger meldete 1816 ein Patent auf die Achsschenkel-Lenkung an⁶ [siehe Eckermann, 1998, S. 43].

⁶„Wöchentlicher Anzeiger für Kunst- und Gewerbefleiß im Königreich Baiern“, No. 24, 15. Juni 1816, Spalte 394: „dem Hofwagner Lankensberger dahier wegen Erfindung eines neuen Riebes des Vorderwagens, der keines sogenannten Scheibengestelles bedarf, und ohne Raumverlust die Wagen sehr verkürzt“. Entnommen aus <http://www.wolfgang-pfaller.de/landespa.htm>

Um einen weiterreichenden Patentschutz zu erhalten, wurde die Achsschenkel-Lenkung in England durch Rudolph Ackermann [1818] ein zweites Mal patentiert⁷. Seitdem ist die Achsschenkel-Lenkung unter dem Namen „Ackermann-Lenkung“ bekannt [vgl. Heißing et al., 2011].

Einen generellen Überblick über die Bewegungsmodelle von Radfahrzeugen geben Campion und Chung [2008].

Der folgende Abschnitt 5.5.1 legt die Grundlagen für die Bewegung auf Kreisbahnen sowie für die Bewegung von Starrkörpern. Anschließend werden Modelle für die Fahrzeugbewegung in der Ebene vorgestellt: Als Einführung dient hier die Bewegung einer Punktmasse in Abschnitt 5.5.2, danach wird in Abschnitt 5.5.3 der Schwerpunkt auf die in dieser Arbeit verwendeten Einspurmodelle gelegt. Weitere Fahrzeugmodelle werden in Abschnitt 5.5.4 kurz vorgestellt. Andere Teilbereiche der Fahrzeugdynamik werden anschließend in Abschnitt 5.5.5 separat betrachtet, beispielsweise die Roll- und Nickbewegung.

5.5.1 Grundlagen

5.5.1.1 Kreisbewegung

Die idealisierte Bewegung eines Punktes mit konstanter Geschwindigkeit v und Gier-rate $\dot{\Psi}$ beschreibt eine Kreisbahn mit dem Radius r . Innerhalb eines Abtastschrittes Δt wird damit eine Länge $s^{\text{Kreis}} = v\Delta t$ entlang eines Kreisbogens zurückgelegt. Es ergibt sich damit folgende Positionsänderung:

$$\begin{aligned}\Delta x^{\text{Kreis}} &= r \sin(\dot{\Psi}\Delta t) = \frac{v}{\dot{\Psi}} \sin(\dot{\Psi}\Delta t) , \\ \Delta y^{\text{Kreis}} &= r \left(1 - \cos(\dot{\Psi}\Delta t)\right) = \frac{v}{\dot{\Psi}} \left(1 - \cos(\dot{\Psi}\Delta t)\right) .\end{aligned}\tag{5.29}$$

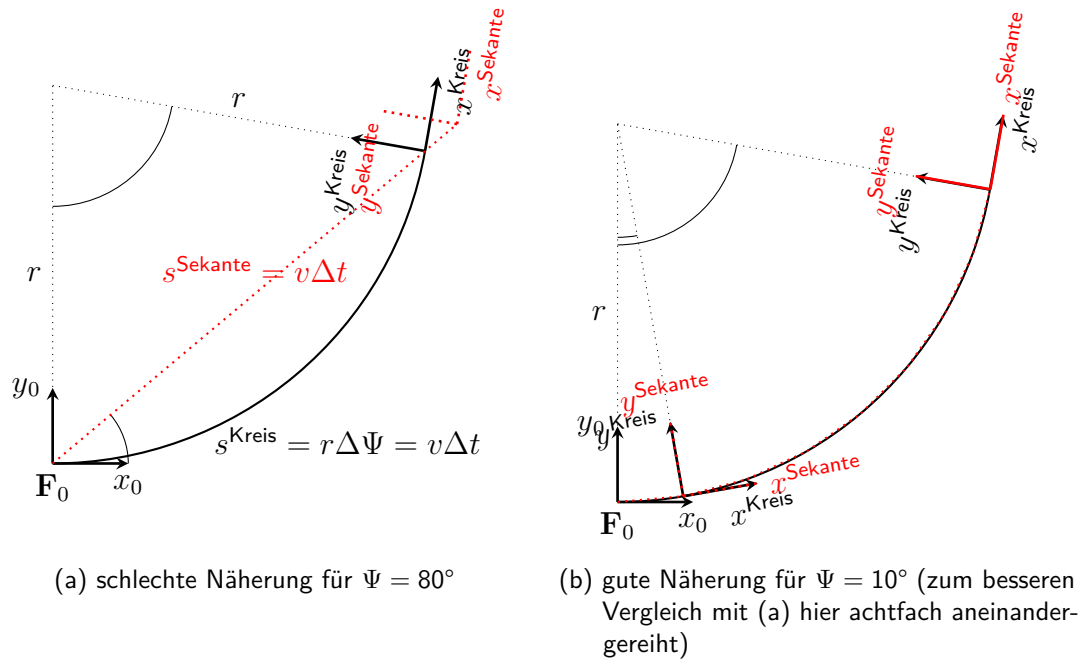
5.5.1.2 Sekante als Näherung der Kreisbewegung

Als Näherung für diese kreisförmige Bewegung kann die lineare Bewegung entlang der Sekante des Kreises gesehen werden, ebenfalls mit der Länge $s^{\text{Sekante}} = v\Delta t$. Aus der Geometrie ergibt sich damit für die Positionsänderung:

$$\begin{aligned}\Delta x^{\text{Sekante}} &= v\Delta t \cos\left(\frac{1}{2}\dot{\Psi}\Delta t\right) , \\ \Delta y^{\text{Sekante}} &= v\Delta t \sin\left(\frac{1}{2}\dot{\Psi}\Delta t\right) .\end{aligned}\tag{5.30}$$

Abbildung 5.13 stellt für zwei unterschiedlich große Winkel die Kreisbahn der Näherungslösung gegenüber.

⁷britisches Patent Nr. 4212, 1818, „Improvements on axletrees applicable to four-wheeled carriages“, mit dem Hinweis „... communicated to me by George Lenkensperger of Munich, in the Kingdom of Bavaria, ...“. Entnommen aus <http://www.wolfgang-pfaller.de/landespa.htm>

**Abbildung 5.13:**

Vergleich von echter Kreisbewegung (schwarz) und einer Näherungslösung (rot gepunktet) auf Basis der Sekante des Kreises für zwei verschiedene Rotationswinkel.

Um die Eignung diese Näherung zu überprüfen, wird im Folgenden mit den Begriffen der Messtechnik der Messfehler betrachtet. Der *absolute Fehler* F ergibt sich aus der Differenz von *angezeigtem Wert* x_a und *richtigem Wert* x_r . Der *relative Fehler* f ergibt sich aus dem Quotienten von absolutem Fehler und richtigem Wert:

$$F = x_a - x_r, \quad (5.31)$$

$$f = \frac{F}{x_r} = \frac{x_a - x_r}{x_r} = \frac{x_a}{x_r} - 1. \quad (5.32)$$

In diesem Fall wird die echte Kreisbewegung aus Gleichung (5.29) als x_r und die Annäherung durch die Sekante aus Gleichung (5.30) als x_a gewählt. Der relative Fehler ergibt sich damit für beide Richtungen wie folgt:

$$f = \begin{pmatrix} \frac{x_{\text{Sekante}}}{x_{\text{Kreis}}} - 1 \\ \frac{y_{\text{Sekante}}}{y_{\text{Kreis}}} - 1 \end{pmatrix} = \begin{pmatrix} \frac{v\Delta t \cos\left(\frac{1}{2}\dot{\Psi}\Delta t\right)}{\frac{v}{\dot{\Psi}} \sin(\dot{\Psi}\Delta t)} \\ \frac{v\Delta t \sin\left(\frac{1}{2}\dot{\Psi}\Delta t\right)}{\frac{v}{\dot{\Psi}} (1 - \cos(\dot{\Psi}\Delta t))} \end{pmatrix} \quad (5.33)$$

$$= \begin{pmatrix} 1 \\ 1 \end{pmatrix} \cdot \frac{\dot{\Psi}\Delta t}{2 \sin\left(\frac{1}{2}\dot{\Psi}\Delta t\right)} - 1. \quad (5.34)$$

Für die beiden Winkeländerungen von $\Psi = 80^\circ$ bzw. $\Psi = 10^\circ$ (vgl. Abb. 5.13a und 5.13b) ergibt sich folgender relativer Fehler:

$$\Psi = \dot{\Psi} \Delta t = 80^\circ \Rightarrow f = 8,61\%, \quad (5.35)$$

$$\Psi = \dot{\Psi} \Delta t = 10^\circ \Rightarrow f = 0,13\%. \quad (5.36)$$

Fordert man nun einen relativen Fehler von höchstens 1‰, so ergibt sich beim Auflösen von Gleichung (5.34) für das Produkt

$$\left(\dot{\Psi} \Delta t\right)_{\max, 1\text{‰}} = 0,1548 \text{ rad} = 8,873^\circ. \quad (5.37)$$

Mit einem größtmöglich angenommenen Zeitintervall von $\Delta t_{\max} = 0,5 \text{ s}$ folgt daraus eine zulässige Gierrate von

$$\dot{\Psi}_{\max, 1\text{‰}} = 17,75^\circ/\text{s}. \quad (5.38)$$

Abbildung 5.13b veranschaulicht mit der Aneinanderreihung mehrerer kleiner Transformationen, dass der Fehler gegenüber der echten Kreisbewegung bei kleinen Winkeln bzw. Zeitabschnitten wesentlich kleiner ist als bei einer großen Transformation (vgl. Abb. 5.13a). Die Annäherung der Kreisbewegung durch eine lineare Bewegung entlang der Sekante ist damit für kleine Zeitabschnitte, wie sie in einem in hochfrequentem Sensortakt ausgeführten Filteralgorithmus auftreten, von hinreichender Genauigkeit und wird im weiteren Teil dieser Arbeit verwendet.

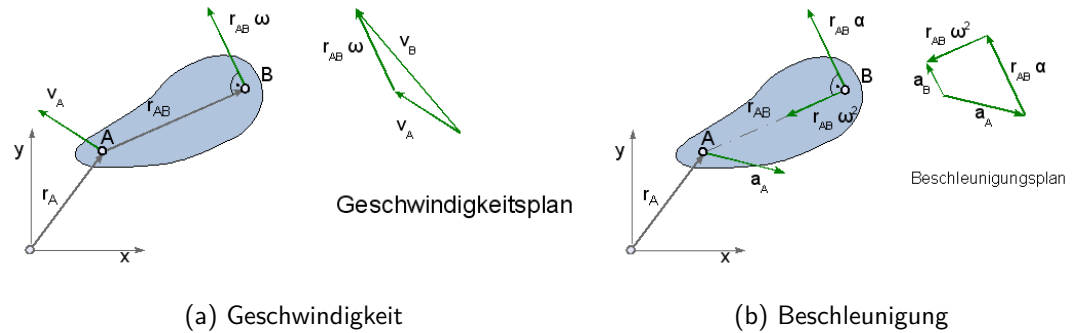
Darstellung mit HTMs Diese Bewegung lässt sich alternativ durch ${}^{k+1}\mathbf{H}_k$ als Produkt dreier Homogener Transformationsmatrizen (HTM, vgl. Anhang A.2) ausdrücken:

$${}^{k+1}\mathbf{H}_k = \mathbf{R}_{[z]} \left(\frac{\Delta\Psi}{2} \right) \cdot \mathbf{T} \left(\begin{array}{c} \Delta x \\ 0 \\ 0 \end{array} \right) \cdot \mathbf{R}_{[z]} \left(\frac{\Delta\Psi}{2} \right). \quad (5.39)$$

Zuerst erfolgt eine Drehung mit dem Winkel $\frac{\Delta\Psi}{2}$ um die z -Achse, anschließend eine Translation um $\Delta x = v\Delta t$ entlang der Sekante, und abschließend eine zweite Rotation mit dem Winkel $\frac{\Delta\Psi}{2}$ um die z -Achse. $\Delta\Psi = \dot{\Psi}\Delta t$ ist dabei die Winkeländerung während eines Abtastschrittes Δt mit konstant angenommener Gierrate $\dot{\Psi}$.

5.5.1.3 Näherung für die Bewegung im dreidimensionalen Raum

Die Beschreibung aus Gleichung (5.39) für die Bewegung in der Ebene lässt sich leicht erweitern für die approximierte Bewegung im 3D-Raum. Zuerst erfolgt eine

**Abbildung 5.14:**

Vereinfachte Darstellung der ebenen Bewegung eines starren Körpers. Bildquelle: <https://goessner.net/learn/dynamik/lec08/>

Rotation mit dem halben Betrag des Zielwinkels um alle drei Achsen, anschließend eine Translation um Δx und abschließend eine zweite Rotation:

$${}^{k+1}\mathbf{H}_k = \mathbf{R}_{[x]}\left(\frac{\Delta\Phi}{2}\right) \cdot \mathbf{R}_{[y]}\left(\frac{\Delta\Theta}{2}\right) \cdot \mathbf{R}_{[z]}\left(\frac{\Delta\Psi}{2}\right) \mathbf{T}\begin{pmatrix} \Delta x = v\Delta t \\ 0 \\ 0 \end{pmatrix} \cdot \mathbf{R}_{[x]}\left(\frac{\Delta\Phi}{2}\right) \cdot \mathbf{R}_{[y]}\left(\frac{\Delta\Theta}{2}\right) \cdot \mathbf{R}_{[z]}\left(\frac{\Delta\Psi}{2}\right) \quad (5.40)$$

5.5.1.4 Starrkörper

Das Fahrzeug wird vereinfachend als starrer Körper angenommen, d. h. es tritt keine Verwindung zwischen verschiedenen Punkten an der Karosserie auf.

Die Bewegung dieses Körpers bezüglich eines Inertialsystems setzt sich zusammen aus einer Translation aller Körperpunkte sowie einer Rotation aller Körperpunkte um eine gemeinsame Körperachse. Die rotatorische Geschwindigkeit, dargestellt als dreidimensionaler Vektor $\boldsymbol{\omega} = (\omega_{[x]} \ \omega_{[y]} \ \omega_{[z]})^T$, ist für alle Körperpunkte dieselbe.

Für die allgemeine räumliche Bewegung gelten folgende Zusammenhänge [vgl. Hauer et al., 1999, S. 111], auch als Satz von Euler bekannt, die hier mithilfe der Vektoraddition dargestellt sind:

$$\mathbf{t}^B = \mathbf{t}^A + \mathbf{t}^{AB} \quad (5.41)$$

$$\mathbf{v}^B = \mathbf{v}^A + \boldsymbol{\omega} \times \mathbf{t}^{AB} \quad (5.42)$$

$$\mathbf{a}^B = \mathbf{a}^A + \dot{\boldsymbol{\omega}} \times \mathbf{t}^{AB} + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{t}^{AB}) \quad (5.43)$$

Die Translation \mathbf{t}^B stellt hier den Vektor vom Ursprung zu einem Punkt B dar, \mathbf{t}^{AB} den Vektor von A nach B. \mathbf{v}^B bezeichnet den Geschwindigkeitsvektor von Punkt B, \mathbf{a}^B seinen Beschleunigungsvektor.

Für die Bewegung in der Ebene ist dieser Zusammenhang in Abb. 5.14 dargestellt.

5.5.2 Punktmassenmodell

Das einfachste Modell zur Beschreibung einer Bewegung in der Ebene ist die „Punktmasse mit Geschwindigkeit und Gierrate“, auch als CTRV (Konstante Gierrate und Geschwindigkeit, engl. *Constant Turn Rate and Velocity*) bezeichnet.

CTRA (Konstante Gierrate und Beschleunigung, engl. *Constant Turn Rate and Acceleration*) nimmt zusätzlich die Beschleunigung in Längsrichtung in den Zustandsvektor mit auf. Zur Erhöhung der Stabilität wird für diese Anwendung zusätzlich die Gierwinkelbeschleunigung in den Zustandsvektor aufgenommen.

Der Zustandsvektor $\mathbf{x}^{\text{Fahrzeug}}$ in Gleichung (5.44) hat die Dimension 7 und beschreibt die Gierbewegung (${}^0\ddot{\Psi}_{\text{veh}}, {}^0\dot{\Psi}_{\text{veh}}, {}^0\Psi_{\text{veh}} = \angle(x_0, x_{\text{veh}})$), die Vorwärtsbewegung im Ego-Fahrzeugkoordinatensystem (${}^{\text{veh}}a, {}^{\text{veh}}v$) sowie die Position im ebenen kartesischen Koordinatensystem (${}^0x_{\text{veh}}, {}^0y_{\text{veh}}$). Zur Vereinfachung wird der Index $(\cdot)_{\text{veh}}$ fallengelassen:

$$\mathbf{x}^{\text{Fahrzeug}} = \left({}^0\ddot{\Psi} \quad {}^0\dot{\Psi} \quad {}^0\Psi \quad {}^{\text{veh}}a \quad {}^{\text{veh}}v \quad {}^0x \quad {}^0y \right)^T. \quad (5.44)$$

Das zeitdiskrete Prozessmodell in Gleichung (5.45) beschreibt den Zustandsübergang für die Zykluszeit Δt . Es besteht dabei aus einem Modell zweiter Ordnung für die Gierbewegung (${}^0\ddot{\Psi}, {}^0\dot{\Psi}, {}^0\Psi$) sowie einem Modell erster Ordnung⁸ für die longitudinale Bewegung (${}^{\text{veh}}a, {}^{\text{veh}}v$) der Punktmasse. Die Position in kartesischen Koordinaten (${}^{\text{veh}}x, {}^{\text{veh}}y$) ergibt sich aus der Überlagerung von rotatorischer und translatorischer Bewegung:

$$\begin{aligned} \mathbf{x}_{k+1}^{*, \text{Fahrzeug}} &= \mathbf{f}^{\text{Fahrzeug}} \left(\hat{\mathbf{x}}_k^{\text{Fahrzeug}}, \mathbf{u}_k, \mathbf{q}_k \right) \\ &= \begin{pmatrix} {}^0\ddot{\Psi} \\ {}^0\dot{\Psi} \\ {}^0\Psi \\ {}^{\text{veh}}a \\ {}^{\text{veh}}v \\ {}^0x \\ {}^0y \end{pmatrix}_{k+1}^* = \begin{pmatrix} {}^0\ddot{\hat{\Psi}}_k \\ {}^0\dot{\hat{\Psi}}_k + \Delta t \cdot {}^0\ddot{\hat{\Psi}}_k \\ {}^0\hat{\Psi}_k + \Delta t \cdot {}^0\dot{\hat{\Psi}}_k + \frac{\Delta t^2}{2} \cdot {}^0\ddot{\hat{\Psi}}_k \\ {}^{\text{veh}}\hat{a}_k \\ {}^{\text{veh}}\hat{v}_k + \Delta t \cdot {}^{\text{veh}}\hat{a}_k \\ {}^0\hat{x}_k + \Delta t \cdot \cos \left({}^0\hat{\Psi}_k + \frac{\Delta t}{2} \cdot {}^0\dot{\hat{\Psi}}_k \right) \cdot {}^{\text{veh}}\hat{v}_k \\ {}^0\hat{y}_k + \Delta t \cdot \sin \left({}^0\hat{\Psi}_k + \frac{\Delta t}{2} \cdot {}^0\dot{\hat{\Psi}}_k \right) \cdot {}^{\text{veh}}\hat{v}_k \end{pmatrix}. \end{aligned} \quad (5.45)$$

In Gleichung (5.45) wird für die Prädiktion der kartesischen Koordinaten die in Abschnitt 5.5.1.2 vorgestellte Näherung für eine Kreisbewegung angenommen, sprich in diesem Teil werden Winkel- und longitudinale Beschleunigung vernachlässigt.

⁸siehe Abschnitt 5.5.5 für Modelle erster Ordnung; Modelle zweiter Ordnung ergeben sich analog.

Das Eigenbewegungskoordinatensystem \mathbf{F}_{EME} kann für Punktmassenmodelle in den Fahrzeugschwerpunkt gelegt werden – bzw. für die Bewegung in 2D in die Aufstandsebene unter dem Schwerpunkt: $\mathbf{F}_{\text{EME}} = \mathbf{F}_{\text{CoGg}}$.

Dieses Modell wird durch eine angenommene Varianz der Beschleunigung und Winkelbeschleunigung parametrisiert, die in der Kovarianzmatrix des Prozessrauschens zusammengefasst werden:

$$\mathbf{Q} = \begin{pmatrix} \sigma_{\ddot{\Psi}}^2 & 0 \\ 0 & \sigma_a^2 \end{pmatrix}.$$

Über die Störgrößeneingangsmatrix \mathbf{D} pflanzen sie sich auf die weiteren Zustände fort.

Die für die Verwendung im EKF benötigte Transitionsmatrix Φ_{k-1} ergibt sich aus Gleichung (5.45) gemäß Gleichung (5.27) durch Bilden der entsprechenden Jacobi-Matrix:

$$\Phi_{k-1} = \frac{\partial f^{\text{Fahrzeug}}(\hat{\mathbf{x}}_{k-1}^{\text{Fahrzeug}}, \mathbf{u}_{k-1}, \mathbf{q}_{k-1})}{\partial \mathbf{x}_{k-1}}. \quad (5.46)$$

Zur Vereinfachung wird im folgenden Schritt auf die Indizes verzichtet und hilfsweise $\bar{\Psi} = \hat{\Psi} + \frac{\Delta t}{2} \dot{\hat{\Psi}}$ verwendet:

$$\Phi_{k-1} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \Delta t & 1 & 0 & 0 & 0 & 0 & 0 \\ \frac{\Delta t^2}{2} & \Delta t & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \Delta t & 1 & 0 & 0 \\ 0 & -\frac{\Delta t^2}{2} \hat{v} \sin(\bar{\Psi}) & -\Delta t \hat{v} \sin(\bar{\Psi}) & 0 & \Delta t \cos(\bar{\Psi}) & 1 & 0 \\ 0 & \frac{\Delta t^2}{2} \hat{v} \cos(\bar{\Psi}) & \Delta t \hat{v} \cos(\bar{\Psi}) & 0 & \Delta t \sin(\bar{\Psi}) & 0 & 1 \end{pmatrix}. \quad (5.47)$$

5.5.3 Einspurmodelle

Die Querdynamik vorderachsgelenkter Fahrzeuge in der Ebene kann vereinfachend durch Einspurmodelle beschrieben werden. Beim kinematischen Einspurmodell ergibt sich die Bewegung allein aus den geometrischen Zusammenhängen (siehe Abschnitt 5.5.3.1), beim dynamischen Einspurmodell aus dem Kräfte- und Momentengleichgewicht (siehe Abschnitt 5.5.3.2).

Diese beiden Einspurmodelle sind für unterschiedliche Geschwindigkeitsbereiche geeignet, im Übergangsbereich findet eine sanfte Überblendung statt (siehe Abschnitt 5.5.3.3).

Das Eigenbewegungskoordinatensystem \mathbf{F}_{EME} wird für Einspurmodelle meist in die Fahrzeughinterachse gelegt – bzw. für die Bewegung in 2D in die Aufstandsebene unter dem Hinterachsmittelpunkt: $\mathbf{F}_{\text{EME}} = \mathbf{F}_{\text{hg}}$.

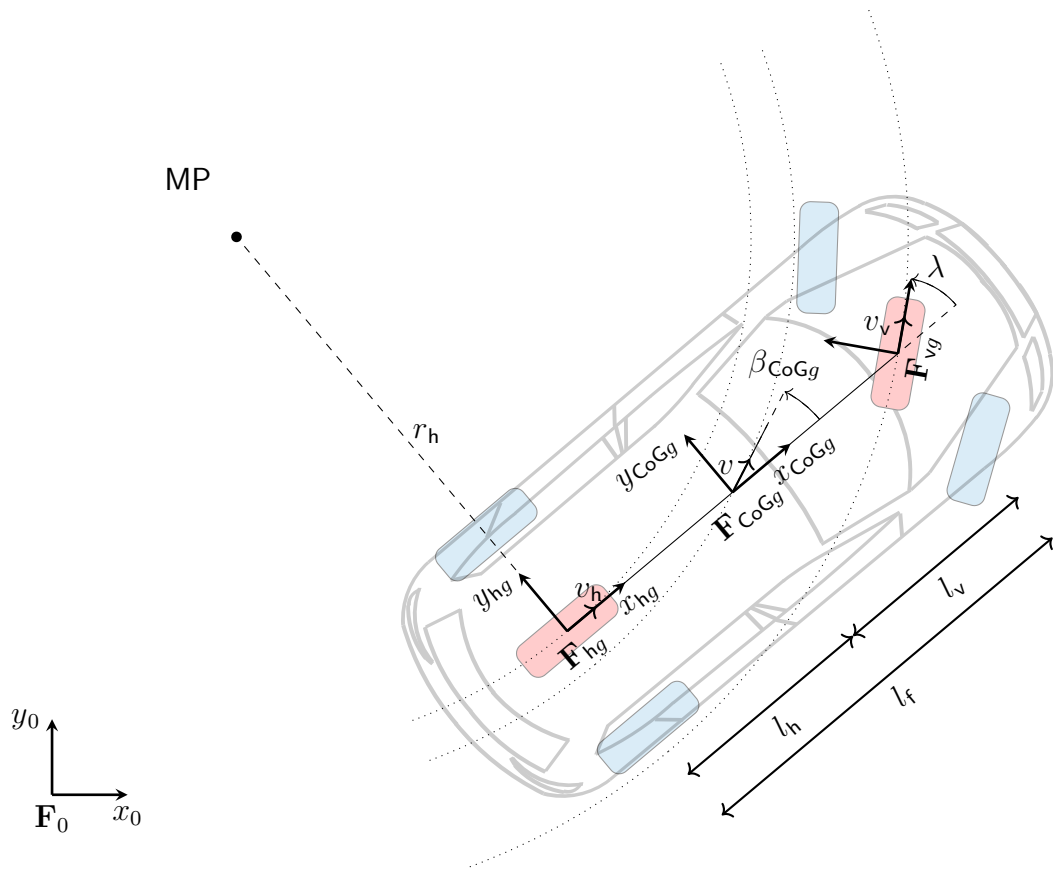


Abbildung 5.15: Kinematisches Einspurmodell

5.5.3.1 Kinematisches Einspurmodell

Im kinematischen Einspurmodell (KESM) wird die Annahme getroffen, dass die Räder von Vorder- und Hinterachse jeweils zu einem virtuellen Rad auf der Fahrzeuglängsachse zusammengefasst werden können. Die virtuellen Räder bewegen sich genau entlang ihrer jeweiligen Längsrichtung, die Bewegung ergibt sich allein aus den geometrischen Zusammenhängen. Der fiktive Schwerpunkt CoGg des Fahrzeugs liegt in der Fahrbahnebene, und es treten weder Roll- noch Nick- noch Hubbewegungen auf. Für diese reine Abrollbewegung werden die Reifenschräglaufwinkel und damit die auftretenden Querkräfte vereinfachend zu Null gesetzt, was eine gute Näherung für die geringen Querbeschleunigungen ist, die bei langsamer Fahrt auftreten.

Bewegungsgleichungen des Hinterachsmittelpunkts Für die Bewegung des Fahrzeugs in der Ebene ergibt sich eine Rotation um seinen Momentanpol MP entsprechend Abb. 5.15, die Fahrzeughinterachse steht dabei orthogonal zur Bewegungsrichtung des Fahrzeugs.

Der Radius r_h des Hinterachsmittelpunkts ergibt sich allein aus geometrischen Bedingungen in Abhängigkeit des fiktiven Lenkwinkels λ am Vorderachsmittelpunkt:

$$r_h = \frac{l_v + l_h}{\tan(\lambda)}. \quad (5.48)$$

Unter der Annahme, dass die Geschwindigkeit am Hinterachsmittelpunkt v_h sowie der fiktive Lenkwinkel λ am Vorderachsmittelpunkt stationär sind, ergibt sich damit für die Bewegung des Hinterachsmittelpunkts im weltfesten Koordinatensystem S_0 folgende Differentialgleichung in Zustandsraumdarstellung:

$$\begin{pmatrix} {}^0\dot{x}_h \\ {}^0\dot{y}_h \\ {}^0\dot{\Psi}_h \end{pmatrix} = \begin{pmatrix} \cos({}^0\Psi) \\ \sin({}^0\Psi) \\ \frac{\tan(\lambda)}{l_v + l_h} \end{pmatrix} v_h. \quad (5.49)$$

An dieser Stelle ist noch anzumerken, dass in anderen Arbeiten anstelle von Gleichung (5.48) oftmals die Näherung $\lambda \approx \frac{l_v + l_h}{r_h}$ für kleine Lenkwinkel verwendet wird. Dieser angenäherte Lenkwinkel wird als Ackermann-Lenk Winkel bezeichnet [vgl. Mitschke und Wallentowitz, 2014, S. 635]). Aufgrund der in dieser Arbeit betrachteten großen Lenkeinschläge wird im Folgenden weiter mit dem exakten Wert nach Gleichung (5.48) gerechnet.

Zeitdiskrete Darstellung Für die Berechnung auf Digitalrechnern werden üblicherweise zeitdiskrete Differenzengleichungen anstelle von zeitkontinuierlichen Differentialgleichungen verwendet, damit ergibt sich:

$$\mathbf{x}_{k+1}^{\text{KESM}} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k, \Delta t), \quad (5.50)$$

$$\begin{pmatrix} {}^0x_h \\ {}^0y_h \\ {}^0\Psi_h \end{pmatrix}_{k+1} = \begin{pmatrix} {}^0x_{h,k} + v_k \cos({}^0\Psi_k) \Delta t \\ {}^0y_{h,k} + v_k \sin({}^0\Psi_k) \Delta t \\ {}^0\Psi_{h,k} + v_k \frac{\tan(\lambda)}{l_v + l_h} \Delta t \end{pmatrix}. \quad (5.51)$$

Bewegungsgleichungen des Schwerpunktes Benötigt man allerdings die Bewegungsgleichungen für einen anderen Punkt des Fahrzeugs, beispielsweise für den Ursprung des bodenprojizierten Schwerpunktkoordinatensystems S_{CoG} , dann werden die Gleichungen umfangreicher. Diese Beziehungen werden im Folgenden dargestellt.

Analog zum Hinterachsmittelpunkt bewegen sich alle anderen Punkte P des Fahrzeugs auf Kreisbahnen um den Momentanpol MP, ihre Bewegungsrichtung ist dabei stets orthogonal zur Gerade durch MP und P .

Mit Hilfe des Satz des Pythagoras ergibt sich damit für den Radius r_{CoG} der Kreisbahn des Schwerpunktes um MP:

$$r_{\text{CoG}} = \sqrt{r_h^2 + l_h^2}. \quad (5.52)$$

Im Gegensatz zu S_{hg} bewegt sich S_{CoGg} nicht in Richtung der Fahrzeuglängsachse, sondern in Richtung der Tangente an seiner Kreisbahn. Dieser Winkel zwischen der Bewegungsrichtung eines Punktes des Fahrzeugs und der Fahrzeuglängsachse wird als Schwimmwinkel β bezeichnet.

Für den Schwerpunkt berechnet sich der Schwimmwinkel ${}^{CoG}\beta$ zu:

$${}^{CoG}\beta = \arctan\left(\frac{l_h}{r_h}\right) = \arctan\left(\frac{l_h \tan(\lambda)}{l_v + l_h}\right). \quad (5.53)$$

Im ebenen Koordinatensystem \mathbf{F}_0 folgt daraus eine Bewegung mit der Geschwindigkeit ${}^{CoG}v$ in Richtung ${}^0\Psi + \beta$. Für ${}^{CoG}v$ folgt aus den geometrischen Beziehungen:

$$\dot{\Psi} = \frac{{}^{CoG}v}{r_{CoG}} = \frac{v_h}{r_h}, \quad (5.54)$$

$${}^{CoG}v = \frac{r_{CoG} v_h}{r_h} = \sqrt{1 + \frac{l_h^2 \tan^2(\lambda)}{(l_h + l_v)^2}} v_h. \quad (5.55)$$

Analog zu Gleichung (5.49) ergibt sich damit in zeitkontinuierlicher Zustandsraumdarstellung:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \quad (5.56)$$

$$\begin{pmatrix} {}^0\dot{x}_{CoG} \\ {}^0\dot{y}_{CoG} \\ {}^0\dot{\Psi}_{CoG} \end{pmatrix} = \begin{pmatrix} \cos({}^0\Psi + {}^{CoG}\beta) \\ \sin({}^0\Psi + {}^{CoG}\beta) \\ \frac{\tan(\lambda)}{\sqrt{(l_v + l_h)^2 + l_h^2 \tan^2(\lambda)}} \end{pmatrix} v_{CoG}. \quad (5.57)$$

Zeitdiskrete Darstellung Für die Berechnung auf Digitalrechnern werden üblicherweise zeitdiskrete Differenzgleichungen verwendet. Dabei werden für eine konsistente Darstellung im weiteren Verlauf des Kapitels die Gierrate $\dot{\Psi}$ und der Schwimmwinkel β_{CoG} in den Zustandsvektor aufgenommen:

$$\mathbf{x}_{k+1}^{KESM} = f(\mathbf{x}_k, \mathbf{u}_k, \Delta t), \quad (5.58)$$

$$\begin{pmatrix} {}^0x_{CoG} \\ {}^0y_{CoG} \\ {}^0\Psi_{CoG} \\ {}^0\dot{\Psi}_{CoG} \\ {}^{CoG}\beta \end{pmatrix}_{k+1} = \begin{pmatrix} {}^0x_{CoG,k} + {}^{CoG}v_k \cos({}^0\Psi_k + \Delta t {}^0\dot{\Psi}_k + {}^{CoG}\beta_k) \Delta t \\ {}^0y_{CoG,k} + {}^{CoG}v_k \sin({}^0\Psi_k + \Delta t {}^0\dot{\Psi}_k + {}^{CoG}\beta_k) \Delta t \\ {}^0\Psi_k + \Delta t {}^0\dot{\Psi}_k \\ \frac{\tan(\lambda_k)}{\sqrt{(l_v + l_h)^2 + l_h^2 \tan^2(\lambda_k)}} {}^{CoG}v_k \\ \arctan\left(\frac{l_h \tan(\lambda_k)}{l_v + l_h}\right) \end{pmatrix}. \quad (5.59)$$

Anmerkung: Eine eventuell vorhandene (ungelenkte) Tandem-Hinterachse wie bei den LKWs TULF und StrAsRob wird durch eine virtuelle Achse (am Mittelpunkt der Tandemachse) ersetzt. Linkes und rechtes Rad einer Achse werden anschließend wie beim Einspurmodell gewohnt zusammengefasst.

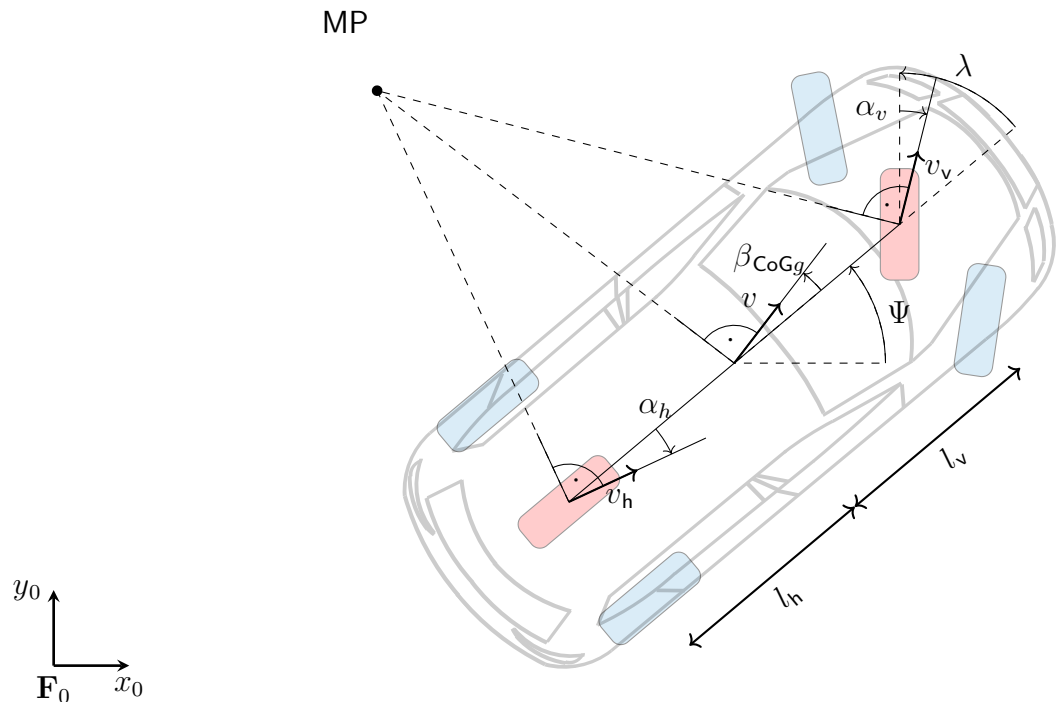


Abbildung 5.16: Dynamisches Einspurmodell.

5.5.3.2 Dynamisches Einspurmodell

Im bisher vorgestellten kinematischen Einspurmodell wurden niedrige Geschwindigkeiten angenommen, sodass auch bei Kurvenfahrt nur geringe Querbewegungen auftreten. Für diesen Fall wurden die Reifenschräglaufwinkel α_i und damit die auftretenden Querkräfte F_{yi} vereinfachend zu Null gesetzt. Bei höheren Geschwindigkeiten ist dies so nicht mehr gültig.

Beim dynamischen Einspurmodell (DESM), erstmals präsentiert von Riekert und Schunck [1940], ergibt sich die Bewegung aus dem Kräfte- und Momentengleichgewicht. Das DESM ist für höhere Geschwindigkeitsbereiche besser geeignet, da es explizit Reifenkräfte berücksichtigt. Gegenüber dem KESM werden jedoch wesentlich mehr Parameter benötigt, die nicht trivial zu bestimmen sind. Auf Seiten der Geometrie handelt es sich um den Achsabstand und die Schwerpunktlage. Für den dynamischen Bereich werden die Masse m , das Massenträgheitsmoment J_z und die Seitenkraftbeiwerte der Reifen c_v , c_h benötigt. Abb. 5.16 skizziert die wesentlichen Parameter des DESM.

Die Seitenkraft ergibt sich im einfachsten Fall linear:

$$F_{yi}(\alpha_i) = \alpha_i c_i, \quad (5.60)$$

bei größeren Schräglaufwinkeln wird diese nichtlinear berechnet, z. B. nach der *Magic Tyre Formula* [Pacejka und Besselink, 1997].

Das Kräftegleichgewicht in Längsrichtung des Fahrzeugschwerpunkts ergibt sich zu:

$$-m \cdot \text{CoG}_v \left({}^0\ddot{\Psi} + \text{CoG}\dot{\beta} \right) \sin(\text{CoG}\beta) = F_{yv}(\alpha_v) \sin(\lambda) . \quad (5.61)$$

Das Kräftegleichgewicht in Querrichtung des Fahrzeugschwerpunkts ergibt sich zu:

$$m \cdot \text{CoG}_v \left({}^0\ddot{\Psi} + \text{CoG}\dot{\beta} \right) \cos(\text{CoG}\beta) = F_{yv}(\alpha_v) \cos(\lambda) + F_{yh}(\alpha_h) . \quad (5.62)$$

Das Momentengleichgewicht um die Hochachse des Fahrzeugschwerpunkts ergibt sich zu:

$$J_z {}^0\ddot{\Psi} = l_v F_{yv}(\alpha_v) \cos(\lambda) - l_h F_{yh}(\alpha_h) . \quad (5.63)$$

Nach einer Kleinwinkelnäherung für Lenkwinkel λ und Schwimmwinkel $\text{CoG}\beta$ geht der Sinus-Term gegen Null, und das Gleichgewicht in Längsrichtung entfällt.

Für die Schräglaufwinkel ergibt sich:

$$\alpha_v = \lambda - \arctan\left(\frac{\text{CoG}_v \sin(\text{CoG}\beta) + l_v {}^0\dot{\Psi}}{\text{CoG}_v \cos(\text{CoG}\beta)} \right) , \quad (5.64)$$

$$\alpha_h = -\arctan\left(\frac{\text{CoG}_v \sin(\text{CoG}\beta) - l_h {}^0\dot{\Psi}}{\text{CoG}_v \cos(\text{CoG}\beta)} \right) . \quad (5.65)$$

Bei Kleinwinkelnäherung für den Schwimmwinkel $\text{CoG}\beta$ ist auch hier eine Linearisierung möglich:

$$\alpha_v = \lambda - \text{CoG}\beta + \frac{l_v {}^0\dot{\Psi}}{\text{CoG}_v} , \quad \alpha_h = \text{CoG}\beta - \frac{l_h {}^0\dot{\Psi}}{\text{CoG}_v} . \quad (5.66)$$

Das lineare dynamische Einspurmodell ist eine Vereinfachung des nichtlinearen dynamischen Einspurmodells und betrachtet nur den dynamischen Bereich der Fahrzeugdynamik, in der die Seitenkraftkennlinie der Reifen durch das lineare Produkt aus dem Seitenkraftbeiwert und dem Schräglaufwinkel (Gleichung (5.60)) beschrieben werden kann.

Zeitkontinuierliche Darstellung Bei Nutzung der linearen Ausdrücke für die Schräglaufwinkel ergibt sich die folgende zeitkontinuierlicher Zustandsraumdarstellung für die Fahrzeugbewegung in der Ebene, die durch Gierrate und Schwimmwinkel beschrieben wird:

$$\begin{pmatrix} \text{CoG}\dot{\beta} \\ {}^0\ddot{\Psi} \end{pmatrix} = \begin{pmatrix} -\frac{c_v + c_h}{m \cdot \text{CoG}_v} & \frac{c_h l_h - c_v l_v}{m \cdot \text{CoG}_v^2} - 1 \\ \frac{c_h l_h - c_v l_v}{J_z} & \frac{c_v l_h^2 - c_v l_v^2}{J_z \cdot \text{CoG}_v} \end{pmatrix} \cdot \begin{pmatrix} \text{CoG}\beta \\ {}^0\dot{\Psi} \end{pmatrix} + \begin{pmatrix} \frac{c_v}{m \cdot \text{CoG}_v} \\ \frac{c_v l_v}{J_z} \end{pmatrix} \lambda . \quad (5.67)$$

Zeitdiskrete Darstellung Für die zeitdiskrete Modellierung des DESM kommt hier eine linearisierte und vereinfachte Implementierung des Kollegen Hecker zum Einsatz, die von der Lenkrate $\dot{\lambda}$ als Eingangsgröße getrieben wird:

$$\mathbf{x}_{k+1}^{\text{DESM}} = \Phi_k^{\text{DESM}} \mathbf{x}_k^{\text{DESM}} + \mathbf{B}_k \mathbf{u}_k \quad (5.68)$$

mit

$$\mathbf{x} = \begin{pmatrix} {}^0\Psi_{\text{CoG}} \\ \text{CoG}\beta \\ \lambda \\ {}^0\Psi_{\text{CoG}} \end{pmatrix}, \quad (5.69)$$

$$\mathbf{u} = \dot{\lambda}, \quad (5.70)$$

$$\Phi = \begin{cases} \begin{pmatrix} 1 - \Delta t \left(\frac{c_v l_v^2 + c_h l_h^2}{J_z v} \right) & -\Delta t \frac{c_v l_v - c_h l_h}{J_z} & \Delta t \frac{c_v l_v}{J_z} & 0 \\ -\Delta t \left(\frac{c_v l_v - c_h l_h}{m v^2} + 1 \right) & 1 - \Delta t \frac{c_v + c_h}{m v} & \Delta t \frac{c_v}{m v} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} & v \geq 0,01 \text{ m/s} \\ \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0,5 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} & \text{sonst,} \end{cases} \quad (5.71)$$

$$\mathbf{B} = \begin{cases} \begin{pmatrix} \left(\frac{\Delta t^2 c_v l_v}{2 J_z} \right) \\ \frac{\Delta t^2 c_v}{2 m v} \\ \Delta t \\ 0 \end{pmatrix} & v \geq 0,01 \text{ m/s} \\ \begin{pmatrix} 0 \\ 0 \\ \Delta t \\ 0 \end{pmatrix} & \text{sonst.} \end{cases} \quad (5.72)$$

Aus Gründen der numerischen Stabilität findet eine Fallunterscheidung über die Geschwindigkeit statt. Die Geschwindigkeit ist in dieser Implementierung nicht Teil des Zustandsvektors, sondern wird als konstanter Parameter aus dem letzten Zeitschritt übernommen: $v = {}^{\text{CoG}}v_k$.

Die Integration der Position erfolgt analog zum KESM in Gleichung (5.59):

$$\begin{pmatrix} {}^0x_{\text{CoG}} \\ {}^0y_{\text{CoG}} \end{pmatrix}_{k+1} = \begin{pmatrix} {}^0x_{\text{CoG},k} + {}^{\text{CoG}}v_k \cos\left({}^0\Psi_k + \Delta t {}^0\dot{\Psi}_k + {}^{\text{CoG}}\beta_k\right) \Delta t \\ {}^0y_{\text{CoG},k} + {}^{\text{CoG}}v_k \sin\left({}^0\Psi_k + \Delta t {}^0\dot{\Psi}_k + {}^{\text{CoG}}\beta_k\right) \Delta t \end{pmatrix}. \quad (5.73)$$

Einschränkungen Insbesondere die Seitenkraftbeiwerte der Reifen sind schwierig zu bestimmen, da sie vom aktuellen Untergrund abhängen und dieser im Offroad-

Betrieb nicht konstant ist. Da sich diese Arbeit primär mit der nicht-urbanen Fahrt auf unbefestigten Wegen beschäftigt, bei denen die Modellannahmen nicht zu treffen, wurde das DESM nur im monolithischen Eigenbewegungsschätzer (siehe Abschnitt 5.6.1) verwendet, aber nicht mehr für den modularen Eigenbewegungsschätzer (siehe Abschnitt 5.6.2) implementiert.

5.5.3.3 Überblendung

Sowohl das KESM als auch das DESM sind nicht für alle Geschwindigkeitsbereiche geeignet.

Daher wurde ähnlich wie in [Kehl, 2007] eine geschwindigkeitsabhängige Überblendung entwickelt: Für $v \leq 6$ m/s kommt ausschließlich das KESM zum Einsatz, für $v \geq 8$ m/s kommt ausschließlich das DESM zum Einsatz. Im Geschwindigkeitsbereich $6 \text{ m/s} \leq v \leq 8 \text{ m/s}$ erfolgt eine kontinuierliche, stoßfreie Überblendung.

5.5.4 Sonstige Fahrzeugbewegungsmodelle

Es gibt eine Vielzahl an weiteren Modellen zur Beschreibung der Fahrzeugbewegung. So gibt es Erweiterungen auf andere Lenk-Geometrien (Hinterachslenkung, Allradlenkung, ...), Modelle für die Beschreibung eines Kettenantriebs oder differentiellen Radantriebs, das Zweispurmodell und viele mehr. Diese werden allerdings im Rahmen dieser Arbeit nicht weiter betrachtet.

5.5.5 Modelle für weitere Teilbereiche der Fahrzeugdynamik

Die Einspurmodelle beschreiben nur die Fahrzeugbewegung in der Ebene. Reale Fahrzeuge, insbesondere im Gelände, bewegen sich jedoch auf einer dreidimensionalen Bahn. In dieser Arbeit werden Roll- und Nickbewegung jeweils als lineares System erster Ordnung modelliert. Diese Entkopplung und vereinfachte Modellierung ist aufgrund der hohen Update-Rate des Filters (Messungen der IMU erfolgen mit 100 Hz) hinreichend genau. Dies wird hier am Beispiel der Nickbewegung gezeigt, zuerst als Differentialgleichung im Zeitkontinuierlichen:

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} \quad (5.74)$$

$$\begin{pmatrix} {}^0\dot{\Theta} \\ {}^0\ddot{\Theta} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} {}^0\Theta \\ {}^0\dot{\Theta} \end{pmatrix}. \quad (5.75)$$

Die Differenzengleichung für die zeitdiskrete Darstellung ergibt sich zu:

$$\mathbf{x}_{k+1} = \mathbf{\Phi}\mathbf{x}_k \quad (5.76)$$

$$\begin{pmatrix} {}^0\Theta \\ {}^0\dot{\Theta} \end{pmatrix}_{k+1} = \begin{pmatrix} 1 & \Delta t \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} {}^0\Theta \\ {}^0\dot{\Theta} \end{pmatrix}_k. \quad (5.77)$$

5.6 Filterarchitektur

Im Rahmen dieser Arbeit wurden zwei unterschiedliche Architekturansätze für die Eigenbewegungsschätzung und Lokalisierung verfolgt: Einerseits ein eher monolithischer Aufbau für ein fest vorgegebenes Set an Sensoren (siehe Abschnitt 5.6.1), andererseits – motiviert durch Versuchsfahrzeuge mit variablem Sensorsetup – ein dynamisch erzeugter modularer Schätzer (siehe Abschnitt 5.6.2).

Während die Beschreibung der Filtervarianten in Abschnitt 5.4 von festen Abtastzeiten $\Delta t = \text{const}$ ausgeht, kommt hier eine dynamische Abtastzeit $\Delta t = \text{var}$ zum Einsatz. Die vorgestellten Filter sind datengetrieben, d. h. jeder Sensoreingang i wird in einem separaten Thread verarbeitet, der ähnlich zu einem Interrupt bei jeder neuen Messung $\mathbf{y}_{i,k}^{\text{Sensor}}$ die Prädiktion auf den Messzeitstempel t_k^{Sensor} sowie die Innovation mit der neuen Messung $\mathbf{y}_k^{\text{Sensor}}$ auslöst. Damit kann jede Messung verarbeitet werden, während bei festen Abtastzeiten einzelne Messungen ignoriert werden, die hochfrequenter als der Systemtakt eintreffen. Für OOSM kommt hier die einfache Brute-Force-Methode zum Einsatz.

Für beide Varianten kommt – im Gegensatz zur Modellierung in \mathbf{F}_{ECEP} innerhalb eines INS – ein ebenes Koordinatensystem \mathbf{F}_0 für die Zustandsschätzung zum Einsatz, da der Einsatzbereich auf Landfahrzeuge begrenzt ist.

Aus denselben Gründen können Euler-Winkel zur Beschreibung der Rotation verwendet werden: bei Landfahrzeugen besteht kein Risiko des sog. Gimbal-Locks. Allerdings muss für alle Winkel beachtet werden, dass es sich um zirkuläre Größen handelt: die Differenz $\Delta\Psi = \Psi_1 - \Psi_2$ zwischen für $\Psi_1 = 1^\circ$ und $\Psi_2 = 359^\circ$ beträgt nicht -358° sondern 2° .

5.6.1 Monolithischer Eigenbewegungsschätzer

Der monolithische Eigenbewegungsschätzer ist auf die in Kapitel 3 beschriebene Sensorarchitektur von MuCAR-3 und MuCAR-4 zugeschnitten und basiert rein auf Messungen; das modellierte System besitzt keine Steuergrößen. Als Messungen kommen bereits vorgefilterte Daten aus der INS vom Typ OxtS RT3003 v1 sowie aggregierte Messungen aus dem Low-Level-System zum Einsatz.

Messungen Aus den Messdaten des INS werden die drei Winkelgeschwindigkeitsmessungen $\omega_{[x,y,z]}$ sowie die drei Eulerwinkel (${}^0\Phi$, ${}^0\Theta$, ${}^0\Psi$) verwendet, zusätzlich die Beschleunigung ${}^{\text{veh}}a_{[x]}$ und Geschwindigkeit ${}^{\text{veh}}v_{[x]}$ in Fahrzeuginnenrichtung, die Vertikalgeschwindigkeit ${}^0v_{[z]}$ zur Erdoberfläche sowie die globale, GNSS-gestützte Positionsschätzung (0x , 0y , 0z):

$$\mathbf{y}_{\text{INS}} = \left(\omega_{[x]}, \omega_{[y]}, \omega_{[z]}, {}^0\Phi, {}^0\Theta, {}^0\Psi, {}^{\text{veh}}a_{[x]}, {}^{\text{veh}}v_{[x]}, {}^0v_{[z]}, {}^0x, {}^0y, {}^0z \right)^T. \quad (5.78)$$

Für diese Messungen werden vom INS zeitlich veränderliche Varianzen mitgeliefert.

Aus den fahrzeugseitigen Messungen kommen die mittlere Fahrzeuggeschwindigkeit v_x , Lenkwinkel λ und Lenkrate $\dot{\lambda}$ sowie Daten aus dem Fahrwerk (Rollwinkel Φ_{chassis} und Nickwinkel Θ_{chassis} sowie Höhe ggü. dem Untergrund z_{chassis}) zur Verwendung:

$$\mathbf{y}_{\text{lowlevel}} = \left(v_x, \lambda, \dot{\lambda}, \Phi_{\text{chassis}}, \Theta_{\text{chassis}}, z_{\text{chassis}} \right)^T. \quad (5.79)$$

Hier werden von der Sensorik keine Varianzen bereitgestellt, daher wurden diese einmalig empirisch ermittelt.

Um die Messmodelle hier simpel zu halten, werden die Messungen vor Übergabe an das Filterframework vom Sensorkoordinatensystem $\mathbf{F}_{\text{Sensor}}$ in das Ego-Koordinatensystem \mathbf{F}_{veh} transformiert. Da die Hebelarme vernachlässigbar klein sind, ist der dadurch entstehende Fehler nach Abschnitt 5.5.1.4 für MuCAR-3 und MuCAR-4 ebenso zu vernachlässigen.

Zustandsvektor und Prozessmodelle Der Zustandsvektor \mathbf{x} besteht hier aus mehreren Unterzustandsvektoren, deren Dynamik weitgehend entkoppelt ist und die daher mit einzelnen Prozessmodellen prädiert werden. Die benötigten Informationen aus anderen Unterzustandsvektoren werden über quasi-konstante Parameter \mathbf{q} berücksichtigt, die mit den Zustandsschätzungen aus dem vorherigen Zeitschritt befüllt werden. Ein Vorteil dieser separierten Prozessmodelle ist die einfachere Ableitung für die Ermittlung der Jacobi-Matrizen.

$$\mathbf{x} = \left(\mathbf{x}_{[\text{lon}]}, \mathbf{x}_{[\text{lat}]}, \mathbf{x}_{[\text{roll}]}, \mathbf{x}_{[\text{pitch}]}, \mathbf{x}_{[\text{pos2EME}]}, \mathbf{x}_{[\text{pos2DR}]}, \mathbf{x}_{[\text{altDR}]}, \mathbf{x}_{[\text{altEME}]}, \mathbf{x}_{[\text{headDR}]} \right)^T. \quad (5.80)$$

Die Längsbewegung wird als System 2. Ordnung modelliert: $f_{\text{lon}}(\mathbf{x}_{[\text{lon}]}, \mathbf{q})$. Hier wird ein Offset b_v zwischen INS- und Odometrie-basierter Geschwindigkeitsmessung mitgeschätzt:

$$\mathbf{x}_{[\text{lon}]} = \left(v_{\text{veh}}, a_{\text{veh}}, b_v \right)^T. \quad (5.81)$$

Die Querdynamik wird über $f_{\text{lat}}(\mathbf{x}_{[\text{lat}]}, \mathbf{q})$ modelliert. Hier werden kinematisches und dynamisches Fahrzeugmodell überblendet. Zusätzlich wird ein Lenkwinkeloffset b_λ mitgeschätzt:

$$\mathbf{x}_{[\text{lat}]} = \left(\dot{\Psi}, \beta_{\text{veh}}, \lambda_{\text{veh}}, \Psi, b_\lambda \right)^T. \quad (5.82)$$

Die Rotation um die Rollachse wird als System 2. Ordnung modelliert: $f_{\text{roll}}(\mathbf{x}_{[\text{roll}]}, \mathbf{q})$. Dabei wird der Winkel in einen niederfrequenten Teil Φ_{env} (der im Prinzip der Fahrbahnneigung resp. Steigung entspricht) und einen hochfrequenten Teil Φ_{ego} (der die Lage zwischen Fahrzeug und Aufstandsebene beschreibt) unterteilt:

$$\mathbf{x}_{[\text{roll}]} = \left(\Phi_{\text{ego}}, \Phi_{\text{env}}, \dot{\Phi} \right)^T. \quad (5.83)$$

Die Nickachse $\mathbf{x}_{[pitch]}$ ist analog dazu modelliert:

$$\mathbf{x}_{[pitch]} = \left({}^0\Theta_{\text{ego}}, {}^0\Theta_{\text{env}}, {}^0\dot{\Theta} \right)^T. \quad (5.84)$$

Für die globale Position $\mathbf{x}_{[pos2EME]}$ wird die in die Ebene projizierte Geschwindigkeit in Fahrtrichtung mittels $f_{\text{pos2EME}}(\mathbf{x}_{[pos2EME]}, \mathbf{q})$ integriert:

$$\mathbf{x}_{[pos2EME]} = \left({}^0x_{\text{EME}}, {}^0y_{\text{EME}} \right)^T. \quad (5.85)$$

Analog wird die globale Position für das Dead-Reckoning $\mathbf{x}_{[pos2DR]}$ modelliert:

$$\mathbf{x}_{[pos2DR]} = \left({}^0x_{\text{DR}}, {}^0y_{\text{DR}} \right)^T. \quad (5.86)$$

Die globale Ausrichtung im Dead-Reckoning $\mathbf{x}_{[headDR]}$ wird aufintegriert, hier kommt wieder ein System 2. Ordnung zum Einsatz:

$$\mathbf{x}_{[headDR]} = \left({}^0\Psi_{\text{DR}}, {}^0\dot{\Psi}_{\text{DR}} \right)^T. \quad (5.87)$$

Die Vertikaldynamik für $\mathbf{x}_{[altDR]}$ und $\mathbf{x}_{[altEME]}$ wird über Geschwindigkeit und Steigung bzw. Nickwinkel getrieben:

$$\mathbf{x}_{[altEME]} = \left({}^0h_{\text{EME}}, {}^0\dot{h}_{\text{EME}} \right)^T, \quad (5.88)$$

$$\mathbf{x}_{[altDR]} = \left({}^0h_{\text{DR}}, {}^0\dot{h}_{\text{DR}} \right)^T. \quad (5.89)$$

Filter Im monolithischen Eigenbewegungsschätzer kommt eine Implementierung des EKF zum Einsatz. Die nichtlinearen Prozess- $f(\cdot)$ und Messmodelle $g(\cdot)$ wurden anhand der Differentialgleichungen mittels Diskretisierung bestimmt. Aufgrund der kleinen Unterzustände konnten die Jacobi-Matrizen Φ und C , die für die Berechnung der Schätzfehlerkovarianz und des Kalman-Gains benötigt werden, analytisch über die partielle Ableitungen bestimmt werden.

Modusumschaltung Im Filterframework kommen unterschiedliche Modi zum Einsatz, um modusabhängige Varianzen zu verwenden und gezielt Messungen unterdrücken zu können, wenn diese empirisch in einzelnen Modi ungeeignet sind. Als Modi kommen hier „Stehen“, „Beschleunigen“, „Fahren“ und „Verzögern“ (jeweils vorwärts und rückwärts) zum Einsatz, um die im Übergangsbereich zwischen Stehen und Fahren auftretenden nichtlinearen Messungen der Odometrie und ein Überschwingen der Geschwindigkeitsschätzung der INS zu behandeln.

Im Normalfall werden für alle Messwerte die vom Sensor bzw. Sensormodell mitgelieferten Varianzen übernommen. Für die fahrzeugseitige Geschwindigkeitsmessung wird die Varianz im Stand massiv auf $1 \cdot 10^{-10} \text{ m}^2/\text{s}^2$ reduziert, beim Beschleunigen und Verzögern auf $1 \cdot 10^{-1} \text{ m}^2/\text{s}^2$. Für die INS-basierte Geschwindigkeitsmessung wird

die Varianz im Stand auf $1 \cdot 10^{-1} \text{ m}^2/\text{s}^2$ fixiert, die Varianzen der Beschleunigungsmessungen auf $1 \cdot 10^{-1} \text{ m}^2/\text{s}^4$. Im Stand werden bei ${}^{\text{veh}}a < 0,1 \text{ m/s}^2$ zusätzlich die Messung ${}^{\text{veh}}a_{[x]}$ sowie die Gierrate ${}^0\dot{\Psi}_{\text{DR}}$ des Dead-Reckonings auf Null gesetzt.

Die Modi (*Zustände*) und Übergänge (*Transitionen*) sind als Zustandsautomat⁹ modelliert, die Übergangsbedingungen wurden empirisch bestimmt.

Ignorieren von Messungen Das entwickelte Filterframework bietet die Möglichkeit, potentiell durch GNSS-Störungen verfälschte Messdaten (Position, Geschwindigkeit, Orientierung Ψ) zu ignorieren. Dies geschieht, wenn das INS in den Statusdaten eine Abweichung zwischen INS- und GNSS-Position oder andere Fehler meldet. Zusätzlich kann dieses Ignorieren über eine externe Schnittstelle aktiviert werden (vgl. Abschnitt 5.9). In diesem Fall entwickelt sich auch $\mathbf{x}_{[\text{pos2EME}]}$, $\mathbf{x}_{[\text{lat}]}$, $\mathbf{x}_{[\text{alt2EME}]}$ temporär analog zum Dead-Reckoning durch Integration.

5.6.2 Dynamisch erzeugter modularer Eigenbewegungsschätzer

Im Gegensatz zum monolithischen Eigenbewegungsschätzer sind beim dynamisch erzeugten modularen Eigenbewegungsschätzer der Aufbau des Zustandsvektors sowie der Prozess- und Messmodelle nicht festgeschrieben. Je nach Aufbau des Roboters wird die Filterstruktur dynamisch generiert.

Als Quelle für alle Informationen zum Fahrzeug und den Sensoren dient eine zentrale Schnittstelle über eine Bibliothek, das sog. `tas::vehicle`, die je nach verwendetem Ego-Fahrzeug die passenden Parameter bereitstellt. Auf Implementierungsdetails wird hier verzichtet.

Basierend auf den Informationen zum Bewegungsmodell, die `tas::vehicle` für das aktuelle Fahrzeug bereitstellt, wird zuerst ein allgemeiner Teilzustandsvektor $\mathbf{x}^{\text{Fahrzeug}}$ erzeugt. Dessen Zustandsgrößen beschreiben die Pose und Bewegung des Fahrzeugs in der Welt sowie weitere dynamische Zustände. Ein fahrzeugspezifisches Prozessmodell

$$f^{\text{Fahrzeug}}(\mathbf{x}^{\text{Fahrzeug}}, \mathbf{u}^{\text{Fahrzeug}}, \mathbf{q}^{\text{Fahrzeug}}) \quad (5.90)$$

dient zur Prädiktion von $\mathbf{x}^{\text{Fahrzeug}}$. Der Parametervektor $\mathbf{q}^{\text{Fahrzeug}}$ enthält konstante Parameter des Fahrzeugs, bspw. die Achsgeometrie.

`tas::vehicle` als zentrale Instanz der Fahrzeugparametrierung verwaltet alle montierten Sensoren \mathbb{S} in einer Baumstruktur. Aus dieser werden alle Sensoren selektiert, die sich in einen Eigenbewegungssensor überführen lassen:

$$\mathbb{S}_{\text{EME}} = \{\mathbb{S} \ni S_i \mid S_i \leq S_{\text{EME}}\}. \quad (5.91)$$

⁹In Abschnitt 7.4 wird der Zustandsautomat für einen anderen Verwendungszweck eingeführt und erklärt.

Für jeden Eigenbewegungssensor $S \in \mathbb{S}_{\text{EME}}$ existiert nun ein Teilzustandsvektor $\mathbf{x}_i^{\text{Sensor}}$, dessen Zustandsgrößen jeweils sensorspezifische Werte wie Skalierungsfehler, Offsets oder Einbaulagen beschreiben. Ein Prozessmodell

$$f_i^{\text{Sensor}}(\mathbf{x}_i^{\text{Sensor}}, \mathbf{u}_i^{\text{Sensor}}, \mathbf{q}_i^{\text{Sensor}}) \quad (5.92)$$

dient zur Prädiktion von $\mathbf{x}_i^{\text{Sensor}}$. Da $\mathbf{x}_i^{\text{Sensor}}$ meist nur quasi-konstante Zustandsgrößen enthält, vereinfacht sich die Prädiktion gemäß Gleichung (5.96). Der Parametervektor $\mathbf{q}_i^{\text{Sensor}}$ enthält weitere Parameter, die nicht Teil des Zustandsraums sind, aber in den Prozess- und Messmodellen verwendet werden.

Der komplette Zustandsvektor \mathbf{x} setzt sich aus einem allgemeinen und n sensorspezifischen Teilzustandsvektoren $\mathbf{x}_i^{\text{Sensor}}$ zusammen:

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}^{\text{Fahrzeug}} \\ \mathbf{x}_1^{\text{Sensor}} \\ \vdots \\ \mathbf{x}_n^{\text{Sensor}} \end{pmatrix}. \quad (5.93)$$

Für die Prädiktion gilt:

$$\mathbf{x}_{k+1}^* = \begin{pmatrix} \mathbf{x}_{k+1}^{*,\text{Fahrzeug}} \\ \mathbf{x}_{1,k+1}^{*,\text{Sensor}} \\ \vdots \\ \mathbf{x}_{n,k+1}^{*,\text{Sensor}} \end{pmatrix} = \begin{pmatrix} f(\hat{\mathbf{x}}_k^{\text{Fahrzeug}}, \mathbf{u}_k, \mathbf{q}_k^{\text{Fahrzeug}}) \\ f(\hat{\mathbf{x}}_{1,k}^{\text{Sensor}}, \mathbf{u}_k, \mathbf{q}_{1,k}^{\text{Sensor}}) \\ \vdots \\ f(\hat{\mathbf{x}}_{n,k}^{\text{Sensor}}, \mathbf{u}_k, \mathbf{q}_{n,k}^{\text{Sensor}}) \end{pmatrix}. \quad (5.94)$$

Für die Innovation sind die Messmodelle $g_i^{\text{Sensor}}(\mathbf{x}, \mathbf{u}, \mathbf{q})$ der einzelnen Sensoren S_i essentiell, die sowohl Elemente des allgemeinen als auch ihres spezifischen Teilzustandsvektors auf die Messgrößen abbilden. Jeder Sensor S_i kann mehrere Messmodelle $g_{i,j}^{\text{Sensor}}(\cdot)$, $j = 1, \dots, m$ bereitstellen, deren Messwerte unkorreliert zueinander sind. Abschnitt 5.7 stellt die Messmodelle der unterschiedlichen Sensoren vor. Die Kovarianzmatrizen des Messrauschens $\mathbf{R}_{i,j}^{\text{Sensor}}$ sind hier jeweils als Diagonalmatrizen modelliert und können entweder einmalig parametrisiert werden oder dynamisch bei neuen Messdaten gesetzt werden.

Im Gegensatz zu einer monolithischen Struktur bietet ein dynamisch generierter modularer Schätzer einige Vorteile:

- Es befinden sich nur tatsächlich benötigte und auch beobachtbare Zustände im Zustandsvektor, damit wird ein kleinerer Zustandsvektor erzielt.
- Kleine spezialisierte Prozessmodelle sind übersichtlicher und leichter auszutauschen.
- Redundanz im Gesamtsystem ist durch Integration von mehreren Sensoren, die dieselben allgemeinen Zustandsgrößen beobachten, einfach zu erreichen. Auch bei baugleichen Sensoren wird durch die Verwendung von sensorspezifischen

Zustandsgrößen $\mathbf{x}_i^{\text{Sensor}}$ sichergestellt, dass z. B. Offsets nur vom zugehörigen Sensor beobachtet werden und sich somit diese Sensoren nicht gegenseitig stören.

- Über die Prozess- und Messmodelle ergibt sich eine Verkopplung der (Teil) Zustandsvektoren, sodass die Schätzfehlerkovarianzmatrix nicht diagonal oder blockdiagonal bleibt. Damit werden auch Sensorfehler und Einbaulagen beobachtbar.

Modumschaltung Ähnlich wie im monolithischen Filter kommen auch im dynamisch generierten modularen Schätzer verschiedene Modi zum Einsatz, um modusabhängige Varianzen zu verwenden und bspw. in der Initialisierungsphase andere Messmodelle zu verwenden. Als Modi kommen hier „Initiales Stehen“, „Initiales Fahren“, „Stehen“ und „Fahren“ zum Einsatz. Die Modellierung der Übergänge erfolgt ebenfalls als Zustandsautomat.

5.7 Modellierung der Sensoren zur Messung der Fahrzeugbewegung

Die Versuchsfahrzeuge aus Kapitel 3 sind mit unterschiedlichen Sensoren ausgestattet, die zur Eigenbewegungsschätzung und Lokalisierung beitragen können.

Die Nomenklatur von Zustandsraum und Filtergleichungen basiert auf Abschnitt 5.4.

Wie in Abschnitt 5.6.2 beschrieben, können die einzelnen Sensoren i einen sensor-spezifischen Teilzustandsvektor $\mathbf{x}_i^{\text{Sensor}}$ besitzen. Dieser wird mit einem separaten Prozessmodell aus Gleichung (5.92) prädiert:

$$\mathbf{x}_{i,k}^{*,\text{Sensor}} = f_i^{\text{Sensor}}(\hat{\mathbf{x}}_{i,k-1}^{\text{Sensor}}, \mathbf{u}_{k-1}, \mathbf{q}_i^{\text{Sensor}}). \quad (5.95)$$

Im Regelfall ist hier der Systemeingang $\mathbf{u}_{k-1} = 0$. Die meisten sensorspezifischen Teilzustandsvektoren enthalten nur quasi-konstante Parameter, sodass sich die Prädiktion vereinfacht zu:

$$\mathbf{x}_{i,k}^{*,\text{Sensor}} = \hat{\mathbf{x}}_{i,k-1}^{\text{Sensor}}. \quad (5.96)$$

Für die Messung besitzt jeder Sensor i ein spezialisiertes *Egomotion-Measurement-Model* $g_i^{\text{Sensor}}(\cdot)$, welches den allgemeinen Zustandsvektor $\mathbf{x}^{\text{Fahrzeug}}$ und den sensor-spezifischen Teilzustandsvektor $\mathbf{x}_i^{\text{Sensor}}$ auf den Messvektor \mathbf{y} abbildet:

$$\mathbf{y}_{i,k}^{\text{Sensor}} = g_i^{\text{Sensor}}\left(\begin{pmatrix} \mathbf{x}_k^{\text{Fahrzeug}} \\ \mathbf{x}_{i,k}^{\text{Sensor}} \end{pmatrix}, \mathbf{q}_i^{\text{Sensor}}\right). \quad (5.97)$$

In den folgenden Abschnitten werden Messgrößen, Messmodelle und sensorspezifische Teilzustandsvektoren vorgestellt.

5.7.1 Allgemeines Messmodell

Bevor auf die Messungen der einzelnen Sensoren eingegangen wird, soll an dieser Stelle zunächst das allen Sensoren gemeinsame Vorgehen erläutert werden.

Messmodell Gleichung (5.21) beschreibt die nichtlineare Abbildung des Zustandsvektors \mathbf{x} mithilfe der Messgleichung $g(\cdot)$ zuzüglich eines Messrauschvektors \mathbf{w} :

$$\mathbf{y}_k = g_k(\mathbf{x}_k) + \mathbf{w}_k. \quad (5.98)$$

Mit der partiellen Ableitung $\mathbf{C}_k = \frac{\partial g(\mathbf{x}_k)}{\partial \mathbf{x}_k}$ aus Gleichung (5.28) ergibt sich die linearisierte Form:

$$\mathbf{y}_k = \mathbf{C}_k \mathbf{x}_k + \mathbf{w}_k. \quad (5.99)$$

Transformation des Zustandsvektors Die meisten Sensoren sind nicht im Ursprung des Eigenbewegungskoordinatensystems \mathbf{F}_{EME} montiert. Daher liefern sie die Messdaten nicht in \mathbf{F}_{EME} , sondern jeder Sensor in seinem spezifischen Sensorkoordinatensystem $\mathbf{F}_{\text{Sensor},i}$. Zur Erhöhung der Übersichtlichkeit wird hier ohne Beschränkung der Allgemeinheit (o.B.d.A.) das kartesische Koordinatensystem verwendet, und der Index i entfällt.

Zunächst ist eine Transformation der relevanten Komponenten des Zustandsvektors \mathbf{x} von \mathbf{F}_{EME} nach $\mathbf{F}_{\text{Sensor}}$ erforderlich, beispielsweise für die globale Position des Sensors in ebenen Koordinaten:

$${}^0\mathbf{P}_{\text{Sensor}} = {}^0\mathbf{P}_{\text{EME}} \cdot \text{out}({}^{\text{EME}}\mathbf{P}_{\text{Sensor}}). \quad (5.100)$$

Die 6-DOF-Pose ${}^0\mathbf{P}_{\text{EME}}$ beschreibt die Lage von \mathbf{F}_{EME} ggü. \mathbf{F}_0 und wird aus dem Zustandsvektor befüllt (mindestens ${}^0x, {}^0y, {}^0\Psi$). Damit ist sie zeitvariant. Der Operator $\text{out}(\mathbf{P})$ bezeichnet das Heraustransformieren, siehe Gleichung (A.13).

Wie in Abschnitt 5.2.3.2 angedeutet, ist ${}^{\text{EME}}\mathbf{P}_{\text{Sensor}}$ im Regelfall zeitinvariant mit Ausnahme der auf MarVEye-8 montierten Kameras, für die eine zeitvariante HTM bzw. Pose bereitgestellt wird.

Ist allerdings die extrinsische stationäre Einbaulage eines Sensors nicht mit hinreichender Genauigkeit bekannt, so kann sie als Teil des sensorspezifischen Teilzustandsvektors mitgeschätzt werden.

5.7.2 Messung von Einzelradgeschwindigkeiten

Für die Geschwindigkeitsmessung kommen bei modernen Radfahrzeugen im Regelfall vier Drehzahlsensoren zum Einsatz – einer pro Rad (vgl. Abschnitt 3.2.3). Vom

Basisfahrzeug wird unter Verwendung eines als fest angenommenen Raddurchmessers eine Geschwindigkeit über Grund berechnet, welche allerdings von der realen Geschwindigkeit abweichen kann – bspw. durch Schlupf oder Verschleiß.

Der sensorspezifische Teilzustandsvektor $\mathbf{x}^{\text{PerWheelSpeed}}$ enthält daher einen Skalierungsfaktor s (welcher mit 1 initialisiert wird):

$$\mathbf{x}^{\text{PerWheelSpeed}} = \left[s^{\text{PerWheelSpeed}} \right]. \quad (5.101)$$

Das zugehörige Messmodell prädiziert unter der Starrkörperannahme und mithilfe der bekannten Radpositionen und des Satzes von Euler Gleichung (5.42) die vier Geschwindigkeiten:

$$\mathbf{y}^{*,\text{PerWheelSpeed}} = \begin{bmatrix} v_{[FL]}^* \\ v_{[FR]}^* \\ v_{[RL]}^* \\ v_{[RR]}^* \end{bmatrix} = g_i^{\text{PerWheelOdom}} \left(\begin{bmatrix} \mathbf{x}_k^{\text{Fahrzeug}} \\ \mathbf{x}_k^{\text{PerWheelSpeed}} \end{bmatrix}, \mathbf{q}^{\text{PerWheelSpeed}} \right). \quad (5.102)$$

Hier wird die Berechnung beispielhaft für das linke Vorderrad gezeigt, die weiteren Räder folgen analog:

$$v_{[FL]}^{*,\text{PerWheelSpeed}} = \left(\begin{bmatrix} \text{veh}v \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \times \begin{bmatrix} \mathbf{p}_{[x]} \\ \mathbf{p}_{[y]} \\ \mathbf{p}_{[z]} \end{bmatrix} \right) s^{\text{PerWheelSpeed}}. \quad (5.103)$$

$[\mathbf{p}_{[x]}, \mathbf{p}_{[y]}, \mathbf{p}_{[z]}]^T$ beschreibt dabei die relative Lage des vorderen linken Einzelradgeschwindigkeitssensors $\mathbf{F}_{\text{Sensor},[FL]}^{\text{PerWheelSpeed}}$ im Eigenbewegungskoordinatensystem \mathbf{F}_{EME} .

In vielen Fahrzeugen liefern die Geschwindigkeitssensoren keine vorzeichenbehaftete Messung des Zustands, sondern nur einen Betrag $|v|$. In der Messgleichung ist dies einfach zu modellieren, sorgt aber bei der analytischen Bestimmung der Jacobimatrix aufgrund von Fallunterscheidungen für wachsende Komplexität.

5.7.3 Weitere Geschwindigkeitsmessungen

Steht nur eine gemittelte Geschwindigkeit zur Verfügung, bspw. das Tachosignal, so wird angenommen, dass es sich um eine mittlere Geschwindigkeit aller Räder handelt, und die Sensorposition $\mathbf{F}_{\text{Sensor}}^{\text{AvgSpeed}}$ wird mittig zwischen den Achsen angenommen. Für den sensorspezifischen Teilzustandsvektor gilt:

$$\mathbf{x}^{\text{AvgSpeed}} = \left[s^{\text{AvgSpeed}} \right]. \quad (5.104)$$

Die Berechnung von $\mathbf{y}^{*,\text{AvgSpeed}}$ erfolgt analog zu Gleichung (5.103).

Ist bei Geländefahrzeugen mit Schlupf zu rechnen, so kommen oftmals optische oder radarbasierte Sensoren zur Messung der Bewegung gegenüber dem Untergrund zum Einsatz. Optische Sensoren ermitteln analog zu einer optischen Maus einen 2D-Geschwindigkeitsvektor.

Die hier verwendeten Radar-Sensoren ermitteln mittels des Dopplereffekts (vgl. Abschnitt 3.5.3) eine skalare Geschwindigkeit. Für den sensorspezifischen Teilzustandsvektor gilt:

$$\mathbf{x}^{\text{OvergroundSpeed}} = \begin{bmatrix} s^{\text{OvergroundSpeed}} \end{bmatrix}. \quad (5.105)$$

Die Berechnung von $\mathbf{y}^{*,\text{OvergroundSpeed}}$ erfolgt erneut analog zu Gleichung (5.103).

5.7.4 Messung des Lenkwinkels

Ein Lenkwinkelsensor (LWS) kommt zur Messung des Lenkwinkels am fiktiven mittleren Rad des Einspurmodells zum Einsatz.

Bei den verwendeten Versuchsträgern aus Kapitel 3 ist kein Sensor zur direkten Messung des Lenkwinkels bzw. des Radeinschlagwinkels vorhanden. Stattdessen kommt im DbW-Kit integrierte Sensorik zur Messung des Lenkradwinkels zum Einsatz. Die Transformation vom Lenkradwinkel auf den Lenkwinkel λ unter Berücksichtigung einer vom Hersteller gelieferten oder experimentell ermittelten Kennlinie findet bereits im Low-Level-Rechner statt und ist nicht Teil des Messmodells.

Die Kennlinie der Lenkung kann als nahezu linear angenommen werden; da sie dem System allerdings nicht bekannt ist, wird vereinfachend von einem konstanten Messfehler (Offset, mit 0 initialisiert) und einem Skalierungsfaktor (mit 1 initialisiert) ausgegangen. Bei einem idealen spielfreien System kann die Standardabweichung des Messfehlers auf Basis der Auflösung des Winkelsensors bestimmt werden. In der Realität ist immer Spiel vorhanden, daher wird das Rauschen entsprechend höher modelliert. Möglicher Raum zur Verbesserung besteht daher in einer expliziten Modellierung von Spiel in der Lenkung.

Der sensorspezifische Teilzustandsvektor \mathbf{x}^{LWS} enthält Offset b^{LWS} und Skalierungsfaktor s^{LWS} des Lenkwinkelsensors:

$$\mathbf{x}^{\text{LWS}} = \begin{bmatrix} b^{\text{LWS}} \\ s^{\text{LWS}} \end{bmatrix}. \quad (5.106)$$

Damit ergibt sich für die Prädiktion der Messung allgemein:

$$\mathbf{y}^{*,\text{LWS}} = g^{\text{LWS}} \left(\begin{bmatrix} \mathbf{x}^{\text{Fahrzeug}} \\ \mathbf{x}^{\text{LWS}} \end{bmatrix}, \mathbf{q}^{\text{LWS}} \right). \quad (5.107)$$

Ist der Lenkwinkel λ_{veh} Teil des Zustandsvektors $\mathbf{x}^{\text{Fahrzeug}}$, so beschreibt das Messmodell $g^{\text{LWS}}(\cdot)$ eine Abbildung mit Offset und Skalierungsfaktor:

$$\mathbf{y}^{*,\text{LWS}} = \lambda^{*,\text{LWS}} = (\lambda_{\text{veh}} + b^{\text{LWS}}) s^{\text{LWS}}. \quad (5.108)$$

Ist der Lenkwinkel λ_{veh} nicht Teil des Zustandsvektors, so kann die Messung $\lambda^{*,\text{LWS}}$ auf Basis der Gierrate ${}^0\dot{\Psi}$, der Geschwindigkeit ${}^{\text{veh}}v$ und des Achsabstands l_f des Fahrzeugs prädiert werden:

$$\mathbf{y}^{*,\text{LWS}} = \lambda^{*,\text{LWS}} = \left(\arctan2\left({}^0\dot{\Psi} l_f, {}^{\text{veh}}v\right) + b^{\text{LWS}} \right) s^{\text{LWS}} \quad \text{für } {}^{\text{veh}}v \neq 0. \quad (5.109)$$

5.7.5 Messung mit Inertialsensoren

Eine IMU besteht im Regelfall aus jeweils drei orthogonal zueinander stehenden Drehratensensoren und Beschleunigungssensoren. Diese Sensoren messen Beschleunigungen und Drehraten des körperfesten Sensorkoordinatensystems $\mathbf{S}_{\text{Sensor}}$ bezüglich des Inertialkoordinatensystems $\mathbf{F}_{\text{inertial}}$ (vgl. Abschnitt 5.2.2.1). Eine in der IMU integrierte Elektronik sorgt – unter der Hinzunahme von weiteren Sensoren wie z. B. einem 3-Achsen-Magnetometer oder Temperatursensoren – für eine erste Verarbeitung der Messwerte. Im Regelfall umfasst diese Vorverarbeitung die Korrektur von Skalierungs- und Offsetfehlern, zusätzlich berechnen die meisten IMUs eine Schätzung der Lage des Sensors ggü. \mathbf{F}_0 . An dieser Stelle soll nicht vertiefend auf das Messprinzip einer IMU eingegangen werden, stattdessen steht die Verwendung ihrer Ausgangssignale als Messung des Fahrzeugzustandes im Vordergrund.

Die Messungen der IMU lassen sich in drei Teile aufsplitten: Winkel, Winkelgeschwindigkeiten und Beschleunigungen. IMU-relevante Messmodelle und zugehörige Teilzustandsvektoren werden in den nächsten Absätzen vertieft.

Der sensorspezifische Teilzustandsvektor \mathbf{x}^{IMU} setzt sich daher aus drei Teilen zusammen:

$$\mathbf{x}^{\text{IMU}} = \begin{pmatrix} \mathbf{x}^{\text{IMU,Orientation}} \\ \mathbf{x}^{\text{IMU,Rate}} \\ \mathbf{x}^{\text{IMU,Acceleration}} \end{pmatrix}. \quad (5.110)$$

Winkel Ein Fahrzeug kann vereinfachend als starrer Körper aufgefasst werden. Winkel sind zu jedem Zeitpunkt an jeder Position des Fahrzeugs identisch, solange die betrachteten Koordinatensysteme \mathbf{F}_{EME} und $\mathbf{F}_{\text{Sensor}}$ achsenparallel sind. Handelt es sich bei der Transformation vom Eigenbewegungskoordinatensystem \mathbf{F}_{EME} ins Sensorkoordinatensystem $\mathbf{F}_{\text{Sensor}}$ um eine reine Translation, misst ein Inertialsensor direkt den Zustand. Ist das Sensorkoordinatensystem $\mathbf{F}_{\text{Sensor}}$ allerdings gegenüber \mathbf{F}_{EME} zusätzlich rotiert, so müssen die Winkel in $\mathbf{F}_{\text{Sensor}}$ transformiert werden.

Um Fehler in der extrinsischen Einbaulage des IMU-Sensors gegenüber dem fahrzeugfesten Eigenbewegungskoordinatensystem \mathbf{F}_{EME} zu kompensieren, wird diese

6-DOF-Pose in den sensorspezifischen Teilzustandsvektor $\mathbf{x}^{\text{IMU,Orientation}}$ aufgenommen, die Initialisierung erfolgt mit der a-priori bekannten extrinsischen Kalibrierung:

$$\mathbf{x}^{\text{IMU,Orientation}} = \text{Sensor} \mathbf{P}_{\text{EME}} \quad (5.111)$$

$$= \left(\mathbf{P}_{\text{EME},[x]}, \mathbf{P}_{\text{EME},[y]}, \mathbf{P}_{\text{EME},[z]}, \mathbf{P}_{\text{EME},[\Phi]}, \mathbf{P}_{\text{EME},[\Theta]}, \mathbf{P}_{\text{EME},[\Psi]} \right)^T. \quad (5.112)$$

Damit ergibt sich für die Prädiktion der Lagemessung:

$$\mathbf{y}^{*,\text{IMU,Orientation}} = \mathbf{g}^{\text{IMU,Orientation}} \left(\begin{bmatrix} \mathbf{x}^{\text{Fahrzeug}} \\ \mathbf{x}^{\text{IMU}} \end{bmatrix}, \mathbf{q}^{\text{IMU}} \right) \quad (5.113)$$

$$= \begin{bmatrix} 0 \Phi^{\text{IMU}} \\ 0 \Theta^{\text{IMU}} \\ 0 \Psi^{\text{IMU}} \end{bmatrix}^*. \quad (5.114)$$

Die drei Euler-Winkel beschreiben die Rotation von \mathbf{F}_0 nach $\mathbf{F}_{\text{Sensor}}$, in Anhang A.1 findet sich die Definition der Rotationsreihenfolge.

Winkelgeschwindigkeiten Analog zur Bestimmung der Winkel mit einer IMU ist die Starrkörperannahme auch für die Bestimmung der Winkelgeschwindigkeiten zu treffen. Ist das Sensorkoordinatensystem $\mathbf{F}_{\text{Sensor}}$ gegenüber \mathbf{F}_{EME} zusätzlich rotiert, so müssen die Winkelgeschwindigkeiten von \mathbf{F}_{EME} nach $\mathbf{F}_{\text{Sensor}}$ transformiert werden. Bei einer reinen Translation misst ein Inertialsensor direkt den Zustand, sodass auf die Transformation verzichtet werden kann.

Eine IMU misst Beschleunigungen und Drehraten des körperfesten Sensorkoordinatensystems $\mathbf{F}_{\text{Sensor}}$ gegenüber dem Inertialkoordinatensystem $\mathbf{F}_{\text{inertial}}$. Betrachtet man einen einzelnen Drehratensensor¹⁰, so misst dieser die Summe der Drehbewegung des Sensors gegenüber der statischen Welt und einer aus der Erdrotation resultierenden Komponente von $\omega_0 \approx 15,04^\circ/\text{h}$ (vgl. Gleichung (5.2)):

$$\omega^{\text{Sensor}} = \mathbf{g}(\mathbf{x}^{\text{Fahrzeug}}, \omega_0). \quad (5.115)$$

Höherpreisige Sensoren wie das INS vom Typ OxTS RT3003v1 geben korrigierte Messwerte aus, die bereits von dieser Komponente der Erdrotation befreit wurden [Oxford Technical Solutions Ltd., 2020, S. 103f.].

Low-cost-Sensoren, wie sie in Kraftfahrzeugen oder auch im INS vom Typ Xsens MTi eingesetzt werden, haben eine zu geringe Empfindlichkeit und sind somit nicht in der Lage, den Einfluss der Erdrotation vom reinen Sensorrauschen zu unterscheiden. Die Bias-Stabilität des einzelnen Drehratensensors beträgt $20^\circ/\text{h}$

¹⁰Bei einigen Funktionsprinzipien von MEMS-Drehratensensoren wird der bei Rotation auftretenden Coriolis-Kraft-Effekt zur Messung der Rotationsgeschwindigkeit genutzt [Marek, 2007]. Faseroptische Kreisel, engl. *Fibre-Optic Gyroscopes* (FOG) nutzen den Sagnac-Effekt, der bei Rotation für unterschiedliche Laufzeiten von gegenläufig durch eine Glasfaser spule geführtem Laserlicht führt, welche mittels eines Interferenzdetektors erfasst werden [von Hinüber, 2002].

[Xsens Technologies B.V., 2010b, S. 5]. Für den Gesamtsensor wird die Bias-Stabilität je nach Baujahr unterschiedlich mit $1\sigma = 1^\circ/\text{s}$ [Xsens Technologies B.V., 2010c, S. 16] bzw. $1\sigma = 5^\circ/\text{s}$ [Xsens Technologies B.V., 2007, S. 12] angegeben.

Aus diesen beiden Gründen kann im Messmodell auf die Modellierung des Einflusses der Erdrotation verzichtet werden.

Der sensorspezifische Teilzustandsvektor $\mathbf{x}^{\text{IMU,Rate}}$ enthält zur Kompensation von Sensorfehlern für jede der drei Achsen einen Offset b :

$$\mathbf{x}^{\text{IMU,Rate}} = \begin{pmatrix} b_{\omega_x} \\ b_{\omega_y} \\ b_{\omega_z} \end{pmatrix}. \quad (5.116)$$

Damit ergibt sich für die Prädiktion der Winkelgeschwindigkeits- oder Drehratenmessung:

$$\mathbf{y}^{*,\text{IMU,Rate}} = g^{\text{IMU,Rate}} \left(\begin{bmatrix} \mathbf{x}^{\text{Fahrzeug}} \\ \mathbf{x}^{\text{IMU}} \end{bmatrix}, \mathbf{q}^{\text{IMU}} \right) \quad (5.117)$$

$$= \begin{bmatrix} \omega_{[x]}^{\text{IMU}} \\ \omega_{[y]}^{\text{IMU}} \\ \omega_{[z]}^{\text{IMU}} \end{bmatrix}^* = \begin{pmatrix} b_{\omega_x} \\ b_{\omega_y} \\ b_{\omega_z} \end{pmatrix} + \text{rot}(\text{Sensor} \mathbf{P}_{\text{EME}}) \begin{pmatrix} 0 \dot{\Phi} \\ 0 \dot{\Theta} \\ 0 \dot{\Psi} \end{pmatrix}. \quad (5.118)$$

Die Messungen der Winkelgeschwindigkeiten um die Achsen von $\mathbf{F}_{\text{Sensor}}$ werden mit $\omega_{[x,y,z]}$ bezeichnet und unterscheiden sich damit von der Bezeichnung der Fahrzeugzustände $\dot{\Phi}$, $\dot{\Theta}$ und $\dot{\Psi}$.

Beschleunigungen Die IMU misst drei Beschleunigungen $a_{[x,y,z]}^{\text{IMU}}$ in Richtung der drei Achsen von $\mathbf{F}_{\text{Sensor}}$. Die gemessene Beschleunigung setzt sich aus vier Komponenten zusammen:

- \mathbf{a}_{long} : longitudinale Beschleunigung des Fahrzeugs $^{\text{veh}}a$,
- \mathbf{a}_{lat} : durch die Zentripetalkraft verursachte Lateralbeschleunigung,
- $\mathbf{a}_{\text{gravity}}$: Einfluss der Erdbeschleunigung,
- $\mathbf{a}_{\text{offset}}$: Offsets der Einzelsensoren.

Die aus der Erdrotation resultierende Komponente der Corioliskraft wird hier aufgrund der zu geringen Empfindlichkeit der Beschleunigungssensoren in den verwendeten IMUs vernachlässigt [Dorobantu, 1999].

Höherpreisige Sensoren wie das INS vom Typ OxTS RT3003v1 geben – analog zu den Winkelgeschwindigkeiten – korrigierte Messwerte aus, die bereits vom Einfluss der Erdbeschleunigung befreit und deren Offsets sensorintern kompensiert wurden [Oxford Technical Solutions Ltd., 2020, S. 103f.].

Zusätzlich ist bei Sensoreinbaupositionen $\mathbf{F}_{\text{Sensor}} \neq \mathbf{F}_{\text{EME}}$ zu berücksichtigen, dass sich die in \mathbf{F}_{EME} geschätzten Winkelgeschwindigkeiten und Winkelbeschleunigungen

linear mit der Entfernung auf die prädizierte Beschleunigungsmessung in $\mathbf{F}_{\text{Sensor}}$ auswirken (vgl. Gleichung (5.43)).

Der sensorspezifische Teilzustandsvektor $\mathbf{x}^{\text{IMU,Acceleration}}$ enthält zur Kompensation von Sensorfehlern für jede der drei Achsen einen Offset b :

$$\mathbf{x}^{\text{IMU,Acceleration}} = \begin{pmatrix} b_{a_x} \\ b_{a_y} \\ b_{a_z} \end{pmatrix}. \quad (5.119)$$

Zuerst wird die Beschleunigungsvektor ${}^{\text{EME}}\mathbf{a}$ in \mathbf{F}_{EME} bestimmt:

$${}^{\text{EME}}\mathbf{a} = \mathbf{a}_{\text{long}} + \mathbf{a}_{\text{lat}} + \mathbf{a}_{\text{gravity}}, \quad (5.120)$$

mit

$$\mathbf{a}_{\text{long}} = \begin{pmatrix} {}^{\text{veh}}a & 0 & 0 \end{pmatrix}^{\text{T}}, \quad (5.121)$$

$$\mathbf{a}_{\text{lat}} = \begin{pmatrix} 0 & {}^{\text{veh}}v \cdot {}^0\dot{\Psi} & 0 \end{pmatrix}^{\text{T}}, \quad (5.122)$$

$$\mathbf{a}_{\text{gravity}} = \text{rot}({}^0\Phi, {}^0\Theta, 0) \begin{pmatrix} 0 & 0 & g \end{pmatrix}^{\text{T}}. \quad (5.123)$$

Anschließend erfolgt nach Gleichung (5.43) die Bestimmung von ${}^{\text{Sensor||EME}}\mathbf{a}$ am Montagepunkt des Sensors, die noch achsenparallel zu \mathbf{F}_{EME} ist.

Final erfolgt die Rotation in das Sensorkoordinatensystem, und es ergibt sich die Prädiktion der Beschleunigungsmessung:

$$\mathbf{y}^{*,\text{IMU,Acceleration}} = g^{\text{IMU,Acceleration}} \left(\begin{bmatrix} \mathbf{x}^{\text{Fahrzeug}} \\ \mathbf{x}^{\text{IMU}} \end{bmatrix}, \mathbf{q}^{\text{IMU}} \right) \quad (5.124)$$

$$= \begin{bmatrix} a_{[x]}^{*,\text{IMU}} \\ a_{[y]}^{*,\text{IMU}} \\ a_{[z]}^{*,\text{IMU}} \end{bmatrix} \quad (5.125)$$

$$= \text{rot}({}^{\text{EME}}\mathbf{P}_{\text{Sensor}}) {}^{\text{Sensor||EME}}\mathbf{a} + \mathbf{a}_{\text{offset}}. \quad (5.126)$$

5.7.6 Messung mittels GNSS

GNSS-Sensoren liefern als Positionsmessung eine geografische Länge und Breite im erdfesten Koordinatenreferenzsystem \mathbf{Y}_{LLH} sowie die ellipsoidische Höhe h gegenüber dem Referenzellipsoid WGS 84. Zusätzlich wird die Geschwindigkeit am Ort der GNSS-Antenne gemessen und in Navigationskoordinaten bereitgestellt (meist in der NED-Darstellung, vgl. Abschnitt 5.2.4): $v_{[\text{North}]}$ in Richtung Nordpol, $v_{[\text{East}]}$ achsenparallel zur Äquatorebene und $v_{[\text{Down}]}$ entlang der Normalen des Ellipsoiden.

Um als Messung für die Eigenbewegungsschätzung verwendet zu werden, werden diese Werte in ein anderes Koordinatensystem transformiert: Die Position wird vom

erdfesten Koordinatensystem \mathbf{Y}_{LLH} auf das ebene Koordinatensystem \mathbf{F}_{EME} abgebildet, dabei kommen UTM-Koordinaten als Zwischenschritt zum Einsatz. Die in Navigationskoordinaten gemessene Geschwindigkeit muss dazu um die z_0 -Achse rotiert werden. Der Rotationswinkel entspricht dem Winkel zwischen GNSS-Breitengrad und UTM-Ostachse (bzw. dem Längengrad und UTM-Nordachse) an der aktuellen Fahrzeugposition, wie in Abbildung 5.12 gezeigt. Zusätzlich muss die Höhe ggf. um die Undulation (vgl. Gleichung (5.1)) korrigiert werden, damit Höhenmodell in Zustand und Sensormessung korrespondieren (entweder geographische Höhe oder orthometrische Höhe).

Damit ergibt sich \mathbf{y}^{GNSS} :

$$\mathbf{y}^{GNSS} = \begin{pmatrix} \mathbf{y}_{[Position]}^{GNSS} \\ \mathbf{y}_{[Geschwindigkeit]}^{GNSS} \end{pmatrix} = \begin{pmatrix} {}^0x = \text{Ost [m]} \\ {}^0y = \text{Nord [m]} \\ {}^0z = \text{Höhe [m]} \\ {}^0\dot{x} = \text{Geschwindigkeit Ost } \left[\frac{\text{m}}{\text{s}} \right] \\ {}^0\dot{y} = \text{Geschwindigkeit Nord } \left[\frac{\text{m}}{\text{s}} \right] \\ {}^0\dot{z} = \text{Geschwindigkeit Höhe } \left[\frac{\text{m}}{\text{s}} \right] \end{pmatrix}. \quad (5.127)$$

GNSS-Positionsmessung Die prädizierte Positionsmessung ergibt sich aus der Transformation der Positions-Zustandsgrößen von \mathbf{S}_{EME} an den Ort der GNSS-Antenne (als Spezialisierung von Gleichung (5.100)):

$${}^0\mathbf{P}_{\text{Sensor}} = {}^0\mathbf{P}_{EME.out}({}^{EME}\mathbf{P}_{\text{Sensor}}), \quad (5.128)$$

$$\mathbf{y}_{[Position]}^{*,GNSS} = \begin{pmatrix} x_0^{GNSS} \\ y_0^{GNSS} \\ z_0^{GNSS} \end{pmatrix} = \begin{pmatrix} {}^0\mathbf{P}_{\text{Sensor},[x]} \\ {}^0\mathbf{P}_{\text{Sensor},[y]} \\ {}^0\mathbf{P}_{\text{Sensor},[z]} \end{pmatrix}. \quad (5.129)$$

Stehen mehrere GNSS-Antennen zur Verfügung und liefern Messungen jeweils mit Zentimeter-Genauigkeit, könnte daraus bereits im Stand das Heading bzw. der Gierwinkel Ψ_0 bestimmt werden – allerdings bei weitem nicht in der Sub-Grad-Genauigkeit wie es mit Zwei-Antennen-Empfängern bestimmt werden kann.

GNSS-3D-Geschwindigkeitsmessung Die Geschwindigkeit an der Montageposition des Sensors, angegeben in der Orientierung von \mathbf{S}_{EME} , ergibt sich analog zu Gleichung (5.42) aus den translatorischen und rotatorischen Geschwindigkeits-Zustandsgrößen sowie der Relativposition:

$$\begin{pmatrix} \dot{x}^{GNSS} \\ \dot{y}^{GNSS} \\ \dot{z}^{GNSS} \end{pmatrix}_{EME} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix}_{EME} + \begin{pmatrix} \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \end{pmatrix}_{EME} \times \begin{pmatrix} \mathbf{P}_{[x]}^{GNSS} \\ \mathbf{P}_{[y]}^{GNSS} \\ \mathbf{P}_{[z]}^{GNSS} \end{pmatrix}. \quad (5.130)$$

$[\mathbf{P}_{[x]}, \mathbf{P}_{[y]}, \mathbf{P}_{[z]}]^T$ beschreibt dabei die relative Lage der GNSS-Antenne $\mathbf{F}_{\text{Sensor}}^{GNSS}$ im Eigenbewegungskoordinatensystem \mathbf{F}_{EME} und ergibt sich aus ${}^{\text{Sensor}}\mathbf{H}_{EME}$.

Liegt \mathbf{F}_{EME} im Hinterachsmittelpunkt, so existiert nur eine Geschwindigkeit in x -Richtung:

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix}_{\text{EME}} = \begin{pmatrix} v^{\text{veh}} \\ 0 \\ 0 \end{pmatrix}_{\text{EME}}. \quad (5.131)$$

Mit der Transformation in das ebene Koordinatensystem \mathbf{S}_0 ergibt sich die prädierte Geschwindigkeitsmessung für die GNSS-Antenne:

$$\mathbf{y}_{[\text{Geschwindigkeit}]}^{*,\text{GNSS}} = \begin{pmatrix} \dot{x}^{\text{GNSS}} \\ \dot{y}^{\text{GNSS}} \\ \dot{z}^{\text{GNSS}} \end{pmatrix}_0 = {}^0\mathbf{R}_{\text{EME}} \begin{pmatrix} \dot{x}^{\text{GNSS}} \\ \dot{y}^{\text{GNSS}} \\ \dot{z}^{\text{GNSS}} \end{pmatrix}_{\text{EME}}. \quad (5.132)$$

GNSS-2D-Geschwindigkeitsmessung Je nach verwendetem Sensortyp ist die gelieferte Aufteilung der Geschwindigkeit in drei Komponenten stark fehlerbehaftet. Für diese Sensoren ist es sinnvoll, nur den Betrag des Geschwindigkeitsvektors in der x - y -Ebene des ebenen Koordinatensystems \mathbf{S}_0 zu betrachten. Dafür muss das Messmodell angepasst werden:

Die Berechnung des Geschwindigkeitsvektors der GNSS-Antenne in \mathbf{S}_0 erfolgt nach Gleichung (5.132), daraus ergibt sich die euklidische Norm:

$$v^{\text{GNSS}} = \left\| \begin{pmatrix} \dot{x}^{\text{GNSS}} \\ \dot{y}^{\text{GNSS}} \\ 0 \end{pmatrix}_0 \right\|. \quad (5.133)$$

5.8 Bereitgestellte Ergebnisse

Die Ergebnisse werden für die Verwendung in anderen Softwaremodulen immer vom Eigenbewegungskoordinatensystem \mathbf{F}_{EME} , welches sich je nach Prozessmodell an unterschiedlichen Stellen im Fahrzeug befinden kann, in das Ego-Fahrzeugkoordinatensystem \mathbf{F}_{veh} transformiert (vgl. Gleichungen (5.41) bis (5.43)).

Dies ist für die Positionsschätzung problemlos machbar. Die Geschwindigkeits- und Beschleunigungsvektoren werden jedoch gestört, da sich hier das Rauschen in den Winkelgeschwindigkeiten und Winkelbeschleunigungen proportional zum Hebelarm auswirkt.

Diese Warnung wird auch von INS-Herstellern geäußert:

Note that the noise in the acceleration outputs will be much higher when Output Displacement is used. Typical installations in moving vehicles have angular vibrations of about 2 rad/s^2 . This equates to 2 m/s^2 of additional vibration with a one metre output displacement. It will be necessary to filter the data if Output Displacement is used. (Oxford Technical Solutions Ltd. [2020, S. 75])

Daher ist eine geschickte Wahl der Koordinatensysteme wichtig, vor allem die Lage des Ego-Fahrzeugkoordinatensystems \mathbf{F}_{veh} ggü. dem Eigenbewegungskoordinatensystem \mathbf{F}_{EME} : ideal sind identische Ursprünge der beiden Koordinatensysteme; wo dies nicht möglich ist, sollten zumindest die Abstände minimiert werden.

Für alle bereitgestellten Daten der Eigenbewegungsschätzung gilt eine Updatefrequenz von mindestens 100 Hz.

5.8.1 Global korrigierte Pose

Die Eigenbewegungsschätzung, engl. *Ego-Motion Estimation* (EME) stellt eine bestmögliche globale Positions-, Lage- und Dynamikschätzung des Ego-Fahrzeugs für andere Softwaremodule bereit, ist jedoch nicht garantiert sprungfrei.

Die Pose ${}^0\mathbf{P}_{\text{veh}}^{\text{EME}}$ beschreibt die Lage des Fahrzeugreferenzpunktes (veh) gegenüber dem ebenen Koordinatensystem \mathbf{F}_0 (welches nach Gleichung (5.9) identisch zu \mathbf{F}_{UTM} ist) unter Verwendung der EME.

Um die globale Ausrichtung ${}^0\Psi$ während der Initialisierung bestimmen zu können, bestehen zwei Möglichkeiten: Falls ein Dual-Antennen-GNSS-Empfänger vorhanden ist, kann dieser im Stand eine Ausrichtung bestimmen. Dies ist bspw. bei den Novatel-Empfängern aus der OxTS RT3003 v1 und Nachfolgeräten der Fall.

Steht solch ein Empfänger nicht zur Verfügung, kann durch gezielte Fahrmanöver eine Initialisierung von ${}^0\Psi$ erfolgen, bewährt hat sich hier eine Geradeausfahrt über eine gewisse Strecke mit einer Mindestgeschwindigkeit.

5.8.2 Dead-Reckoning-Pose

Die Hauptanforderung an die Dead-Reckoning-Pose ist ein sprungfreies System, welches sofort bei Start der Algorithmen ohne Initialisierung zur Verfügung steht. Hier wird ein Wegdriften von der wahren Position in Kauf genommen.

Die Positionsbestimmung erfolgt auf Basis der bestmöglichen Schätzung der Geschwindigkeit, des Lenkwinkels und/oder der Winkelgeschwindigkeiten, deren Offset- und Skalierungsfehler vom Filter kompensiert werden, sowie der Roll- und Nickwinkel des Fahrzeugs.

Im monolithischen Eigenbewegungsschätzer (vgl. Abschnitt 5.6.1) ist die Dead-Reckoning-Pose Teil des Zustandsvektors und wird über die Prozessmodelle integriert, aber nicht durch Messungen und die Innovationsschritte beeinflusst.

Dagegen wird im modularen Eigenbewegungsschätzer die Dead-Reckoning-Pose erst bei der Ausgabe in Richtung Middleware unter Verwendung des jeweiligen Prozessmodells und des aktuellen Zustandsvektors prädiert.

Die Pose ${}^0\mathbf{P}_{\text{veh}}^{\text{DR}}$ beschreibt die Lage des Fahrzeugreferenzpunktes gegenüber dem ebenen Koordinatensystem \mathbf{F}_0 unter Verwendung des Koppelnavigation (engl. *Dead Reckoning*) (DR).

Die Integration der Dead-Reckoning-Pose kann an jedem Startpunkt beginnen (z. B. $(0, 0, 0)^T$). Zur besseren Interpretation und Visualisierung hat es sich bewährt, sie mit der ersten globalen Positionsschätzung oder -messung zu initialisieren.

5.9 Umgebungsabhängige Besonderheiten

In bestimmten Situationen ist es für eine gute Lokalisierung besser, GNSS-basierte Messungen nicht zu verwenden, bspw. in engen bewaldeten Tälern oder Parkhäusern. Hier kommt es vor, dass ein GNSS-Empfänger anhand von vier sichtbaren Satelliten eine Positionslösung bestimmt, die aber allein auf ungültigen Messungen beruht, beispielsweise aufgrund von Signalreflexion bzw. Multipath-Effekten. Halten diese Spiegelungen länger an, „glaubt“ das Filter in einem Loosely-Coupled INS nach einer längeren Zeit diesen Messungen: da die Schätzfehlerkovarianz bei ausbleibenden Messungen anwächst, werden Falschmessungen irgendwann nicht mehr anhand der Innovationskovarianz aussortiert, sodass ab diesem Zeitpunkt auch die Positionslösung des INS verfälscht wird.

An dieser Stelle kann nun weiteres Wissen genutzt werden, um die Ungültigkeit von Messungen vorherzusagen: Über Karten ist menschengemachtes Vorwissen über Bereiche mit potentiell stark gestörtem GNSS-Empfang bekannt (z. B. Bewaldung aus Anhang B.2.3, dichte Bebauung, Tunnel, ...). Ebenso kann mithilfe von Kamerabildern oder LiDAR-Daten bestimmt werden, dass keine freie Sicht zum Himmel besteht.

Für die monolithische Eigenbewegungsschätzung aus Abschnitt 5.6.1 wurde eine Schnittstelle für die extern bestimmte Konfidenz implementiert, welche die Nutzung der vom GNSS beeinflussten Positionsinformation, der Geschwindigkeit und der Orientierung Ψ verhindert, sodass auch die globale Pose ${}^0\mathbf{P}_{\text{veh}}^{\text{EME}}$ temporär analog zum Dead-Reckoning bestimmt wird.

5.10 Ergebnisse Eigenbewegung

Im Folgenden werden die beiden entwickelten Varianten der Eigenbewegungsschätzung in einzelnen Szenarien analysiert.

5.10.1 Monolithischer Eigenbewegungsschätzer

Abbildung 5.17 zeigt die Verwendung des monolithischen Eigenbewegungsschätzers in MuCAR-3 auf der M-ELROB 2012 in Thun. Das *Approach*-Szenario fand auf der in Abb. 5.17a gezeigten Gelände-LKW-Fahrschulstrecke statt. Der zu bewältigende Parcours ging im Zick-Zack durch Gräben und über Wälle mit Steigungen von über

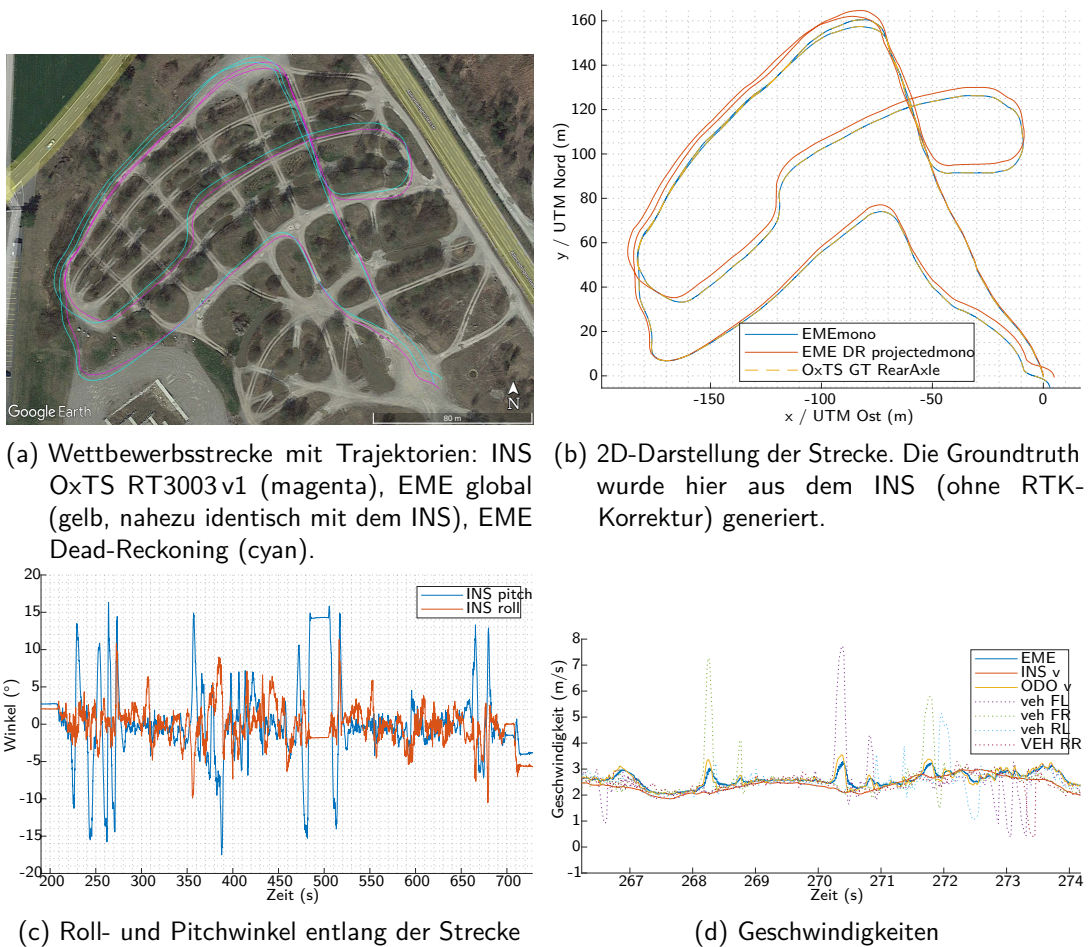


Abbildung 5.17:

Ergebnisse der monolithischen Eigenbewegungsschätzung und Lokalisierung auf der M-ELROB 2012. (a) zeigt das sehr wellige Fahrschulgelände, (b) die dazugehörigen 2D-Trajektorien. (c) zeigt Steigungen von über 15° resp. mehr als 25% . (d) zeigt die Geschwindigkeiten – hier ist gut zu erkennen, dass im Gelände einzelne Räder beim Bergauffahren durchdrehen (grün, lila) oder beim Bergabfahren stärker gebremst werden (bei $t = 273\text{s}$). Dieses wirkt sich auch auf die mittlere Fahrzeuggeschwindigkeit des Basisfahrzeugs (gelb) aus und erzeugt Fehler auf der geschätzten Geschwindigkeit (blau).

25 %, wie in Abb. 5.1 zu sehen ist. Für die Hinderniskartierung und Navigation wurde eine stabile 6-DOF-Dead-Reckoning-Pose benötigt.

Abbildung 5.17b zeigt das Resultat des Dead-Reckonings ggü. dem INS. Trotz der starken, in Abb. 5.17c gezeigten Roll- und Pitchbewegungen des Fahrzeugs gab es nur einen leichten Versatz im Gierwinkel, der über die Strecke zu wenigen Metern Abweichung führte. Zu leichten Abweichungen in der Länge führten die in Abb. 5.17d dargestellten falschen Geschwindigkeitsmessungen durch durchdrehende oder zu stark abgebremste Räder. Diese beeinflussten für kurze Zeiträume von unterhalb 1 s auch die Geschwindigkeitsschätzung und verursachten damit einen Längenfehler im Dead-Reckoning. Beides hatte keine Auswirkungen auf die lokalen Hinderniskarten und die autonome Navigation.

5.10.2 Dynamisch erzeugter modularer Eigenbewegungsschätzer

Der dynamisch erzeugte modulare Eigenbewegungsschätzer wurde primär für die beiden Bundeswehr-LKWs TULF und StrAsRob entwickelt, daher wird er auch auf Basis von Messdaten dieser Fahrzeuge ausgewertet. Tabelle 5.4 zeigt die für die Experimente verwendeten Standardabweichungen der Sensoren.

Abbildung 5.18 zeigt eine Analyse des Rauschens der Gierratensensoren von TULF und StrAsRob. Bei abgeschaltetem Motor rauscht der faseroptische 1-Achs-Kreisel wie erwartet mit wesentlich niedrigerer Amplitude als der MEMS-Kreisel in der Xsens IMU. Bei stehendem Fahrzeug mit laufendem Motor sind erste Vibrationen sichtbar. Dieser Effekt verstärkt sich während der Fahrt, bei der alle Sensoren eine ähnliche Amplitude im Rauschen zeigen – dieses wurde in den Sensoreigenschaften durch entsprechende Varianzen berücksichtigt.

Die im Folgenden analysierten Messfahrten fanden mit dem TULF in leicht hügeligem Gelände entlang von Schotter- und Sandwegen auf dem Standortübungsplatz (StÜb-

Tabelle 5.4:

Verwendete Standardabweichungen der Sensoren für den modularen Eigenbewegungsschätzer.

Sensortyp	Messung	Standardabweichung σ
Basisfahrzeug & DbW	mittlere Radgeschwindigkeit	0,02 m/s
	Einzelradgeschwindigkeit	0,02 m/s
	Lenkwinkel	1°
Übergrund-Radar	Geschwindigkeit	0,5 m/s
	Orientierung (Φ, Θ, Ψ)	1°
	Drehrate $\omega_{[x,y,z]}$	6°/s
IMU	Beschleunigung $a_{[x,y,z]}$	1,2 / 1,3 / 1,5 m/s ²
	Position (horizontal, vertikal)	2,5 / 2,5 m
GPS	Geschwindigkeit	0,3 m/s
1-Achs-Kreisel	Gierrate $\omega_{[z]}$	1°/s

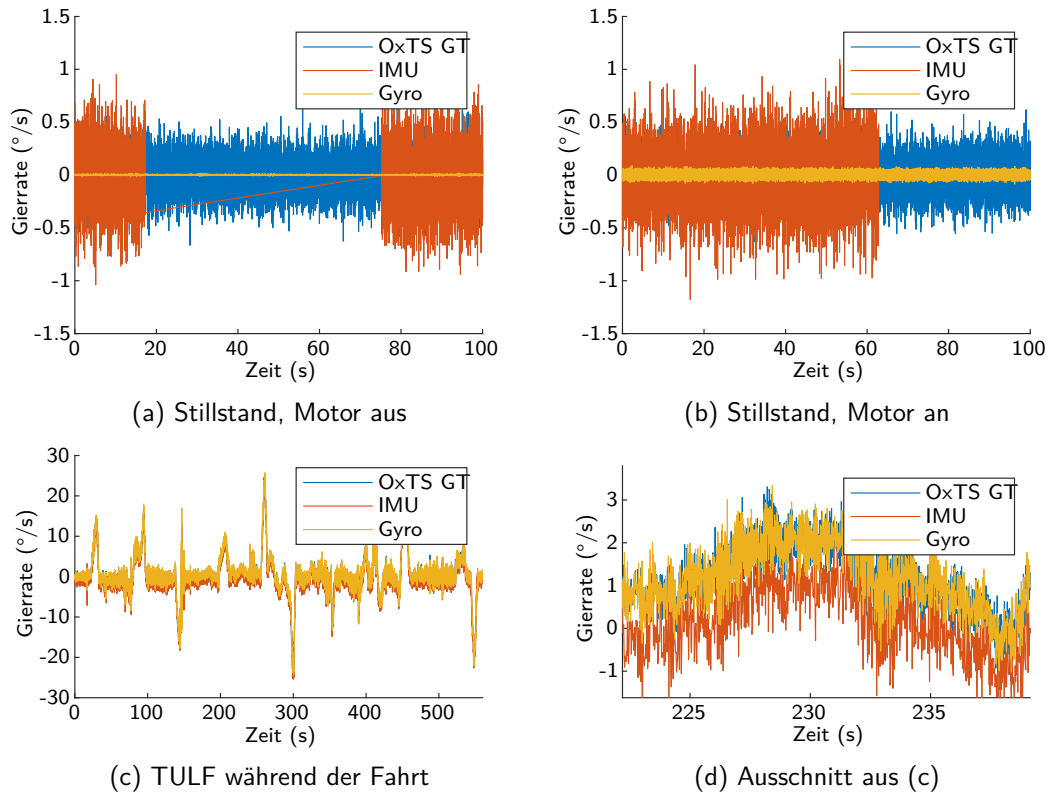


Abbildung 5.18:

(a) und (b) zeigen das Rauschen unterschiedlicher Gierratensoren bei stehendem Versuchsfahrzeug StrAsRob. Die Vibrationen des Motors sind nur im erhöhten Rauschen des faseroptischen 1-Achs-Kreisels sichtbar, bei den anderen MEMS-Sensoren gehen sie im Grundrauschen unter. Das INS vom Typ OxTS RT3003 v1 war als temporärer Ground-Truth-Sensor am Chassis montiert, die anderen Sensoren in der Sensorbay. (c) und (d) zeigen die gemessene Gierrate während der Fahrt mit dem TULF auf einer Schotterpiste auf dem StÜbPI Pfullendorf. Hier ist bis auf den kleineren Bias ggü. der Xsens IMU kaum mehr ein Vorteil des faseroptischen 1-Achs-Kreisels zu erkennen. Als Ground-Truth kommt hier das fest verbaute INS aus Abschnitt 3.5.3.2 zum Einsatz.

PI) Pfullendorf statt. Abbildung 5.19 zeigt die globale Lokalisierung auf Basis des modularen Eigenbewegungsschätzers ggü. der Ground Truth (inkl. RTK-Korrekturdaten) in vier unterschiedlichen Konfigurationen bzw. Sensor-Setups: einerseits mit oder ohne den faseroptischen 1-Achs-Kreisel, andererseits mit Einzelradgeschwindigkeiten oder der gemittelten Fahrzeuggeschwindigkeit. Die Ground-Truth-Pose wurde auf die Hinterachse projiziert, um besser mit der geschätzten Pose vergleichbar zu sein. Die dargestellte GNSS-Position entspricht der Antennenposition an der Fahrzeugfront und weicht aufgrund des „Ausholens“ insbesondere in Kurven stärker ab. Abbildung 5.19b zeigt einen Stopp nach durchdrehenden Rädern an einer Steigung – dort traten die größten Abweichungen auf. Hier ist zu erkennen, dass im Stillstand alle Positionsschätzungen in Richtung der GNSS-Messung gezogen werden.

In der globalen Lokalisierung wird hier mit Low-Cost-Sensorik eine Genauigkeit von 5 m, meist wesentlich besser, erreicht. Für Navigationsentscheidungen „an der folgenden Kreuzung nach rechts abbiegen“ ist dies genau genug (im Gegensatz zum Karten- und GNSS-gestütztes Fahren in den People-Movern, vgl. Abschnitt 2.3.10).

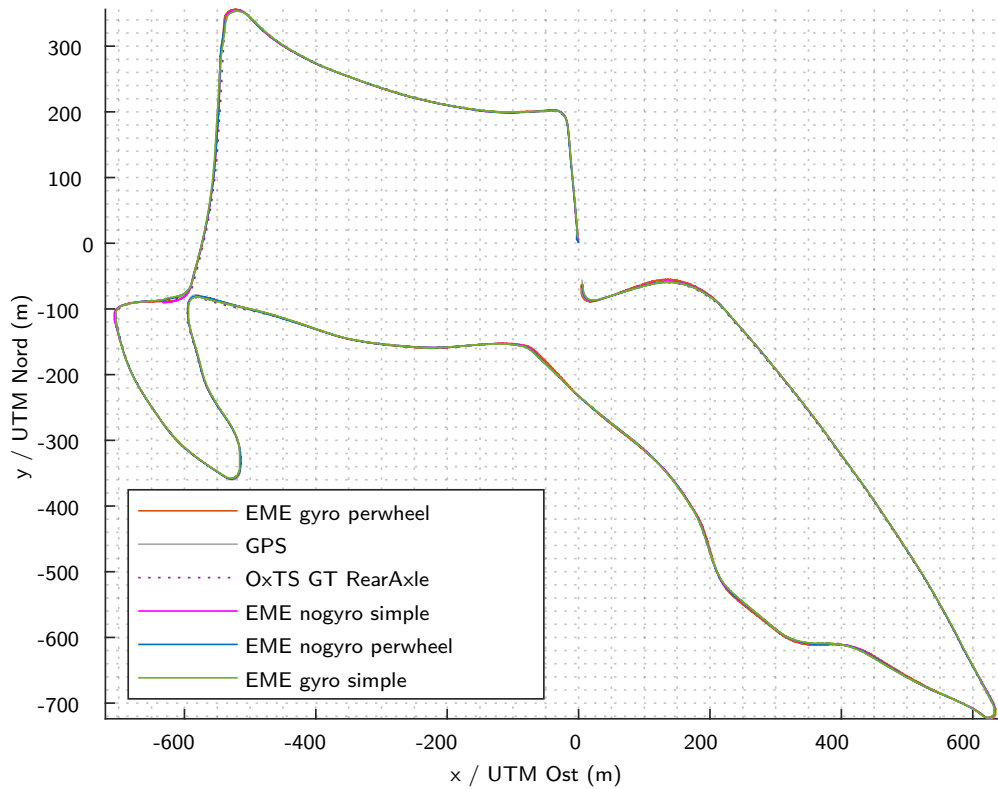
Die modulare Eigenbewegungsschätzung beobachtet neben dynamischen Zuständen auch die Sensorfehler, wie in Abb. 5.20 dargestellt. Da die Initialisierung der Algorithmen während der Fahrt gestartet wurde, konnte kein sog. Zero-Velocity-Update durchgeführt werden, sodass der Bias der Gierratensensors nur langsam geschätzt werden konnte.

Abbildung 5.21 zeigt durchdrehende Räder des LKWs im Gelände: in diesem Fall sind nicht mehr alle Geschwindigkeitsmessungen brauchbar und müssen entsprechend im Filter berücksichtigt werden. Normalerweise ist auch der TULF mit einem Übergrundgeschwindigkeitssensor ausgerüstet, der hier für Abhilfe sorgen könnte – dieser war allerdings in der betrachteten Messkampagne defekt.

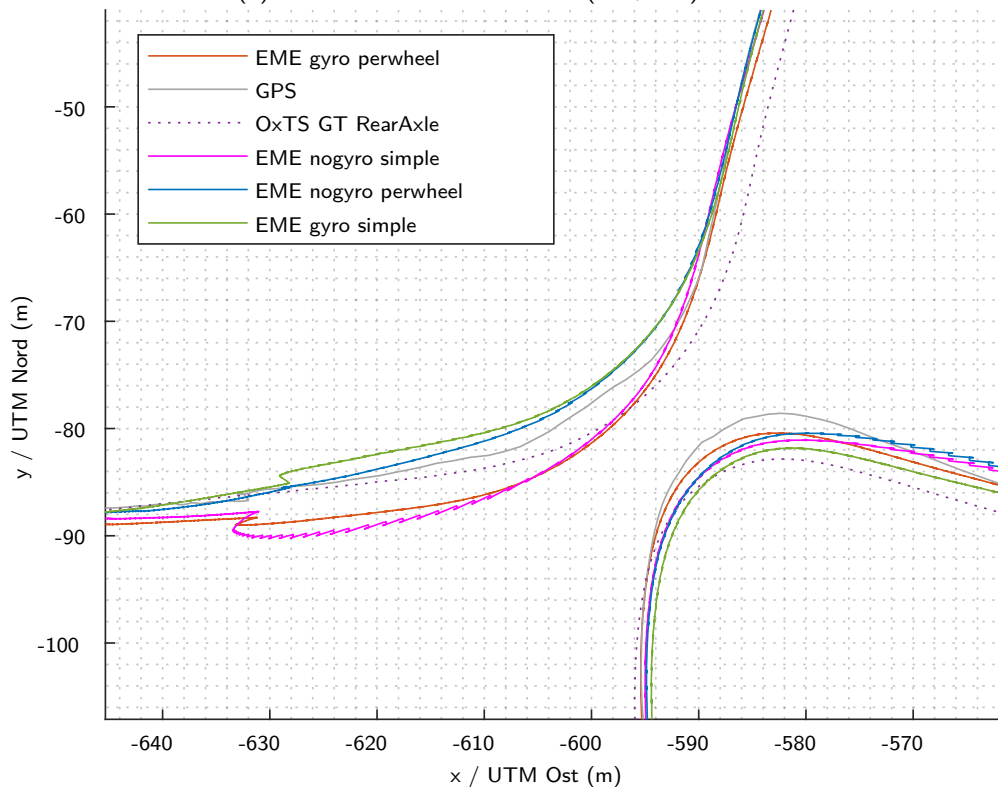
Betrachtet man die Sensorkonfigurationen in Abbildung 5.19b, so erscheint die blaue Konfiguration (mit Einzelradgeschwindigkeiten, ohne faseroptischen 1-Achs-Kreisel) in der ersten Kurve für die globale Lokalisierung am besten geeignet zu sein, da sie am dichtesten an der Ground-Truth (gestrichelt) liegt. In der zweiten Kurve zeigt sich aber eine Art Sägezahnmuster in der blauen und magentafarbenen Lösung, verursacht durch eine schlechtere Winkelschätzung. Das Sägezahnmuster entsteht durch eine Prädiktion „aus der Kurve heraus“ und eine Innovation mit der GPS-Positionsmessung ins Kurveninnere. Dieses Verhalten ist unerwünscht und tritt in den Konfigurationen mit faseroptischem 1-Achs-Kreisel nicht auf. Aus Sicht des Autors ist daher die orange Variante mit faseroptischem 1-Achs-Kreisel und Verwendung der Einzelradgeschwindigkeiten zu bevorzugen – diese sorgt neben einem glatten Verlauf und geringer Positionsabweichung zusätzlich für Redundanz bei Ausfällen von Einzelsensoren.

Abschließend wird das Dead-Reckoning betrachtet, welches für diese Fahrzeuge und ihre Anwendungen der wichtigste Ausgang der Eigenbewegungsschätzung ist. Für die Visualisierung in Abb. 5.22 wurden alle Trajektorien auf einen gemeinsamen Startpunkt transformiert.

5 Eigenbewegungsschätzung und Lokalisierung



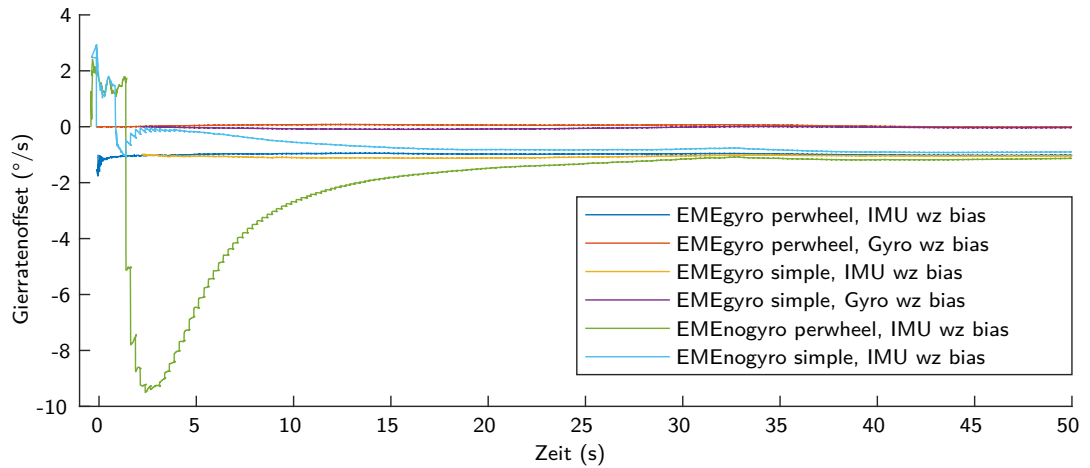
(a) Gesamte Strecke: Start bei (0 m, 0 m) nach Norden



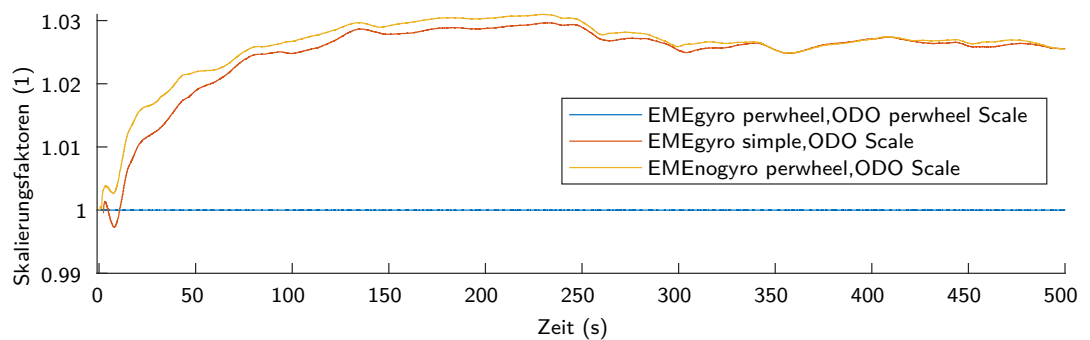
(b) Ausschnitt bei Stopp an (-630 m, -86 m) – in dieser Kurve traten die größten Abweichungen auf

Abbildung 5.19:

Globale Lokalisierung auf Basis des modularen Eigenbewegungsschätzers (vier verschiedene Sensor-Setups ggü. der Ground Truth). (a) zeigt die komplette Testrunde, (b) legt den Fokus auf einen Stopp nach durchdrehenden Rädern an einer Steigung.



(a) Bias der Gierratensoren



(b) Skalierungsfehler der Odometriesensoren

Abbildung 5.20:

Schätzung der Sensorfehler bei unterschiedlichen Konfigurationen des modularen Eigenbewegungsschätzers. (a) zeigt den Bias der Gierratensoren. Der Bias des faseroptischen 1-Achs-Kreisels ist weitgehend zu vernachlässigen und bewegt sich um $0^\circ/\text{s}$ (rot, violett). Der Bias der MEMS-basierten IMU ist größer und konvergiert in allen Konfigurationen gegen $-1^\circ/\text{s}$, allerdings mit unterschiedlicher Geschwindigkeit. Bei zusätzlicher Verwendung des 1-Achs-Kreisels konvergiert der Bias innerhalb weniger Sekunden (blau, gelb). Stehen nur Odometrie und GNSS zur Verfügung, dauert es wesentlich länger. (b) zeigt den Skalierungsfaktor der Geschwindigkeitssensoren. Für die Einzelradgeschwindigkeitsmessungen wurde dieser Faktor fest zu 1 gesetzt (blau), für die mittlere Geschwindigkeit konvergiert er aufgrund von niedrig angesetztem Systemrauschen sehr langsam gegen 1,025. Damit ist der Skalierungsfaktor nicht in der Lage – aber auch nicht dazu gedacht – kurzfristig auf veränderlichen Schlupf zu reagieren.

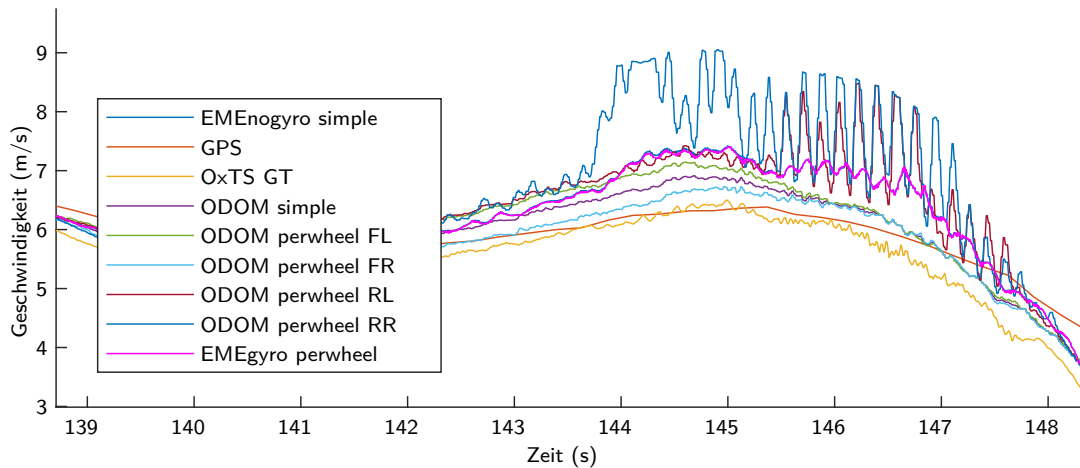


Abbildung 5.21:

Geschwindigkeiten in einer Rechtskurve (im Bereich von Abb. 5.19b): Die kurvenäußeren Räder drehen schneller, und aufgrund von Schlupf steigen die Abweichungen aller Räder ggü. der Ground Truth ab $t = 140$ s an. Ab $t = 143,5$ s ist an den Hinterrad-Geschwindigkeiten ein Durchdrehen zu erkennen, welches sich auch auf die geschätzte Geschwindigkeit auswirkt.

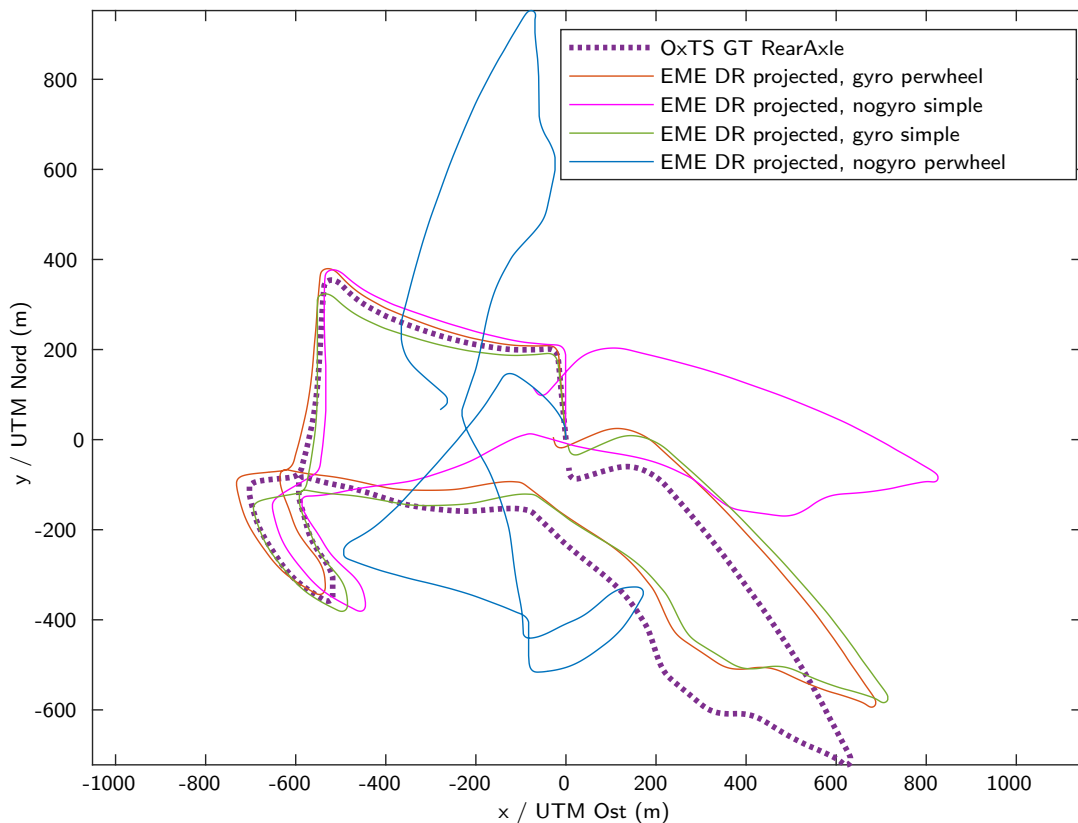


Abbildung 5.22:

Dead-Reckoning auf Basis des modularen Eigenbewegungsschätzers (dieselben Konfigurationen wie in Abb. 5.19). Die Varianten ohne 1-Achs-Kreisel (blau, magenta) driften stärker. Das Wegdriften auf den ersten Abschnitten der blauen Linie korreliert mit einem fehlerhaft geschätztem und erst gegen $t = 30$ s konvergierendem Bias des Gierratensensors der Xsens IMU (vgl. Abb. 5.20a).

Wie oben erwähnt, fand kein initiales Zero-Velocity-Update statt, sodass der Algorithmus während der Fahrt initialisieren musste – in einem Fall mit künstlich stark reduzierter Sensoranzahl führte dies zu einer anfänglich falschen Bias-Schätzung und damit zu einer stark driftenden Dead-Reckoning-Trajektorie (blau). Nach der ersten Kurve konnte der Bias besser beobachtet und die Drift verringert werden.

In den anderen drei Sensorkonfigurationen war die Drift wesentlich geringer. Alle diese Lösungen sind für den Einsatz des autonomen Konvoifahrens geeignet, was in vielen Messkampagnen gezeigt werden konnte. Abbildung 5.23a zeigt exemplarisch die Aufzeichnung einer autonomen Konvoifahrt bei einer Projektdemonstration in der Logistikschule der Bundeswehr in Garlstedt. Die Genauigkeit ist auch für die Erstellung von Hinderniskarten und die autonome Navigation gut geeignet, wie in Abb. 5.23b zu erkennen.

5.11 Fazit

Aus Sicht des Autors kann die *Forschungsfrage 4* „Kann die Eigenbewegungsschätzung für autonome Geländefahrzeuge mit kostengünstiger Sensorik erfolgen?“ nur teilweise als beantwortet gelten:

Die Verwendung von günstigerer Sensorik zur Eigenbewegungsschätzung innerhalb der im Rahmen dieser Arbeit entwickelten *modularen Eigenbewegungsschätzung* ist möglich und für den Hauptanwendungsfall geeignet: die Wahrnehmungs- und Navigationsmodule des Gesamtsystems benötigen vor allem eine stabile, sprungfreie und möglichst driftarme Dead-Reckoning-Pose, um Funktionalitäten wie Objekttracking oder Fahrspurtracking zu realisieren. Für den Konvoi-Betrieb in den Studien TULF und vor allem StrAsRob konnte der Regelkreis unter der Verwendung der Dead-Reckoning-Pose geschlossen werden, sodass alle Demonstrations- und Abnahmefahrten auch in engerem Gelände mit dem hier vorgestellten System erfolgen konnten.

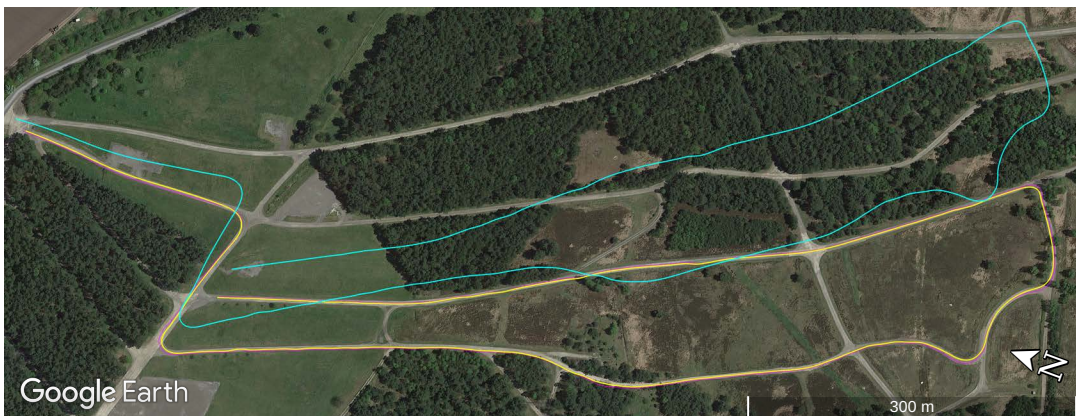
Betrachtet man die dynamischen Zustände an sich, so bestehend allerdings noch Potential für Verbesserungen – die aber im Rahmen der begrenzten Projektlaufzeit nicht mehr umgesetzt wurden.

Im LKW liegen wesentlich stärkere Vibrationen als im PKW vor, welche mit den verwendeten Low-Cost-Sensoren nicht vom Sensorrauschen separiert werden können. Die Fahrerkabine mit dem Großteil der Sensoren (GNSS, IMU, 1-Achs-Kreisel) ist beweglich ggü. dem Fahrzeugchassis aufgehängt, es kann um die x - und y -Achsen rotieren und sich parallel zur z -Achse verschieben. Da diese Bewegung bislang nicht separat gemessen werden kann, wird sie durch erhöhtes Mess- und Systemrauschen abgefangen, worunter die Filterperformance insgesamt leidet.

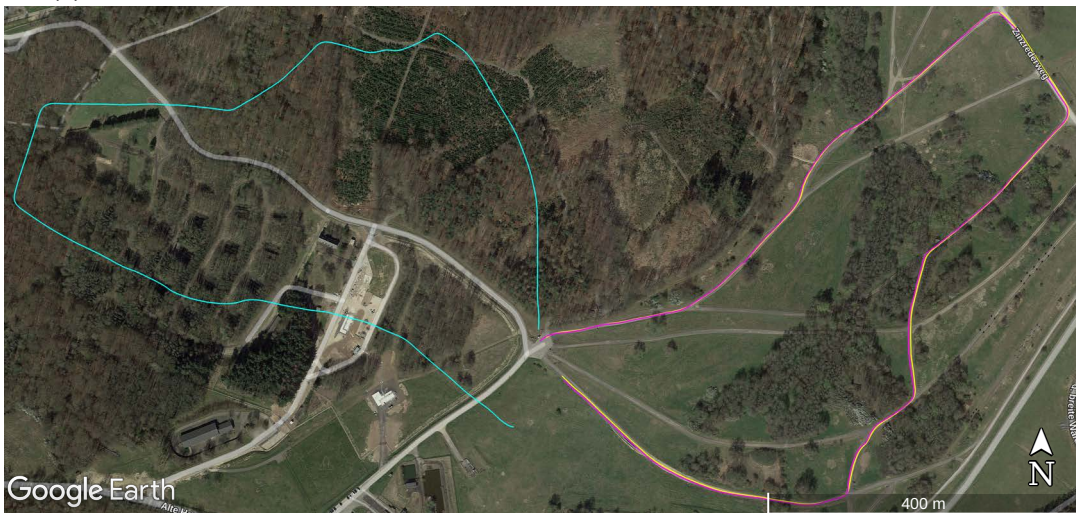
Die im Algorithmus integrierte Mitschätzung der Sensoreinbaulagen hat sich aufgrund der starken Vibrationen und des hohen Sensorrauschens als nicht geeignet herausgestellt und wurde für die Experimente deaktiviert.

Bei Nickbewegungen führt die nicht modellierte Relativbewegung aufgrund der gewählten Lage des Eigenbewegungskoordinatensystems F_{EME} im Hinterachsmittelpunkt und der Starrkörperannahme zu großen prädierten Pseudo-Änderungen in der z -Koordinate, also der Höhe. Für den Anwendungsfall des Konvois ist die Höhe jedoch nicht relevant. Daher wurde bei den meisten Fahrten auf eine Beobachtung der z -Koordinate verzichtet und nur eine 5-DOF-Pose bereitgestellt.

Eine Vielzahl an Sensoren erhöht auf der einen Seite die Ausfallsicherheit für das Gesamtsystem, andererseits wächst die Komplexität des Gesamtsystems, und die Fehleranfälligkeit steigt mit zunehmender Anzahl an Sensoren. Dieser Effekt tritt bei den



(a) Logistikschule der Bundeswehr, Garlstedt: Autonome Konvoifahrt mit dem StrAsRob



(b) StOÜbPI Koblenz: Datenaufzeichnung zur Wegeerkennung mit dem TULF.

Abbildung 5.23:

Anwendungsszenarien der Eigenbewegungsschätzung im Gelände: GPS-Positionsmessungen (magenta), geschätzte globale Pose (gelb) und Dead-Reckoning-Pose (cyan). Die Dead-Reckoning-Pose wurde nicht in einen gemeinsamen Startpunkt transformiert, daher bewegt sich die Trajektorie hier teilweise in die andere Richtung. Die in (b) an Abzweigungen erkennbaren Winkelfehler der Dead-Reckoning-Pose sind einer älteren Version der Algorithmen geschuldet.

beiden LKWs verstärkt auf, da die Sensoren über Schnittstellenwandler angebunden sind und es in den Experimenten regelmäßig zu Ausfällen kam.

Die *monolithische Eigenbewegungsschätzung* wurde primär in den MuCAR-Fahrzeugen in Verbindung mit einem hochwertigen INS verwendet, um auch im Fall von längeren GNSS-Störungen wie in Abb. 5.3 ein sprungfreies und driftarmes Dead-Reckoning bereitzustellen – hier jedoch als volle 6-DOF-Pose. Dieses System hat sich über viele Jahre bewährt.

5.11.1 Mögliche Erweiterungen

Zur Verbesserung der Lokalisierung können einerseits modernere Mehrsystem-GNSS-Empfänger genutzt werden – dieses machte sich beim Generationswechsel von RT3003 v1 auf v3 extrem bemerkbar. Andererseits bietet es sich vor allem in GNSS-gestörten Umgebungen an, weitere globale Positionsmessungen neben dem GNSS zu integrieren.

Am Institut wurden von Burger [2022] die Grundlagen für eine globale Lokalisierung auf Basis detektierter Landmarken und bekannter Karten entwickelt. Auf dieser Basis sind von Beer et al. [2022, 2023] Weiterentwicklungen mit detaillierteren und präziseren Karten entstanden, die zur Prädiktion zurzeit ein eigenes rudimentäres 2D-Bewegungsmodell nutzen. Eine Integration würde hier das detailliertere Bewegungsmodell aus dieser Arbeit mit GNSS-unabhängigen Positionsmessungen verbinden.

Eine andere Möglichkeit, die Drift der Lokalisierung bei GNSS-Ausfall zu reduzieren, sind zusätzliche Messungen der Relativbewegung zwischen zwei Zeitpunkten, bspw. aus einer visuellen Odometrie wie [Backhaus et al., 2023], einem Landmarken-SLAM-Verfahren (Simultaneous Localization and Mapping) wie [Beer et al., 2022] [Schütz et al., 2020] oder auf Basis von Radar-Messungen (Distanz und Doppler) [Barjenbruch et al., 2015]. Insbesondere letztere erlauben durch direkte Relativgeschwindigkeitsmessungen eine gute Stützung der Geschwindigkeitsschätzung und der zugehörigen Sensorfehler.

Für Fahrzeuge wie TULF und StrAsRob, bei denen ein Teil der Sensoren an der Fahrerkabine beweglich ggü. dem Fahrzeugchassis aufgehängt ist, wäre eine zusätzliche Schätzung der zeitveränderlichen 6-DOF-Fahrerkabinepose und damit der Sensorposen wünschenswert. Bei korrekter Bestimmung der Relativlage wird die aus Front- und Heck-LiDAR-Sensoren erzeugte gemeinsame Punktwolke konsistenter. Die Relativlage kann entweder durch Winkelsensoren an der Aufhängung gemessen werden oder durch an beweglichen und starren Teilen montierte IMUs beobachtet werden.

6 Fahrzeugregelung und Manöver

Die Fahrzeugregelung ist der zwingend notwendige Baustein im autonomen Fahren und der Robotik allgemein, um vom Open-Loop-System zum Closed-Loop-System zu gelangen. Open-Loop ist zwar die Erprobung der maschinellen Wahrnehmung möglich, allerdings hat sie keine Rückwirkungen auf das Fahrzeug. Im Closed-Loop-System werden nun die Ergebnisse der Wahrnehmungs- und Navigationsmodule (die später im Detail vorgestellt werden) genutzt, um eine Ansteuerung des Fahrzeugs zu realisieren.

Aus historischen und organisatorischen Gründen wird die Regelung in Quer- und Längsführung unterschieden. Ziel der Querführung ist die Regelung der Lage des Fahrzeugs innerhalb einer *Soll-Fahrspur*, also bspw. ein mittiges Fahren auf einem Feldweg oder das Passieren eines Hindernisses mit ausreichend Abstand. Die Längsregelung behandelt die Geschwindigkeits- oder Abstandsregelung zu anderen Verkehrsteilnehmern in der eigenen Fahrspur.

Die Quer- und Längsregler sind jeweils kaskadiert ausgeführt. Die oberen Reglerebenen werden meist niederfrequent im Takt der Wahrnehmung ausgeführt, die die Führungsgröße für die Regelung bereitstellt. Die unteren Reglerebenen werden mit höherer Frequenz ausgeführt, um Modellierungsfehler zu kompensieren und eine bessere Regelgüte zu erzielen. In den Abschnitten 6.1 und 6.2 werden die einzelnen Regler vorgestellt.

Die in Kapitel 3 vorgestellten Versuchsfahrzeuge sind mit unterschiedlichen Schnittstellen für die Ansteuerung im autonomen Betrieb ausgerüstet. Daher ist die gesamte Regelungskaskade modular aufgebaut, sodass möglichst große Anteile der Regelung unterlagert im Low-Level-System auf der Fahrzeugseite ausgeführt werden. Grund hierfür ist, dass dort im Normalfall das bessere Systemwissen vorliegt und daher eine höhere Regelgüte erreichbar ist.

Diese Schnittstelle ist für jedes Versuchsfahrzeug spezifiziert. Die Information über die fahrzeugseitig vorhandenen Regler für Quer- und Längsführung ist entweder für jedes Versuchsfahrzeug vorab bekannt oder wird dynamisch vom Versuchsfahrzeug an die oberen Regler kommuniziert. In Abschnitt 6.3 finden sich die Details zu dieser Aufteilung.

Abschnitt 6.4 stellt die *Manöver* vor, eine flexible Verkettung von Reglern mit Vorsteuerungen für die Realisierung unterschiedlicher Fahrzeugfähigkeiten für das Verhalten.

6.1 Längsregelung

Die Längsregelung ist kaskadiert ausgeführt und beginnt im Basisfahrzeug: basierend auf einer Soll-Beschleunigung wird die Aktorik, bestehend aus Gas und Bremspedal, bedient. Darüber liegen die Geschwindigkeitsregelung (Tempomat) und eine Anhalterege lung, und zuoberst die Abstandsregelung (ACC). Abbildung 6.1a gibt einen

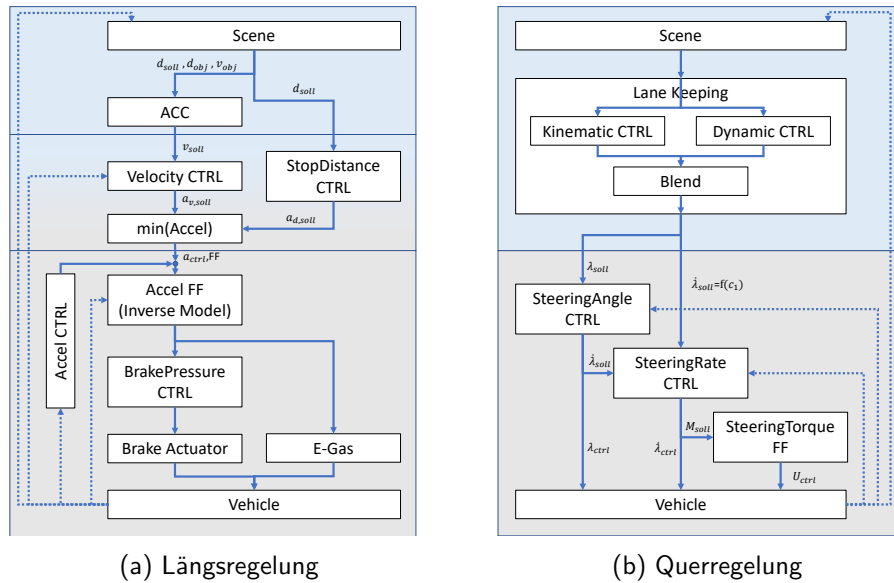


Abbildung 6.1:

Mögliche Kaskadierungen der Fahrzeugregelung. Blaue Elemente sind Teil des High-Level-Systems, grau hinterlegte Teil des Low-Level-Systems, Elemente innerhalb des Farbverlaufs sind je nach Fahrzeug auf High- oder Low-Level-System angesiedelt. In der Querregelung unterscheidet sich je nach Fahrzeug bzw. Aktorik, wie viele der unteren Regelkreise zum Einsatz kommen müssen.

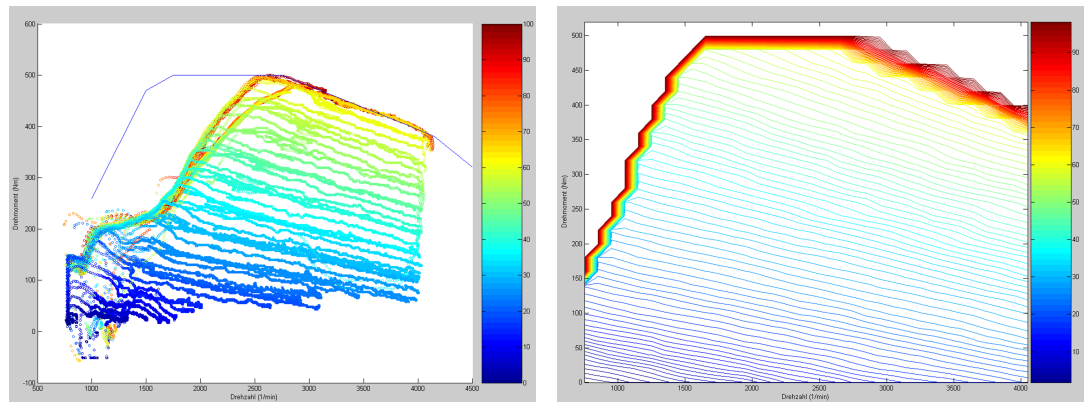
Überblick über die Kaskadierung der einzelnen Regler, die im Folgenden vorgestellt werden.

6.1.1 Beschleunigungsvorsteuerung und -regelung

Ziel der Beschleunigungsvorsteuerung ist eine möglichst gute Ausnutzung von Modellwissen über das Fahrzeug und das System Fahrzeug–Fahrweg. Basierend auf der Eingangsgröße Stellbeschleunigung a_{soll} gilt es eine Ansteuerung der Aktorik umzusetzen, die bereits ohne aktive Beschleunigungsregelung einen minimalen Beschleunigungsfehler erzielt. Die verbleibende Abweichung kann mit einem PI-Regler kompensiert werden.

Moderne Fahrzeuge verfügen meist für Serien-ADAS (Fahrerassistenzsystem, engl. *Advanced Driver Assistance System*) über eine Beschleunigungsschnittstelle. Da diese bei den verwendeten Versuchsträgern nicht vorhanden ist, wurde im Rahmen dieser Arbeit diese Low-Level-Regelung für MuCAR-3 umgesetzt – dies war nur möglich durch zahlreiche Messkampagnen zur Parameteridentifikation im Rahmen der Diplomarbeit von Bruns [2008].

Kern ist ein inverses Fahrzeugmodell, welches die im aktuellen Zustand bei den einwirkenden Fahrwiderständen benötigte Antriebskraft für die Wunschbeschleunigung vorhersagt und dazu eine Vielzahl an Messdaten aus dem Fahrzeug-CAN-Bus verwendet. Daraus kann in einem aufwändigen Modell, z. B. dem in Abb. 6.2 dargestellten



(a) Messdaten zum Erstellen des Drehmoment-Drehzahl-Kennfeldes

(b) Drehmoment-Drehzahlkennfeld

Abbildung 6.2:

Identifikation des Motorkennfeldes: Anhand zahlreicher Beschleunigungsfahrten mit konstanter Gaspedalstellung (blau=0 %, rot=100 %) ergaben sich die in (a) gezeigten Daten. Daraus wurde das Kennfeld in (b) abgeleitet, welches invertierbar ist. Quelle: [Bruns, 2008]

Motorkennfeld, und mit unterlagerten Regelkreisen die notwendige Ansteuerung für Gas-, Bremsaktorik bestimmt werden.

Das mit Unterstützung von Bruns [2008] entwickelte und parametrisierte System hat sich im Einsatz über viele Jahre bewährt – obwohl sich viele Parameter (Fahrzeugmasse, Windwiderstand) durch weitere Ein- und Aufbauten verändert haben und nicht im Modell angepasst wurden.

6.1.2 Geschwindigkeitsregler

Die Geschwindigkeitsregelung wird im normalen Personenkraftwagen (PKW) oft als Tempomat bezeichnet, sie dient der Minimierung des Geschwindigkeitsfehlers zwischen der Sollgeschwindigkeit v_{soll} und der aktuell gemessenen Ego-Geschwindigkeit v_{ego} .

In MuCAR-3 und MuCAR-4 kommt dazu ein PID-Regler zum Einsatz, der eine Sollbeschleunigung a_{soll} für die unterlagerte Regelung bestimmt. In den LKWs TULF und StrAsRob gibt es keine unterlagerte Beschleunigungsvorsteuerung und -regelung, hier werden direkt E-Gas, Bremse und Retarder angesteuert.

6.1.3 Abstandsregler / ACC

Der Abstandsregeltempomat, engl. *Adaptive Cruise Control* (ACC) dient dem Einhalten eines Wunschabstandes bzw. Sicherheitsabstandes d_{soll} zu einem vorausfahrenden Fahrzeug bei einer eingestellten Maximalgeschwindigkeit v_{soll} .

Eingangsgrößen sind der Ist-Abstand d_{obj} und die Ist-Geschwindigkeit v_{obj} zum Vorderfahrzeug, die aus den Daten der in Abschnitt 7.3.3 vorgestellten Objekt-trackingsysteme gewonnen werden. Ein geschwindigkeitsabhängiger Sollabstand d_{soll} mit konstanter Zeitlücke Δt_{gap} ergibt sich gemäß [Siedersberger, 2003] nach:

$$d_{\text{soll}} = \Delta t_{\text{gap}} \cdot \|v_{\text{ego}}\| + d_{\text{min}} . \quad (6.1)$$

Ziel ist die Fahrt mit derselben Geschwindigkeit v_{obj} wie das Führungsfahrzeug bei Minimierung des Distanzfehlers:

$$v_{\text{soll}} = v_{\text{obj}} + k_{\text{dist},p} (d_{\text{obj}} - d_{\text{soll}}) + k_{\text{vel},p} (v_{\text{obj}} - v_{\text{ego}}) . \quad (6.2)$$

Eine Formulierung mit einer Sollbeschleunigung wäre ebenfalls möglich, wird aber aufgrund des modularen Ansatzes für mehrere Fahrzeuge mit unterschiedlichen Schnittstellen nicht eingesetzt.

6.1.4 Anhalteregler

Der Anhalteregler dient dazu, das Fahrzeug an einem vorbestimmten Punkt in der Entfernung d_{soll} zum Stillstand zu bringen und dabei sowohl eine hohe Präzision als auch ein sanftes Abbremsen und Anhalten zu gewährleisten. Bei weit entfernten Punkten wird solange die eingestellte Sollgeschwindigkeit v_{soll} beibehalten, bis die Berechnung eine Mindestverzögerung für das Anhalten überschreitet. Eingangsdaten sind der Sollabstand d_{soll} , Ausgang ist eine Stellbeschleunigung.

Mit Unterstützung der Studienarbeit von Reichert [2009] wurde dies in Anlehnung an [Werling et al., 2008] für MuCAR-3 umgesetzt.

6.2 Querregelung

Die Querregelung ist ebenfalls kaskadiert aufgebaut, wie in Abb. 6.1b dargestellt. In dieser Arbeit findet dabei auf oberster Ebene eine Spurregelung Verwendung, die von den Kollegen Hecker [2007] und Heinrich [2023] entwickelt wurde und hier nur angerissen wird. Diese greift unterlagert auf Lenkwinkel- oder Lenkratenregler zurück.

6.2.1 Spurregelung

Bei der Spurführung ist das Ziel, die Lage des Fahrzeugs innerhalb einer *Soll-Fahrspur* zu regeln und dabei Fehler in Ablage d und Kurswinkelfehler $\Delta\Psi$ zu minimieren.

Die *Fahrspur* wird hier als Klothoidensegment mit einer Startkrümmung c_0 und Krümmungsänderung c_1 modelliert und enthält die Informationen zur Relativlage des Ego-Fahrzeugs (siehe Abb. 6.3). Die Verwendung der Klothoide zur Modellierung

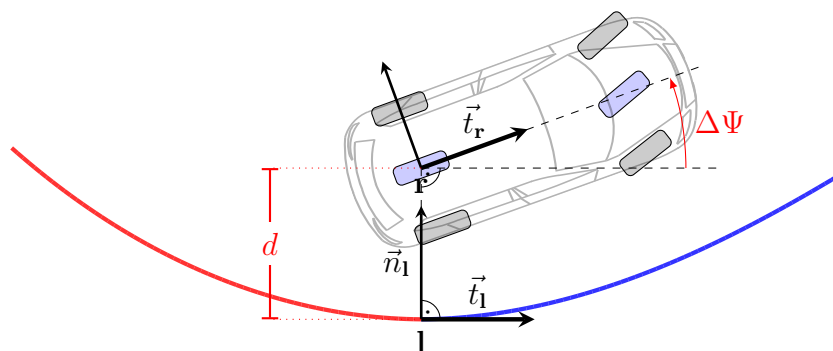


Abbildung 6.3:

Lokalisierung des Hinterachsmittelpunktes auf dem Pfad an der Stelle I. Die Ablage d ist der kürzeste Abstand zwischen Klothoide und Hinterachsmittelpunkt entlang des Normalenvektors \vec{n}_1 . Der Kurswinkelfehler $\Delta\Psi$ ist der Winkel zwischen der Tangente \vec{t}_1 an der Klothoide und der Fahrzeuglängsachse. Vorderes Klothoidensegment (blau) mit Startkrümmung $c_0 > 0$ und Krümmungsänderung $c_{f,1} < 0$. Hinteres Klothoidensegment (rot) mit gleicher Startkrümmung $c_{b,0}(0) = c_{f,0}(0)$ und Krümmungsänderung $c_{b,1} > 0$.

der Fahrspur hat eine lange Tradition im autonomen Fahren, sie wird seit den 1980er Jahren genutzt [Dickmanns und Zapp, 1987, Dickmanns und Mysliwetz, 1992, Eidehall und Gustafsson, 2004, Strygulec et al., 2013] und findet auch in der Industrie immer noch Verwendung [Karamanov et al., 2018].

Fahrspuren werden im hier verwendeten System von den unterschiedlichen Wahrnehmungs- und Navigationsmodulen (siehe Abschnitt 7.3.4) bereitgestellt, die bei der Erzeugung unterschiedliche Kriterien berücksichtigen: sie kann bspw. die Mitte eines Feldwegs beschreiben, auf Basis des von einem anderen Fahrzeugs gefahrenen Pfads bestimmt werden, oder eine Bahn beschreiben, um ein Hindernis mit ausreichend Abstand zu passieren.

Die Fahrspur wird im Regelfall von den Wahrnehmungsmodulen mit der Sensorfrequenz von 10 Hz aktualisiert. Um die Bewegung des Ego-Fahrzeugs zwischen Aufnahmezeitpunkt und Auswertung der Daten zu kompensieren, erfolgt mit der Frequenz der Eigenbewegungsschätzung (100 Hz oder höher) eine neue Bestimmung der Relativlage zum letztgültigen Fahrspursegment. Das erlaubt die Kompensation von Modellierungsfehlern und führt zu einer besseren Regelgüte.

Für die Regelung an sich können unterschiedliche Reglertypen zum Einsatz kommen, die ihre Stärken und Schwächen in unterschiedlichen Einsatzbereichen haben und denen unterschiedliche Modellierungen der Fahrzeugbewegung zugrunde liegen. Die Querdynamik vorderachsgelenkter Fahrzeuge in der Ebene kann vereinfachend durch Einspurmodelle beschrieben werden.

6.2.1.1 Kinematisches Einspurmodell

Ein auf dem kinematischen Einspurmodell (KESM) basierender Regler kommt für niedrige Geschwindigkeiten zum Einsatz. Das KESM beschreibt die Bewegung um einen Momentanpol, der sich rein aus geometrischen Größen ergibt (vgl. Abschnitt 5.5.3.1).

Die Parametrierung dieses Reglers, der Ablage d und Kurswinkelfehler $\Delta\Psi$ minimiert, erfolgt durch einen Zielpunkt in der Lauflänge L auf der Klothoide und eine Dämpfung, zusätzlich kann die Krümmung c_0 als Vorsteuerung verwendet werden (vgl. [Heinrich, 2023, Kap. 4.3.3]). Die Ausgangsgröße ist der Lenkwinkel $\lambda_{\text{solI,KESM}}$.

6.2.1.2 Dynamisches Einspurmodell

Ein auf dem dynamischen Einspurmodell (DESM) basierender Regler kommt für höhere Geschwindigkeiten zum Einsatz. Sein Bewegungsmodell basiert auf einem Kräfte- und Momentengleichgewicht im Schwerpunkt (vgl. Abschnitt 5.5.3.2). Die Herausforderung liegt in der Parameteridentifikation für das DESM, z. B. die Seitenkraftbeiwerte der Reifen in Abhängigkeit des Untergrunds. Die Ausgangsgröße ist der Lenkwinkel $\lambda_{\text{solI,DESM}}$.

Kriterien bei der geschwindigkeitsabhängigen Reglerauslegung von Hecker [2007, S. 23ff.] waren neben der Minimierung von Ablage d und Kurswinkelfehler $\Delta\Psi$ ein Reglerverhalten mit möglichst kleiner Anregelzeit, niedrigen Lenkraten und Schwingungsfreiheit.

Details zu den Fahrzeugmodellen finden sich z. B. bei Mitschke und Wallentowitz [2014]. Für Details zu den am Institut verwendeten Reglern für die Spurregelung sei explizit auf die Arbeiten von Hecker [2007] und Heinrich [2023] verwiesen.

6.2.1.3 Überblendung

Keiner der Regler ist für alle Geschwindigkeitsbereiche geeignet. Daher wurde ähnlich wie bei Kehl [2007] eine geschwindigkeitsabhängige Überblendung entwickelt: Für $v \leq 6 \text{ m/s}$ kommt ausschließlich der auf dem KESM basierende Regler zum Einsatz, für $v \geq 8 \text{ m/s}$ kommt ausschließlich der auf dem DESM basierende Regler zum Einsatz. Im Geschwindigkeitsbereich $6 \text{ m/s} \leq v \leq 8 \text{ m/s}$ erfolgt eine kontinuierliche, stoßfreie Überblendung.

6.2.2 Lenkwinkelregler

Eingang für den Lenkwinkelregler ist der Soll-Lenkwinkel λ_{solI} .

Von der Eigenbewegungsschätzung beobachtete Fehler (Offset b^{LWS} , Skalierung s^{LWS}) können in der Vorsteuerung genutzt werden, um einen verbesserten Stell-Lenkwinkel

$\lambda_{ctrl,ff}$ zu berechnen. Zusätzlich zur Vorsteuerung kann ein PID-Regler zur Korrektur von Abweichungen genutzt werden.

Das Versuchsfahrzeug MuCAR-3 (Munich Cognitive Autonomous Robot Car 3rd Generation) verfügt über eine unterlagerte Lenkratenregelung, dafür wird dann im Lenkwinkelregler die Soll-Lenkrate $\dot{\lambda}_{soll}$ mithilfe eines PID-Reglers bestimmt. Für die anderen Versuchsfahrzeuge ist die Ausgangsgröße direkt der Stell-Lenkwinkel λ_{ctrl} .

6.2.3 Lenkratenregler

Im Versuchsfahrzeug MuCAR-3 kommt ein zusätzlicher Lenkratenregler zum Einsatz, der die Lenkwinkelgeschwindigkeit $\dot{\lambda}_{soll}$ einregelt. Der Sollwert kommt meist aus dem Lenkwinkelregler, zusätzlich kann auf Basis der Krümmungsänderung der Klothoide eine Vorsteuerung bestimmt werden.

Der Ausgang ist ein Soll-Lenkmoment, welches proportional zum benötigten Strom für den elektrischen Lenkaktor ist.

6.3 Verteilung der Regler auf die Ebenen

Wie eingangs erwähnt, sind die in Kapitel 3 vorgestellten Versuchsfahrzeuge mit unterschiedlichen Schnittstellen für die Ansteuerung im autonomen Betrieb ausgerüstet. Daher wurde die gesamte Regelungskaskade modular aufgebaut, sodass möglichst große Anteile der Regelung unterlagert im Low-Level-System auf der Fahrzeugseite ausgeführt werden können. Grund hierfür ist, dass dort im Normalfall – gerade bei nicht selbst am Institut aufgebauten Versuchsfahrzeugen – das bessere Systemwissen vorliegt und daher eine höhere Regelgüte erreichbar ist.

Eine über die verschiedenen Fahrzeuge hinweg gleiche Schnittstelle (mindestens auf Middleware-Ebene, teilweise auch im CAN- (Controller Area Network) oder UDP-Protokoll (User Datagram Protocol) identisch) erleichtert hier die Wartungsfähigkeit der entwickelten Algorithmen.

Die Kombination der unterschiedlichen Regler ist angelehnt an das Reglerkonzept von Werling et al. [2008] (siehe Abb. 6.4) für das gemeinsame Versuchsfahrzeug AnnieWAY für die Teilnahme an der DARPA Urban Challenge, bei der sämtliche Regler auf einem System ausgeführt wurden. Aufgrund der für unsere Versuchsfahrzeuge notwendigen Trennung in High-Level- und Low-Level-Systeme wurde der Abstandsregler separiert.

6.3.1 Low-Level

Auf den institutseigenen Versuchsfahrzeugen MuCAR-3 und MuCAR-4 stehen auf dem Low-Level-Rechner Geschwindigkeits- und Anhalteregler sowie die Beschleunigungsvorsteuerung und -regelung inklusive Bremsdruckregelung zur Verfügung. Sie

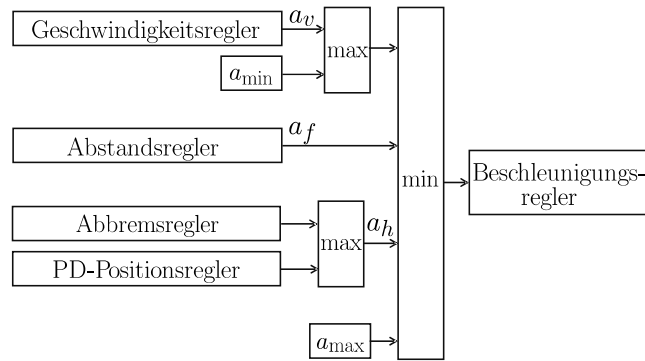


Abbildung 6.4: Multiregler mit Ablösestrategie. Bildquelle: [Werling et al., 2008]

werden auf echtzeitfähiger Hardware mit einer Frequenz von 1000 Hz resp. 500 Hz ausgeführt und steuern die Aktorik zum größten Teil über analoge Schnittstellen an. Nur auf den Low-Level-Rechnern stehen die dafür notwendigen, proprietären Informationen aus dem Serien-CAN-Bus des Basisfahrzeugs zur Verfügung.

Für die Versuchsfahrzeuge TULF und StrAsRob unserer Projektpartner ist das Low-Level-System als Teil der Fahrzeugplattform anzusehen. Der Geschwindigkeitsregler wurde seitens Rheinmetall Landsysteme GmbH (RLS) realisiert, der Lenkwinkel wird vom Low-Level-System an das DbW-System (engl. *Drive-by-Wire*) weitergeleitet.

Die Low-Level-Systeme der Versuchsfahrzeuge StreetScooter B14 Work *Eva* und *Adam* wurden in Anlehnung an MuCAR-4 aufgebaut, zusätzlich steht ein Zustandsregler (Geschwindigkeit, Beschleunigung) zur Verfügung. Für die in diesem Projekt geforderte hohe Positionierungsgenauigkeit wurden weitere Sensormessungen integriert sowie die Positions- und Bremsdruckregler für den Grenzbereich zwischen Haft- und Gleitreibung angepasst [Heinrich, 2023, Abschnitte 4.2.2 und 4.3.5].

Für das Versuchsfahrzeug StreetScooter B16 wurde die Aktorik vom Fahrzeughersteller umgerüstet und das Low-Level-System passend weiterentwickelt. Dieses wird hier als Teil der Fahrzeugplattform angesehen. Auf Basis der Geschwindigkeits- oder Beschleunigungsvorgaben werden das Soll-Antriebsmoment der E-Maschine (inkl. Rekuperation) sowie der Soll-Bremsdruck bestimmt, die von unterlagerten Systemen umgesetzt werden.

Insgesamt kann die Low-Level-Ebene in allen Versuchsfahrzeugen als langzeitstabil angesehen werden, sie ist eher Ingenieurshandwerk als Forschung.

6.3.2 High-Level

Alle nicht genannten Regler werden auf dem High-Level-Rechner ausgeführt, mindestens die Spurregelung und das ACC. Hier stehen die vollständigen Daten aus der Eigenbewegungsschätzung sowie den Wahrnehmungsmodulen über die Middleware zur Verfügung und ermöglichen somit eine einfache Entwicklung.

6.4 Manöver

Das in dieser Arbeit entwickelte High-Level-Regelungsframework verfügt über eine Reihe an Fahrzeugfähigkeiten, die als *Manöver* bezeichnet werden. Jedes Manöver besteht für Quer- und Längsführung aus je einer flexiblen Verkettung von Eingangsfiltern, Vorsteuerungen und Reglern; die verwendeten Bausteine können mehrfach verwendet werden. Einige dieser Manöver bieten zusätzlich die Möglichkeit, ihre Verfügbarkeit vorher abzufragen, wie bspw. „ist innerhalb einer Sekunde ein Spurwechsel möglich?“, und bieten eine Fortschrittskontrolle, bspw. in wie weit ein Spurwechsel abgeschlossen ist.

Zur Abbildung eines komplexeren Verhaltens werden die Manöver aus den einzelnen Zuständen eines hierarchischen Zustandsautomaten parametrisiert und aufgerufen (siehe Abschnitt 7.4). Notwendige Parameter sind bspw. die Höchstgeschwindigkeit, ein situationsabhängiger Mindestabstand zu anderen Verkehrsteilnehmern, und eine explizite Information über die relevante Fahrspur bzw. das relevante Objekt. Abbildung 6.5 gibt einen Überblick über den Systemaufbau.

Ein großer Vorteil ist der asynchrone Betrieb: während die Verhaltensentscheidung eher niederfrequent arbeitet und die Szene, bestehend aus unterschiedlichen möglichen Fahrspuren und einer Reihe von dynamischen Objekten, analysiert und die Manöver parametrisiert, erfolgt die Ausführung der Manöver und damit der Regler mit höherem Takt. Je nach verwendetem Eingangsfilter findet die Verarbeitung entweder getriggert durch neue Wahrnehmungsdaten statt (10 bis 20 Hz), oder getriggert durch neue Lokalisierungsdaten (100 Hz).

Im Rahmen der Diplomarbeit von Reichert [2010] wurde eine Entwicklungsumgebung erstellt, um einzelne Manöver in der Simulation und auf dem realen Versuchsfahrzeug losgelöst vom Gesamtsystem parametrisieren, ausführen und damit testen zu können.

Im Folgenden wird eine Reihe der häufig verwendeten Manöver mit ihren Bausteinen vorgestellt.

Bahnregelung mit ACC Die *Bahnregelung mit ACC* ist das Manöver, welches für den Großteil der Verhaltensweisen zum Einsatz kommt: Spurfolgen (auf Basis von visuell erkannten Fahrspuren oder unter Verwendung der GNSS- und kartenbasierten Spurgenerierung), Objektfolgen im Konvoi, oder bei der hindernisvermeidenden Tentakel-Navigation. Über die Parametrisierung werden ID der zu verwendeten Fahrspur sowie die ID des im ACC maßgeblichen dynamischen Objekts konfiguriert sowie die Parameter für das ACC (Mindestabstand d_{\min} , Zeitlücke Δt_{gap} und Geschwindigkeitsobergrenze v_{soll}).

Ein einfacher Eingangsfilter liest nun die korrespondierenden Objekte aus der Middleware ein. In der Querführung erfolgt eine Krümmungs- bzw. Krümmungsänderungsvorsteuerung für Lenkwinkel und Lenkrate. Zusätzlich wird die Änderungsgeschwindigkeit von Ablage und Kurswinkelfehler limitiert, um ungewollte Sprünge zu dämpfen, bevor

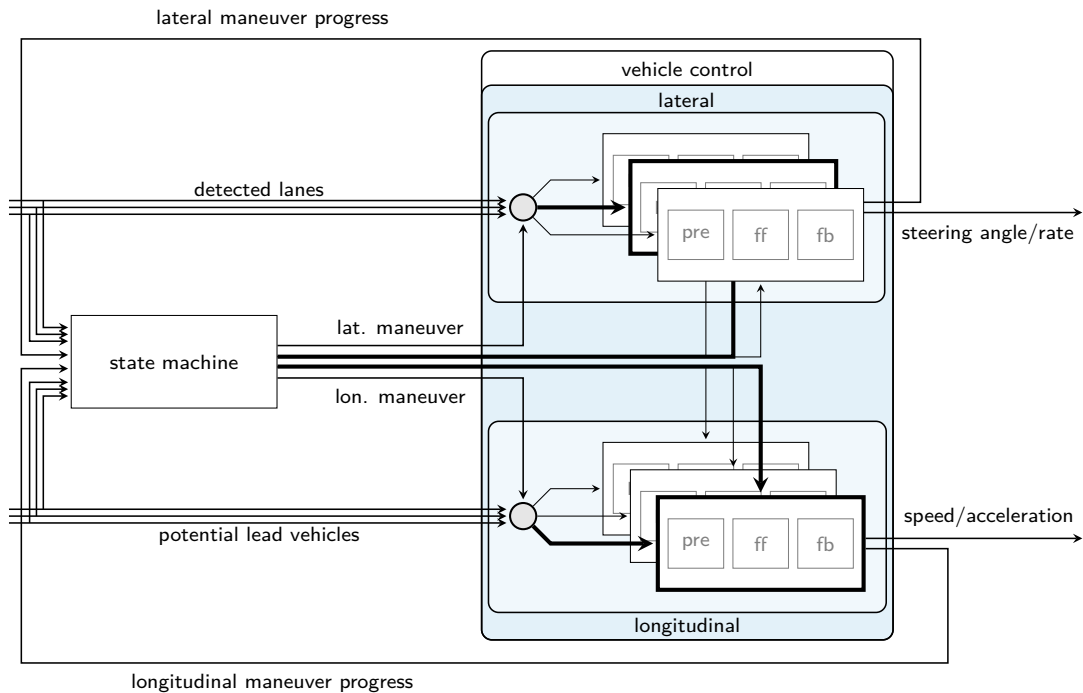


Abbildung 6.5:

Überblick über die Verwendung der Manöver im High-Level-Regelungsframework. Bildquelle: [Heinrich, 2023, S. 18].

in der Spurregelung die verbleibenden Fehler korrigiert werden. In der Längsführung entfällt die Vorsteuerung, das Objekt wird direkt an die ACC-Regelung übergeben.

Spurwechsel Das Manöver *Spurwechsel* dient bei mehrspurigen Straßen zum Spurwechsel, bspw. zum Überholen anderer Fahrzeuge. Es baut auf der *Bahnregelung mit ACC* auf, zusätzlich wird hier mit einer weiteren Vorsteuerung in der Querführung eine Soll-Ablage zur bisherigen Fahrspur aufgebaut. Die Parametrierung gibt hier Betrag und Richtung der Soll-Ablage vor, anschließend werden Vorsteuerungen für Lenkwinkel und Lenkrate sowie ein modifizierter Ablage- und Kursverlauf an den Regler übergeben.

Für dieses Manöver ist die Rückmeldung an die Verhaltensebene relevant: sobald die Zielablage erreicht wurde, das Fahrzeug also neben seiner alten Fahrspur und weitgehend in der neuen Fahrspur ist, kann ein neues Manöver *Bahnregelung mit ACC* mit neuer Spur-ID aktiviert werden.

In der gemeinsamen Veröffentlichung mit Hecker et al. [2010] wird darauf aufbauend eine Erweiterung für ein objektrelatives Ablagemanöver vorgestellt.

Abbiegen Dieses Manöver dient zum gezielten Abbiegen an Kreuzungen, um eine kontrollierte Fahrt vom eingehenden Kreuzungsast auf den Zielast zu gewährleisten. Als Parameter werden eine Maximalgeschwindigkeit und eine Pose auf dem Zielast relativ zum Ego-Fahrzeug übergeben. Während der Anfahrt auf eine Kreuzung wird

das Manöver auf Verfügbarkeit und spätestmöglichem Startpunkt geprüft, und erst dann gestartet.

In der Umsetzung unterscheidet es sich maßgeblich von den bisher vorgestellten Manövern, da es für die Querführung nicht mehr auf einer von außen bereitgestellten Fahrspur basiert. Bei Aktivierung des Manövers wird innerhalb der Abbiege-Vorsteuerung eine geschwindigkeitsabhängige Bahn berechnet, die aus schließender Klothoide, einem optionalen Kreissegment und einer öffnenden Klothoide besteht. Reicht der gegebene Platz nicht zum direkten Abbiegen aus, so wird automatisch etwas „ausgeholt“, wie in Abb. 6.6a dargestellt. Während der Manöverausführung wird das Ego-Fahrzeug mithilfe der Eigenbewegungsschätzung zyklisch auf der Bahn lokalisiert, daraus ergeben sich die Sollwerte für den Querregler.

Eine erste Umsetzung dieser Abbiegemanöver entstand in der Diplomarbeit von Reichert [2010], eine exemplarische Auswertung ist in Abb. 6.6 gezeigt. Daraus lässt sich folgern, dass bei der Vorberechnung der Bahn nicht die physikalisch maximalen Lenkwinkel und Lenkraten verwendet werden, um Stellgrößenreserve für die Regelung zu behalten und damit die Ablage zu limitieren.

Das neu entwickelte Verfahren basiert auf den Ansätzen für das Versuchsfahrzeug VaMoRs (Versuchsfahrzeug für autonome Mobilität und Rechnersehen) von Müller [1996, S. 52ff.], der ebenfalls mithilfe des Ackermann-Modells eine Kurvenfahrt modellierte und daraus Sollgrößen für die Vorsteuerung und Regelung ableitete.

Gangschalten Ebenfalls als Manöver realisiert ist das *Gangschalten* bzw. der Fahrstufenwechsel zwischen Vorwärtsfahrt „D“, Rückwärtsfahrt „R“ oder Stillstand „P“, die als Parameter übergeben werden. Hier entfallen in der Querführung die Vorsteuerungen und Regler komplett. In der Längsführung sorgt eine spezialisierte Vorsteuerung für die Kommandierung der gewünschten Fahrstufe an das Low-Level-System.

Lenken im Stand Das Manöver *Lenken im Stand* wird beispielsweise für eine Dreipunktswende benötigt, bei der der Ziellenkwinkel als Parameter übergeben wird. Eine Lenkwinkel-Vorsteuerung übergibt diesen Ziellenkwinkel an einen Lenkwinkelregler, der diesen mit definiert begrenzter Lenkrate einregelt. Für die Längsführung kommt eine Anhalte-Vorsteuerung in Kombination mit dem oben eingeführten Anhalteregler zum Einsatz.

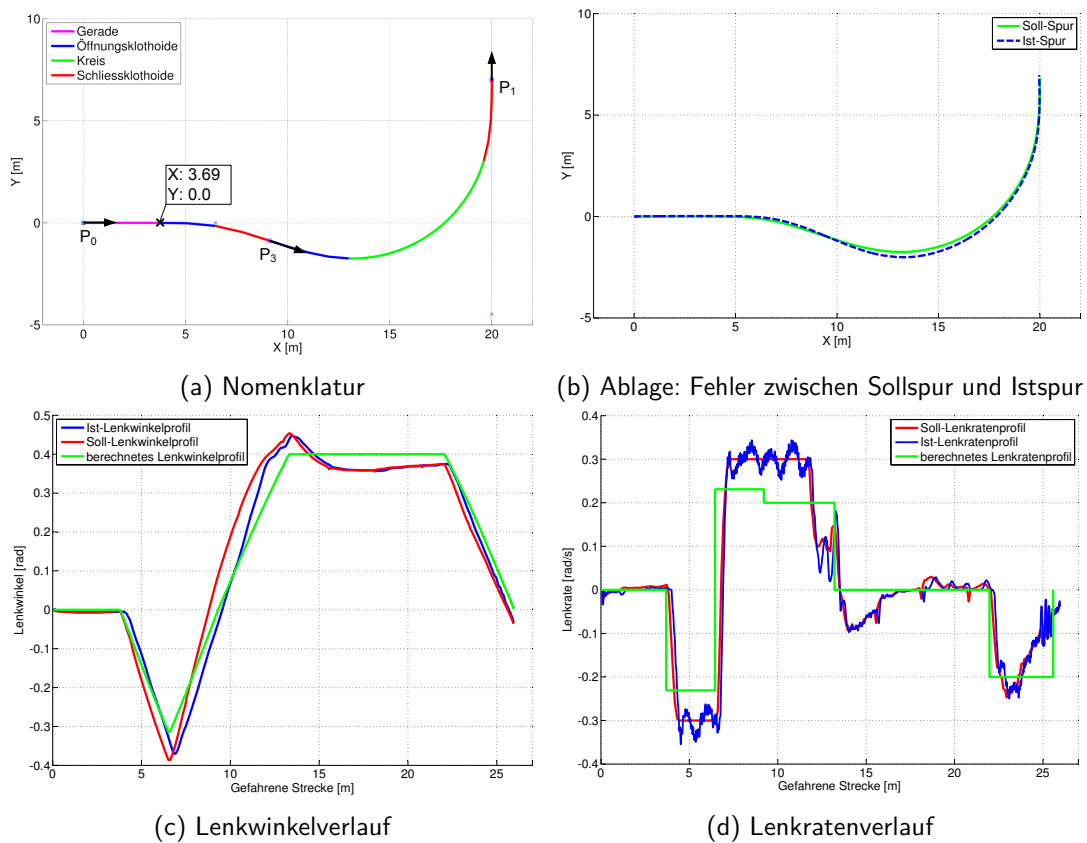


Abbildung 6.6:

(a) Bahnberechnung für ein Abbiegemanöver auf Basis konstanter Lenkraten. Aufgrund träger Aktordynamik kann dem Sprung auf der Lenkrate nicht gefolgt werden (d), woraus Fehler auch in Lenkwinkel und Ablage resultieren. Der Bahnfolgeregler kompensiert dies durch höhere Beträge des Soll-Lenkwinkels (c), die in höheren Soll-Lenkraten resultieren. Insgesamt kann damit trotz der unzureichenden Vorsteuerung eine Ablage von ca. 20 cm beim Bahnfolgen erzielt werden (b). Bildquelle: [Reichert, 2010].

7 Missionen, Aufgaben und Verhaltensweisen

Das Institut für Technik Autonomer Systeme (TAS) hat seit 2007 regelmäßig als *Team MuCAR* mit dem Versuchsfahrzeug MuCAR-3 an der European Land Robot Trial (ELROB) teilgenommen, die je nach Perspektive Leistungsshow oder Wettbewerb ist:

The European Land Robot Trial (ELROB) is Europe's toughest & longest-running field robotics and unmanned systems event. Designed and guided by practitioners, made for the users, the industry and the R&D sector and all in the spirit of open-sciences & open-research. Since 2006 the ELROB successfully offers realistic scenarios that reflect the most up-to-date end-user requirements. ELROB is

- not organised as a competition but as a trial,
- a strictly non-profit activity and
- strictly R&D oriented.

(<https://www.elrob.org/>)

Nichtsdestotrotz wurde von den Veranstaltern in jedem Jahr auch eine Rangfolge der Teams verkündet.

Bei den ersten Teilnahmen in den Jahren 2007 bis 2009 konnte unser neu aufgestelltes Team unter Verwendung einzelner Verhaltensweisen bereits eine gute Performance zeigen, aber auch viele Dinge für die Folgejahre lernen:

2007 Im *Autonomous Reconnaissance-Szenario* der C-ELROB (Civilian European Land Robot Trial) 2007 in Monte Ceneri kam der ursprüngliche Tentakel-Ansatz (siehe Abschnitt 7.3.6) aus der Kategorie der *reaktiven Navigation* zum Einsatz, der rein auf den LiDAR-Daten (Light Detection And Ranging) basierte und keine Kartennutzung integriert hatte. Dies bedingte manuelle Eingriffe an allen Abzweigungen, trotzdem konnte die komplette, aus urbanen und nicht-urbanen Anteilen bestehende Strecke mit einem hohen Anteil an Autonomie und mit der schnellsten Zeit¹ befahren werden.

2008 Im *Transport-Convoy-Szenario* der M-ELROB (Military European Land Robot Trial) 2008 in Hammelburg kam eine frühe Version des LiDAR-basierten Fahrzeug-trackings (siehe Abschnitt 7.3.3.1) in Verbindung mit einer Spurgenerierung (Abschnitt 7.3.4.2 zum Einsatz – dies gehört in die Kategorie der *geführten Navigation*. Eine fehlerhafte Gewichtung von GNSS-Messungen (Globales Navigationssatellitensystem, engl. *Global Navigation Satellite System*) im dichten Wald führte zu einem Lokalisierungssprung, zu einem Sprung in der Sollbahn und letztendlich zu einem Verlassen der Straße – dieser Verzögerung führte zum zweiten Platz². Daraus

¹<https://www.elrob.org/files/elrob2007/elrob-2007-result-table.png>

²<https://www.elrob.org/elrob-2008-results>

erwuchs die Notwendigkeit, eine Eigenbewegungsschätzung zu entwickeln, die neben der globalen Lokalisierung eine Pose in einem driftarmen Dead-Reckoning-System bereitstellt.

2009 Das *Autonomous Navigation-Szenario* der C-ELROB 2009 in Oulu ermöglichte uns die Erprobung der um visuelle Merkmale und Kartenwissen erweiterten Tentakel (vgl. Abschnitt 7.3.6 und Abschnitt 7.3.6) und zeigte beeindruckende Performance in engem Gelände, womit der erste Platz erreicht werden konnte³. Allerdings war das System wenig modular aufgebaut, sondern von der globalen Routenplanung bis zur Auswahl des besten Tentakels in einem Programm kombiniert. Dieses System kombinierte die Navigationsart *reaktive Navigation* mit Elementen der *globalen Navigation*.

Unter anderem aus den Erkenntnissen dieser ersten Wettbewerbsteilnahmen entstand der Bedarf, für komplexere Szenarien eine stärker modularisierte Architektur aufzubauen und diese Module flexibel zu verknüpfen – die Geburt der *Mission* für die folgenden Wettbewerbe.

Was sind Missionen im Allgemeinen, und warum werden sie genutzt? Wie bereits in der Einleitung eingeführt, soll sich das autonome Fahrzeug entgegen der reinen Begriffsdefinition nicht völlig frei entscheiden, welche Ziele es verfolgt und wie es sie erreicht. Vielmehr soll das Fahrzeug definierte Aufgaben für den menschlichen Nutzer erfüllen, z. B. den Warentransport entlang einer vorgegebenen Route zu einem anderen Ort.

Im Rahmen dieser Arbeit wurde ein hierarchisches Konzept entwickelt, welches aus *Missionen*, *Aufgabentypen* und *Verhaltensweisen* besteht, und experimentell erprobt. Eine Mission wird zu Beginn vom Nutzer definiert und kann eine oder mehrere Aufgaben umfassen. Jede Aufgabe hat ein Missionsziel, beispielsweise das Erreichen einer globalen Zielkoordinate, welches mit einem bestimmten Aufgabentyp erreicht werden soll. Mögliche Aufgabentypen sind hier beispielsweise die Navigation auf einem Straßennetz oder das Verfolgen eines anderen Fahrzeugs im Konvoi. Zur Erfüllung einer Aufgabe kommen unterschiedliche Verhaltensweisen zum Einsatz, die auf den Ergebnissen unterschiedlicher *Wahrnehmungs- und Navigationsmodule* aufbauen, die vornehmlich von den Kolleginnen und Kollegen am Institut entwickelt wurden, und für die Fahrzeugführung auf Manöver und die Fahrzeugregelung aus Kapitel 6 zurückgreifen.

Teile dieses Kapitels wurden vorab als Konferenzbeitrag auf der *IEEE Intelligent Transportation Systems Conference* präsentiert [Luettel et al., 2011a]. Im Journal-Beitrag [Luettel et al., 2017] in der *at – Automatisierungstechnik* liegt der Schwerpunkt auf weiteren Anwendungsfällen.

³<https://www.elrob.org/elrob-2009-results>

Gliederung Zunächst werden die unterschiedlichen *Aufgabentypen* in Abschnitt 7.1 vorgestellt. Abschnitt 7.2 stellt die für die Aufgaben benötigten *Verhaltensweisen* vor, die wiederum die Ergebnisse der *Wahrnehmungs- und Navigationsmodule* aus Abschnitt 7.3 benötigen. Im anschließenden Abschnitt 7.4 wird die Koordination der Einzelverhalten durch einen hierarchischen Zustandsautomaten präsentiert. In Abschnitt 7.5 werden die Missionen zur Verkettung von Aufgaben vorgestellt: Dazu gehören neben der Missionserstellung auch Missionsüberwachung und -kontrolle, z. B. die Übergänge zwischen den unterschiedlichen Aufgaben. Abschließend werden in Anhang C die Ergebnisse mehrerer real durchgeführter Missionen präsentiert.

Eine grafische Übersicht über die Aufgaben, Verhaltensweisen, Wahrnehmungs- und Navigationsmodule und deren Verknüpfung findet sich in Abbildung 7.1.

7.1 Aufgabentypen

Eine *Aufgabe*, engl. *Task*, definiert ein konkretes Ziel für das autonome Fahrzeug, welches es unter Verwendung geeigneter Verhaltensweisen erreichen soll. Dabei sind gewisse Beschränkungen zu beachten, unter anderem zugelassene Untergrund- und Umgebungsklassen oder Geschwindigkeitsbegrenzungen.

Für die im Rahmen dieser Arbeit untersuchten Missionen wurde eine Reihe möglicher Aufgabentypen realisiert, welche in diesem Abschnitt vorgestellt werden. Die in den einzelnen Aufgaben genutzten Verhaltensweisen werden im nächsten Abschnitt 7.2 detailliert.

Konvoi

Dieser Aufgabentyp bezeichnet das präzise Verfolgen eines Objekts auf seinem gefahrenen Pfad, wobei Abstand und Fahrzeuggeschwindigkeit unter Berücksichtigung dieses Objekts geregelt werden. Bei den verfolgten Objekten handelt es sich in der Regel um andere Fahrzeuge, aber auch Personen oder Radfahrer sind möglich. Im Folgenden wird hier zur Vereinfachung vom sog. *Führungsfahrzeug* gesprochen.

Der *Konvoi* ist weitgehend unabhängig von der Umgebung, benötigt keine Straßen oder Wege und kann quer durch Wiesen oder über offene Plätze führen. Die Entscheidung über den „besten Pfad“ wird dem Fahrer des Führungsfahrzeugs überlassen, der dabei die Dimension des Folgefahrzeugs und dessen geometrische Beschränkungen wie z. B. den Wendekreis berücksichtigen und idealerweise Schlaglöcher usw. vermeiden sollte. Für den Fall, dass das Führungsfahrzeug falsch abbiegt oder in eine Sackgasse gerät, kann das Führungsfahrzeug das Ego-Fahrzeug „rückwärts rangieren“.

Für diesen Aufgabentyp ist es notwendig, den Typ des Führungsfahrzeugs zu definieren. Zusätzlich wird durch die Wahl des Wunschabstandes d_{soll} sowie der Maximalgeschwindigkeit v_{max} die Parametrierung des ACC (Abstandsregeltempomat, engl. *Adaptive Cruise Control*) vorgenommen. Ist *Konvoi* nicht die einzige Aufgabe in einer Mission, so ist noch eine Abbruchbedingung erforderlich, hier kann es sich

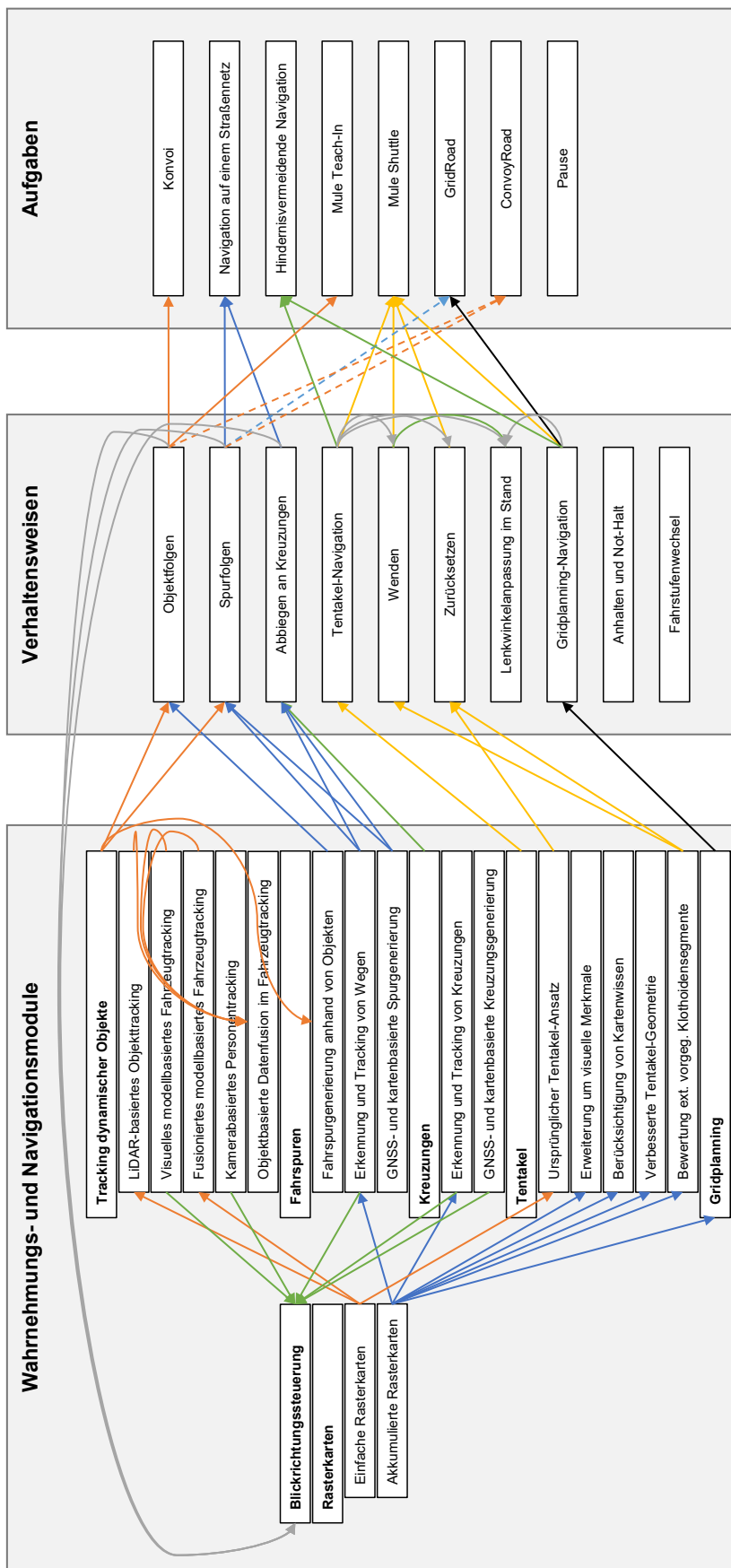


Abbildung 7.1: Übersicht über die Aufgaben, Verhaltensweisen, Wahrnehmungs- und Navigationsmodule dieses Kapitels.

beispielsweise um das Erreichen eines in globalen Koordinaten gegebenen Zielpunktes $\mathbf{p}_{\text{target}}$ handeln. Die vom Führungsfahrzeug gewählte Route zum Zielpunkt ist dabei unerheblich, der menschliche Fahrer darf – bspw. bei lokalen Sperrungen oder besserer Ortskenntnis – von der im Folgefahrzeug geplanten Route abweichen.

Für diesen Aufgabentyp wird das Verhalten *Objektfolgen* (siehe Abschnitt 7.2.1) benötigt.

Navigation auf einem Straßennetz

Dieser Aufgabentyp bezeichnet die autonome Navigation entlang von Straßen, Abzweigungen und Kreuzungen, um mit dem Fahrzeug einen vorgegebenen Zielpunkt $\mathbf{p}_{\text{target}}$ zu erreichen. Benötigte Verhaltensweisen für diese Aufgabe sind demnach das *Spurfolgen* (siehe Abschnitt 7.2.2), welches im Normalfall auf Basis der Sensordaten eine Fahrspur schätzt und dieser folgt, sowie das *Abbiegen an Kreuzungen* (siehe Abschnitt 7.2.3). Diese Verhaltensweisen enthalten keine explizite Behandlung von Hinderniszellen, sondern liefern nur die unter gewissen Rahmenbedingungen optimale Schätzung der Fahrspur bzw. Kreuzung, die unter Umständen nicht vollständig hindernisfrei sind.

Für diese Aufgabe wird eine globale Route von der aktuellen Position des Fahrzeugs \mathbf{p}_{ego} zum Zielpunkt $\mathbf{p}_{\text{target}}$ benötigt, welche auf Basis der Karte unter Berücksichtigung von Beschränkungen, wie z. B. Vermeidung von bebauten Gebieten, sowie entlang weiterer optionaler Wegpunkte geplant wird. Die Routenplanung (Details siehe Anhang B.3.2) liefert semantisch annotierte Wegpunkte $\mathcal{P}_{\text{routeAnnotated}}$ entlang der Straßen sowie eine Modellierung der entlang der Route zu passierenden Kreuzungen $\mathcal{C}_{\text{route}}$.

Die Anzahl der notwendigen Parameter für diesen Aufgabentyp ist geringer als für den *Konvoi*, wichtig ist vor allem die Definition der Maximalgeschwindigkeit v_{max} .

Wird von einer optionalen Objekt- bzw. Hinderniserkennung in den Sensordaten ein statisches Objekt erkannt, welches die Fahrspur komplett und permanent blockiert, so wird global eine neue Route zum Zielpunkt $\mathbf{p}_{\text{target}}$ geplant.

Als Abbruchbedingung für die Aufgabe *Navigation auf einem Straßennetz* dient das Erreichen eines in globalen Koordinaten gegebenen Zielpunktes $\mathbf{p}_{\text{target}}$. Liegt $\mathbf{p}_{\text{target}}$ aufgrund von Abweichungen zwischen GNSS-basierter Lokalisierung in der Karte und erkannter Straße abseits der Straße, wird ersatzweise der nächstgelegene Punkt $\mathbf{p}_{\text{targetOnTrack}}$ auf der Straße gewählt.

Hindernisvermeidende Navigation

Der Aufgabentyp *Hindernisvermeidende Navigation*, an anderen Stellen auch als *Wegpunktnavigation* bezeichnet, dient ebenfalls dem Verfolgen einer globalen Route. Er unterscheidet sich von der *Navigation auf einem Straßennetz* vor allem dadurch, dass keine explizite Erkennung von Leitstrukturen wie Fahrspuren oder Kreuzungen

benötigt wird. Damit ist dieser Typ für die Navigation in unstrukturiertem Gelände geeignet, bei dem die Karte oftmals nur grob den realen Wegverlauf abbildet.

Die verwendete Verhaltensweise *Tentakel-Navigation* benötigt eine Sequenz an globalen Wegpunkten, welche entweder von der Routenplanung (Details siehe Anhang B.3.2) bereitgestellt wird oder vom Bediener manuell mit Hilfe eines Luftbildes gesetzt werden. In besonderen Situationen, z. B. bei Blockaden, kommen Verhaltensweisen wie das *Wenden* oder *Zurücksetzen* zum Einsatz.

Eine alternative Verhaltensweise für diese Aufgabe ist die *Gridplanning-Navigation*. Diese verknüpft bereits Vorwärts- und Rückwärtssegmente, um in engem Gelände navigieren zu können, daher sind weniger Wechsel zwischen den Verhaltensweisen notwendig als bei der *Tentakel-Navigation*.

Die Anzahl der notwendigen Parameter für diesen Aufgabentyp ist ebenfalls gering: die Maximalgeschwindigkeit v_{\max} .

Als Abbruchbedingung für die Aufgabe *Hindernisvermeidende Navigation* dient das Erreichen eines in globalen Koordinaten gegebenen Zielpunktes $\mathbf{p}_{\text{target}}$.

Mule Teach-In

Der Aufgabentyp *Mule Teach-In* ist eine Variante des *Konvoi*. Hier geht es allerdings nicht allein darum, von einem anderen Objekt, beispielsweise einer Person oder einen Fahrzeug, zu einem anderen Ort geführt zu werden, sondern zusätzlich darum, den gefahrenen Weg $\mathcal{P}_{\text{recorded}}$ zu kartieren, um diese Karte in der anschließenden Aufgabe *Mule Shuttle* für die Routenplanung verwenden zu können.

Da der Zielpunkt im *Mule Teach-In* üblicherweise zu Beginn noch unbekannt ist, muss eine andere Abbruchbedingung zum Einsatz kommen, bspw. der Stillstand der Person für eine definierte Mindestzeit $\Delta t_{\text{stoppingLongTime}}$.

Mule Shuttle

Dieser Aufgabentyp ist eine Variante der *Hindernisvermeidenden Navigation*. Allerdings stehen bei Missionsbeginn weder Karte noch Zielpunkte zur Verfügung, diese werden erst während der Aufgabe *Mule Teach-In* als $\mathcal{P}_{\text{recorded}}$ generiert. Im *Mule Shuttle* pendelt das Fahrzeug zyklisch entlang des angelernten Pfads. Der jeweilige Sollpfad $\mathcal{P}_{\text{route}}$ ist das Ergebnis der Routenplanung auf einer nur aus $\mathcal{P}_{\text{recorded}}$ bestehenden Karte, wobei die Start- und Zielpunkte jeweils beim Erreichen eines der Enden vertauscht werden. Eine typische Abbruchbedingung wäre das Erreichen einer definierten Anzahl von Zyklen oder das Überschreiten einer Maximaldauer $\Delta t_{\text{desiredTaskDuration}}$.

Pausieren

Das *Pausieren* ist der simpelste Aufgabentyp, hier verweilt das Fahrzeug am Zielpunkt der vorigen Aufgabe für einen nicht näher definierten Zweck. Denkbar ist hier eine Be- bzw. Entladung, deren Ende durch das Erteilen eines externen Signals mitgeteilt wird. Ebenso möglich ist eine Mindestverweilzeit zu Aufklärungszwecken. Eine typische Abbruchbedingung wäre das Erreichen der Mindestverweilzeit $\Delta t_{\text{desiredTimeout}}$.

7.2 Verhaltensweisen

Verhaltensweisen, engl. *Behavior*, beschreiben langfristige Aktionen, die das autonome Fahrzeug ausführen kann, z. B. das Verfolgen eines anderen Fahrzeugs oder das Fahren entlang einer Fahrspur. Unterschiedliche Navigationsmodule, wie beispielsweise die visuelle Spurerkennung, das Kamera- oder LiDAR-basierte Fahrzeugtracking mitsamt einer nachgelagerten Spurgenerierung oder die hindernisvermeidende Navigation, stellen als gemeinsame Repräsentation des Soll-Fahrwegs jeweils ein Klothoidensegment bereit, welches die Relativlage des eigenen Fahrzeugs zur jeweils erkannten Fahrspur beschreibt. Mehrere Navigationsmodule können dabei parallel ausgeführt werden, jedes liefert dabei abhängig von der Umgebung Klothoiden als Ausgangsdaten.

Die Wahrnehmungsmodule liefern zusätzlich Informationen zu dynamischen Objekten (z. B. andere Fahrzeuge) sowie zu statischen Bestandteilen der Umgebung (z. B. Kreuzungen). Die sog. *Szene* setzt sich aus all diesen Objekten zusammen und ist ein abstraktes Modell der Fahrzeugumgebung.

In Abhängigkeit der gerade aktiven Verhaltensweise sowie der *Szene* kommen unterschiedliche *Manöver* für die Ansteuerung des Fahrzeugs zum Einsatz, welche in Abschnitt 6.4 eingeführt wurden, z. B. die Fahrzeugquerführung auf einer Fahrspur bei gleichzeitiger Fahrzeuglängsführung über ein ACC. Diese Manöver werden in Abhängigkeit der *Szene* parametrisiert und wählen geeignete Kombinationen aus Vorsteuerung und Regelung aus.

Je nach Szenario können an jeder Aufgabe verschiedene Arten von Verhaltensweisen beteiligt sein, die zur Erfüllung der aktuellen Aufgabe des Ego-Fahrzeugs zusammenwirken. Die in [Unterholzner, 2016] entwickelte Aufmerksamkeitssteuerung (siehe Abschnitt 7.3.1) wird in Abhängigkeit der Verhaltensweise parametrisiert, um die Blickrichtung der aktiven Kameraplattform MarVEye-8 (Multifokales aktives / reaktives Fahrzeug-Auge der 8. Generation, engl. *Multifocal active / reactive Vehicle Eye 8th Generation*) (siehe Abschnitt 3.2.3.3) zu beeinflussen und somit den Informationsgewinn in den Wahrnehmungsmodulen zu maximieren.

7.2.1 Objektfolgen

Die Verhaltensweise *Objektfolgen* bezeichnet das präzise Verfolgen eines Objekts auf seinem gefahrenen resp. gelaufenen Pfad, wobei Abstand und Fahrzeuggeschwin-

digkeit unter Berücksichtigung dieses Objekts geregelt werden. Sie kommt in den Aufgabentypen *Konvoi* und *Mule Teach-In* zum Einsatz.

Für die Verhaltensweise *Objektfolgen* sind drei Komponenten notwendig: Zuerst muss das Führungsfahrzeug in beliebigen Sensordaten, beispielsweise im Kamerabild oder in der 3D-Punktwolke des LiDAR-Sensors, detektiert und getrackt werden. Unterschiedliche Verfahren sind in Abschnitt 7.3.3 beschrieben. Auf Basis dieser Schätzung, welche Position und Orientierung im Raum relativ zum Ego-Fahrzeug liefert, sowie der bekannten Bewegung des Ego-Fahrzeugs kann anschließend der vom Führungsfahrzeug gefahrene Pfad im inertialen Raum F_{DR} generiert werden. Nach Lokalisierung des Ego-Fahrzeugs relativ zum Pfad kann eine Klothoide abgeleitet werden, die zur Querführung des Ego-Fahrzeugs dient. Die Funktionsweise der Fahrspurgenerierung für das Objektfolgen ist in Abschnitt 7.3.4.2 erläutert.

Für die Längsführung kommt hier die ACC-Implementierung aus Abschnitt 6.1.3 zum Einsatz, wie es seit einigen Jahren auch in Serienfahrzeugen Standard ist. Hierzu ist die Entfernung zum Führungsfahrzeug entlang des Pfads sowie dessen aktuelle geschätzte Geschwindigkeit erforderlich. Im Manöver *Bahnregelung mit ACC* werden die hier verwendete Quer- und Längsführung zusammengefasst, siehe Abschnitt 6.4.

Für den gelegentlichen Fall, dass das Führungsfahrzeug falsch abbiegt oder in eine Sackgasse gerät, kann das Führungsfahrzeug das Ego-Fahrzeug rückwärts schieben. Hier können unterschiedliche Navigationsmodule zur Querführung verwendet werden: Eine simple Lösung in statischer Umgebung ist das Zurückschieben entlang des gefahrenen Weges, der präzise bekannt ist, unter Verzicht auf jegliche Hinderniserkennung. Hier kommt dieselbe Art der Klothoidengenerierung wie bei der Vorwärtsfahrt zum Einsatz (vgl. Abschnitt 7.3.4.2). Eine Alternative ist die Verwendung der *Tentakel* zur Hindernisvermeidung beim Zurückschieben (vgl. Abschnitt 7.3.6). Die Längsführung erfolgt in beiden Varianten über ein ACC, allerdings mit im Vergleich zur Vorwärtsfahrt verkleinertem Sollabstand, um nicht unnötig weit zurückzusetzen.

Im *Objektfolgen* ist der Fokus der Sensorik primär auf das Führungsfahrzeug zu richten, da anhand dessen getrackter Position die eigene Fahrzeugführung durchgeführt wird. Ist im Ego-Fahrzeug eine aktive Kameraplattform vorhanden, kann dies mittels geeigneter Parametrierung der in Abschnitt 7.3.1 vorgestellten Aufmerksamkeits- und Blickrichtungssteuerung umgesetzt werden.

7.2.2 Spurfolgen

Die Verhaltensweise *Spurfolgen* dient zum Fahren entlang von Straßen und Wegen und wird primär für die Aufgabe *Navigation auf einem Straßennetz* verwendet. Auf der Wahrnehmungsseite wird hierzu primär ein sensorbasiertes Tracking von Fahrspuren benötigt (siehe Abschnitt 7.3.4.1), welches in fest gewählter Vorausschauweite eine Schätzung der Fahrspurgeometrie bereitstellt. Aus diesen statischen Punkten wird analog zum Konvoi eine Fahrspur generiert (siehe Abschnitt 7.3.4.2).

Zusätzlich steht aus der globalen Routenplanung eine GNSS-basierte Beschreibung des Wegverlaufs $\mathcal{P}_{\text{routeAnnotated}}$ bereit. Dieser kann einerseits als Hypothese für die sensorbasierte Spurerkennung genutzt werden, aber dient auch zur Bereitstellung einer Rückfallebene. In Abschnitt 7.3.4.3 wird die Erzeugung der GNSS-basierten Klothoide für die Querführung behandelt.

Die Sollgeschwindigkeit v_{soll} für das *Spurfolgen* ergibt sich aus der Maximalgeschwindigkeit v_{max} , die von der *Aufgabe* vorgegeben wird, der Geschwindigkeitsinformation v_{map} der fahrzeugnahen Punkte aus $\mathcal{P}_{\text{routeAnnotated}}$, sowie einer – aufgrund der fahrzeugabhängigen maximalen Querbesehleunigung – krümmungsabhängigen Geschwindigkeitsbegrenzung $v_{\text{querbeschl}}$:

$$v_{\text{soll}} = \min(v_{\text{max}}, v_{\text{map}}, v_{\text{querbeschl}}) . \quad (7.1)$$

Befinden sich weitere Fahrzeuge auf der eigenen Fahrspur, deren Position und Dynamik von einem Objekttracker (siehe Abschnitt 7.3.3) bereitgestellt wird, so kommt für die Längsführung ein ACC zum Einsatz, welches unter Beachtung von v_{soll} einen entsprechenden Sicherheitsabstand d_{safety} einregelt.

Im Manöver *Bahnregelung mit ACC* werden die hier verwendete Quer- und Längsführung zusammengefasst, siehe Abschnitt 6.4.

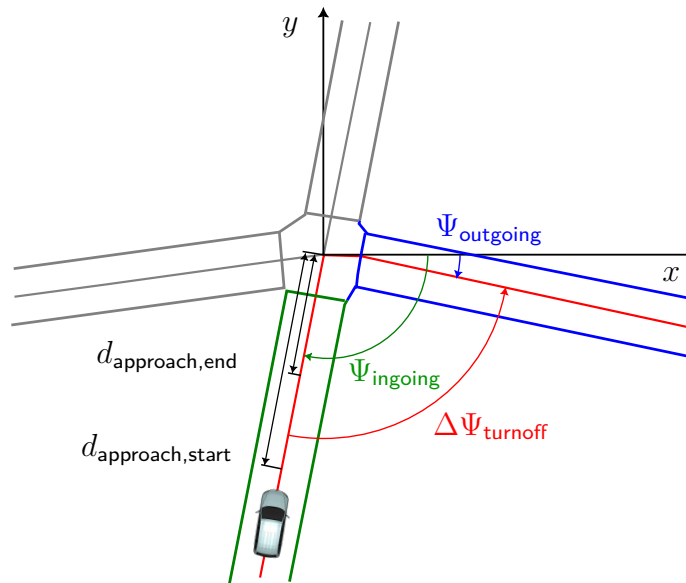
Ist im Ego-Fahrzeug eine aktive Kameraplattform vorhanden, wird im *Spurfolgen* der Fokus der Sensorik primär auf die getrackte Fahrspur gerichtet. Dies wird über die Bereitstellung von artifiziellen Objekten gelöst, die in einer geschwindigkeitsabhängigen Vorausschauweite auf der getrackten Ego-Fahrspur und seinen etwaigen Nachbarfahrspuren platziert werden. Diese werden von der in Abschnitt 7.3.1 vorgestellten Aufmerksamkeits- und Blickrichtungssteuerung zur Bestimmung der optimalen Blickrichtung genutzt.

7.2.3 Abbiegen an Kreuzungen

Die Verhaltensweise *Abbiegen an Kreuzungen* ist ein weiterer wichtiger Bestandteil für die Aufgabe *Navigation auf einem Straßennetz*. Im Gegensatz zur getrackten Klothoide, die beim Spurfolgen auf Wegen für die Querführung benötigt wird, ist das Abbiegen ein komplexerer Vorgang, sowohl auf Seiten der Wahrnehmung als auch der Fahrzeugführung.

Die Trackingalgorithmen aus Abschnitt 7.3.5.1 nutzen Sensordaten aus Kamera- und LiDAR-Sensoren, um Position und Geometrie einer Kreuzung rekursiv zu schätzen. Zur Initialisierung der Kreuzungsgeometrie werden Hypothesen aus der Karte (siehe Abschnitt 7.3.5.2) verwendet. Diese liefern die Anzahl und Richtung der Kreuzungsäste und eine grobe Positionsinformation des Kreuzungsmittelpunktes. Abbildung 7.2 zeigt schematisch die Kreuzung mit den verwendeten Parametern.

Da insbesondere in der kartenbasierten Rückfallebene keine Information darüber vorliegt, ob eine ausgedehnte Kreuzungsfläche oder eine eher kompakte Kreuzung vorliegen, wird sicherheitshalber ein Abbiegen auf engem Raum, d. h. mit starkem

**Abbildung 7.2:**

Abbiegen an einer Kreuzung mit vier Ästen. Der Eingangsast ist grün, der Ausgangsast blau dargestellt. Während der Annäherung an die Kreuzung wird die Geschwindigkeit des Ego-Fahrzeugs zwischen $d_{\text{approach,start}}$ und $d_{\text{approach,end}}$ in Abhängigkeit der Winkeldifferenz $\Delta\Psi_{\text{turnoff}}$ reduziert.

Lenkeinschlag, gefordert. Für die Start- und Zielpunkte des Abbiegemövers auf den Kreuzungsästen wird die minimale Entfernung zum Kreuzungsmittelpunkt $d_{\text{approach,end}}$ ermittelt, für die das Abbiegemanöver von Reichert [2010] noch genutzt werden kann.

Die Fahrzeuggeschwindigkeit wird während der Annäherung an die Kreuzung zwischen $d_{\text{approach,start}}$ und $d_{\text{approach,end}}$ linear von v_{soll} auf v_{turnoff} reduziert. Die Geschwindigkeitsreduktion ist zweierlei motiviert: Einerseits wird mithilfe der aktiven Blickrichtungssteuerung [Unterholzner, 2016] der Fokus abwechselnd auf die einzelnen Kreuzungsäste gelegt und somit erst ein dichtes Bild des Umfeldes erzeugt und das Tracking der kompletten Kreuzungsgeometrie ermöglicht. Andererseits dient die niedrige Geschwindigkeit dazu, überhaupt mit stärkerem Lenkeinschlag und trotzdem geringer Querbeschleunigung abbiegen zu können.

Für die Bestimmung von v_{turnoff} ist die Winkeldifferenz $\Delta\Psi_{\text{turnoff}}$ zwischen Eingangs- und Ausgangsast maßgeblich, sie ergibt sich aus der Geometrie der getrackten Kreuzung, ersatzweise der Kreuzungshypothese. Damit erfolgt eine lineare Interpolation von v_{turnoff} :

$$\Delta\Psi_{\text{turnoff}} = \|(-\pi + \Psi_{\text{ingoing}} - \Psi_{\text{outgoing}}) \bmod 2\pi\| \quad (7.2)$$

$$v_{\text{turnoff}} = v_{\text{vehicle,min}} + (v_{\text{turnoff,max}} - v_{\text{vehicle,min}}) \frac{\Delta\Psi_{\text{turnoff}} - \Delta\Psi_{\text{slow}}}{\Delta\Psi_{\text{fast}} - \Delta\Psi_{\text{slow}}} \quad (7.3)$$

Experimentell wurden folgende Parameter bestimmt:

$$v_{\text{vehicle,min}} = 1 \text{ m/s}, \quad v_{\text{turnoff,max,mapbased}} = 4 \text{ m/s}, \quad v_{\text{turnoff,max,vision}} = v_{\text{vehicle,min}}, \quad (7.4)$$

$$\Delta\Psi_{\text{fast}} = 10^\circ, \quad \Delta\Psi_{\text{slow}} = 45^\circ. \quad (7.5)$$

Eine Kreuzung wird geradeaus (und bei leichten Richtungsänderungen $\Delta\Psi_{\text{turnoff}} \leq \Delta\Psi_{\text{fast}}$) also schneller durchfahren als beim scharfen Abbiegen, hier ist die langsamste vom Fahrzeug fahrbare Geschwindigkeit $v_{\text{vehicle,min}}$ maßgeblich. Zusätzlich erfolgt eine Unterscheidung nach der Herkunft der Kreuzung: in der Kartenbasierten Rückfallebene gilt eine höhere obere Grenze $v_{\text{vehicle,max}}$ als bei der visuellen Kreuzungserkennung.

Im Manöver *Abbiegen* werden die hier verwendete Quer- und Längsführung zusammengefasst, siehe Abschnitt 6.4.

Liegt in den beiden zuvor genannten Verhaltensweisen der Fokus der Aufmerksamkeitssteuerung weitgehend konstant auf einem Objekt, so muss hier ein größerer Sichtbereich abgedeckt werden. Ab Unterschreiten einer Mindestentfernung zum Kreuzungsmittelpunkt wird nicht nur ein artifizielles Objekt auf der Ego-Fahrspur platziert, sondern auch weitere unentdeckte artifizielle Objekte auf den weiteren Kreuzungsästen in 20 m Entfernung vom Kreuzungsmittelpunkt. Die Aufmerksamkeitssteuerung optimiert nun durch zyklische Blickrichtungswechsel auf die artifiziiellen Objekte den Informationsgewinn. Dadurch wird der komplette Kreuzungsbereich abgedeckt und die Datengrundlage für die Kreuzungserkennung (siehe Abschnitt 7.3.5.1) auf Basis einer eingefärbten akkumulierten Rasterkarte (siehe Abschnitt 7.3.2.2) gelegt.

7.2.4 Tentakel-Navigation

Anders als die zuvor genannten Verhaltensweisen wird die *Tentakel-Navigation* vor allem in unstrukturierten Umgebungen verwendet. Die Sollspur $\mathcal{P}_{\text{routeAnnotated}}$ zu einem bestimmten Zielpunkt $\mathbf{p}_{\text{target}}$ wird für die Aufgabe *Hindernisvermeidende Navigation* aus groben bzw. ungenauen GIS-Karteninformationen (Geoinformationssystem) und für die Aufgabe *Mule Shuttle* anhand einer zuvor aufgezeichneten und potentiell verfälschten Spur generiert. Bei der Missionsplanung ist zudem nicht bekannt, ob die geplante Route hindernisfrei ist.

Die *Tentakel-Navigation* folgt näherungsweise dieser Sollspur, ohne jeden Wegpunkt dabei genau erreichen zu müssen. Abweichungen im Bereich mehrerer Meter sind durchaus typisch [Luettel et al., 2009] und in ungenauen Karten und fehlerhafter Lokalisierung begründet.

In den meisten untersuchten Szenarien wurde diese Verhaltensweise auf die Vorwärtsfahrt beschränkt. Im Konvoi-Szenario der M-ELROB 2010 wurde die *Tentakel-Navigation* in der Aufgabe *Konvoi* verwendet, wenn das Ego-Fahrzeug vom Führungsfahrzeug rückwärts geschoben wurde.

Die Verhaltensweise *Tentakel-Navigation* nutzt primär die in Abschnitt 7.3.6 beschriebenen *Tentakel* zur integrierten Wahrnehmung und Bewegungsplanung. Zur Vorverarbeitung der Sensordaten kommen die Rasterkarten aus Abschnitt 7.3.2 zum Einsatz. Die *Tentakel-Navigation* ist nur dann möglich, wenn die von der Wahrnehmung bereitgestellte Klothoide – hier als *Tentakel* bezeichnet – in ihrer Anfangskrümmung mit dem aktuellen Lenkwinkel des Fahrzeugs korrespondiert. Ansonsten ist ein temporärer Übergang zur Verhaltensweise *Lenkwinkelanpassung im Stand* (siehe Abschnitt 7.2.7) erforderlich.

Die Blickrichtungssteuerung wird in der *Tentakel-Navigation* so parametrisiert, dass der Fokus auf dem ausgewählten *Tentakel* und damit in Fahrtrichtung liegt.

7.2.5 Wenden

Die Verhaltensweise *Wenden* dient primär zur Korrektur einer größeren Abweichung zwischen der Ausrichtung des eigenen Fahrzeugs und der Sollspur und damit auch zum Fahrtrichtungswechsel auf engem Raum. Bei hindernisfreier Umgebung führt die empirisch gestaltete Verhaltensweise zu einer Dreipunktswende für einen 180°-Richtungswechsel. Sind Hindernisse vorhanden, werden entsprechend mehr Züge erforderlich.

Für die Querführung wird eine maximal gekrümmte Klothoide erzeugt, die die kinematischen Fahrzeugeigenschaften berücksichtigt. Um die Manövrierfähigkeit zu erhöhen, wird der Lenkwinkel im Stand – durch temporären Übergang zur Verhaltensweise *Lenkwinkelanpassung im Stand* (siehe Abschnitt 7.2.7) – angepasst. Die maximale Fahrstrecke pro Zug wird auf die Länge des 90°-Kreisbogensegments bzw. auf die Strecke bis zum ersten Hindernis begrenzt, die sich aus der Prüfung mithilfe der *Tentakel* aus Abschnitt 7.3.6 ergibt:

$$d = \min\left(\frac{\pi}{2} \frac{l_f}{\tan(\lambda_{\max})}, d_{\text{obstacle}}\right). \quad (7.6)$$

Rückwärts- und Vorwärtzüge werden solange iteriert, bis die Soll-Ausrichtung erreicht ist.

7.2.6 Zurücksetzen

Ist mittels der *Tentakel-Navigation* kein Fortkommen mehr möglich, ermöglicht diese Verhaltensweise dem Fahrzeug durch gerades *Zurücksetzen* neue Perspektiven für die Vorwärtsfahrt. Hierzu wird analog zu Abschnitt 7.2.5 ein – diesmal gerades – Klothoidensegment mit einer Standardlänge von $l_{\text{backward}} = 10\text{ m}$ generiert und mittels der *Tentakel* aus Abschnitt 7.3.6 auf Hindernisse überprüft. Dieses Segment wird langsam mit $v_{\text{soll}} = -2\text{ m/s}$ befahren, maximal bis zum ersten erkannten Hindernis oder dem Ende des Segments:

$$d = \min(l_{\text{backward}}, d_{\text{obstacle}}). \quad (7.7)$$

Dieser Verhaltensschritt wird wiederholt, wenn sich nach dem *Zurücksetzen* noch keine neuen Perspektiven in der Vorwärtsfahrt ergeben haben.

7.2.7 Lenkwinkelanpassung im Stand

Diese Verhaltensweise kommt zum Einsatz, um größere Änderungen des Lenkwinkels durchzuführen, ohne dabei durch gleichzeitige Längsbewegung größere Abweichungen von der Soll-Klothoide aufzubauen. Falls notwendig, wird das Fahrzeug zunächst bis zum Ende des Klothoidensegments in den Stillstand abgebremst, wobei es mittels aktiver Querregelung der zuletzt aktiven Klothoide weiter folgt. Anschließend wird der Soll-Lenkwinkel mithilfe eines Proportionalreglers und limitierter Lenkrate eingestellt.

Diese Verhaltensweise *Lenkwinkelanpassung im Stand* kommt häufig als temporäres Verhalten in der *Tentakel-Navigation* oder beim *Wenden* zum Einsatz, wenn es eine starke Krümmungsänderung oder einen Krümmungssprung in der nachfolgend ausgewählten Klothoide gibt.

7.2.8 Gridplanning-Navigation

Bei der Verhaltensweise *Gridplanning-Navigation* handelt es sich um eine globale Pfadplanung mit größerem Planungshorizont, anstelle der reaktiven *Tentakel-Navigation*. Der Einsatzzweck ist weiterhin das unstrukturierte Umfeld und enges Gelände.

In dieser Verhaltensweise kommt das Wahrnehmungsmodul *Gridplanning* zum Einsatz. Dieses erhält dieselben Eingangsdaten wie die *Tentakel*: akkumulierte Hinderniskarten (vgl. Abschnitt 7.3.2.2), die geplante Sollspur $\mathcal{P}_{\text{routeAnnotated}}$ zu einem bestimmten Zielpunkt $\mathbf{p}_{\text{target}}$, sowie die aktuelle Ego-Lokalisierung.

In jedem Zeit- bzw. Planungsschritt wird eine Vielzahl möglicher Trajektorien ausgewertet und die unter mehreren Kriterien optimale ausgewählt. Daraus kann die Fahrspur-Schnittstelle \mathcal{L} befüllt werden.

Die Verhaltensweise *Gridplanning-Navigation* kam ab der euRathlon 2013 für das *Autonomous Navigation*-Szenario und auf den folgenden M-ELROBs 2016 und 2018 im *Mule*-Szenario zum Einsatz.

7.2.9 Anhalten und Not-Halt

Diese einfache Verhaltensweise dient zwei Zwecken: Einerseits dient es zum geordneten Anhalten, d. h. eine Geschwindigkeitsreduktion bei weiterhin aktivierter Querführung auf einer Klothoide und Einhaltung eines sicheren Abstands zu getrackten Objekten, bspw. dem Führungsfahrzeug im Konvoi oder einem Fremdfahrzeug bei der Navigation auf einem Straßennetz.

Andererseits – wenn bspw. nach Ausfall eines Wahrnehmungsmoduls keine gültige Klothoide mehr vorliegt – wird eine Notbremsung bei konstant gehaltenem Lenkwinkel durchgeführt, um das Fahrzeug in einen sicheren Zustand zu überführen.

Nach einer Stillstandsphase von $\Delta t_{\text{standstill}} \geq 10\text{ s}$ erfolgt ein Übergang zur Verhaltensweise *Fahrstufenwechsel*.

7.2.10 Fahrstufenwechsel

Diese Verhaltensweise dient dem *Fahrstufenwechsel* basierend auf der Soll-Fahrtrichtung. Der Fahrstufenwechsel wird bei den institutseigenen Versuchsträgern durch eine elektromechanische Betätigung des Wählhebels des Automatikgetriebes erreicht, bei anderen Fahrzeugen geschieht dies über eine digitale Ansteuerung des Getriebe-Steuergeräts. Für die Soll-Fahrtrichtung *vorwärts* wird die Fahrstufe *Drive* genutzt, für *rückwärts* dementsprechend *Reverse*. Ein Sonderfall ist die Fahrstufe *Park*, die in längeren Stillstandsphasen von $\Delta t_{\text{standstill}} \geq 10\text{ s}$ zur Schonung der Bremsaktorik eingelegt wird.

7.3 Wahrnehmungs- und Navigationsmodule

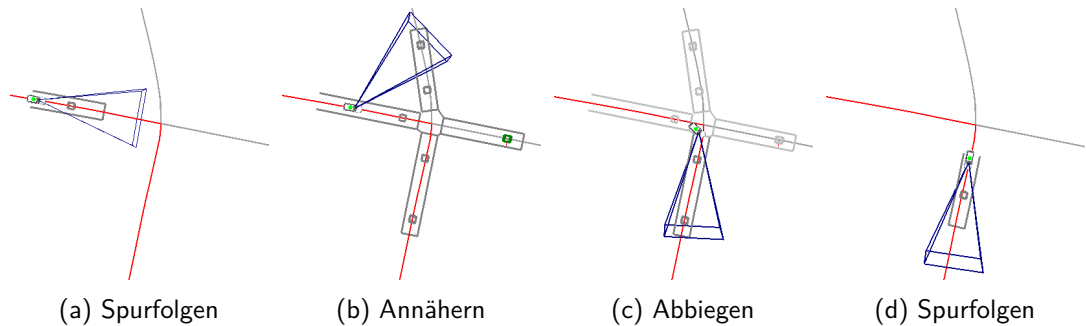
In diesem Abschnitt werden die Wahrnehmungs- und Navigationsmodule vorgestellt, die zur Realisierung der Verhaltensweisen verwendet wurden. Sie wurden federführend von den Kolleginnen und Kollegen am Institut entwickelt, siehe entsprechende Referenzen, und gemeinsam für die Verwendung in dem in dieser Arbeit entwickelten Gesamtsystem adaptiert und getestet. Für eine Gesamtintegration und Austauschbarkeit einzelner Module wurden einheitliche Schnittstellen definiert.

Entscheidende eigene Beiträge in diesem Abschnitt sind neben der Schnittstellendefinition die Ground-Truth-Bereitstellung im dynamischen Umfeld in Abschnitt 7.3.3.5, die kartenbasierten Rückfallebenen in Abschnitt 7.3.4.3 und 7.3.5.2 sowie die Weiterentwicklung der Tentakel unter besserer Berücksichtigung der Fahrzeuggeometrie und der Bewegungsmöglichkeiten des Fahrzeugs sowie einer ortsabhängigen Gewichtung der Karteninformation in Abschnitt 7.3.6.

7.3.1 Blickrichtungssteuerung

Vor allem in mobilen Versuchsträgern ist die verfügbare Rechenleistung durch die Energieversorgung aus der Batterie bzw. Lichtmaschine sowie durch den Bauraum limitiert. Um Systemressourcen (Rechenzeit, Bandbreite, Speicherplatz) zu sparen, wurde bei MuCAR-3 und MuCAR-4 lange auf eine 360°-Abdeckung mit hochauflösenden Kameras verzichtet.

Stattdessen kommt von Anfang an die aktive Kameraplattform MarVEye-8 zum Einsatz: Diese ist mit einer Tele- und im Regelfall zwei Weitwinkelkameras ausgestattet,

**Abbildung 7.3:**

Aufmerksamkeitssteuerung während eines Rechtsabbiegevorgangs: Die geplante Route ist rot markiert. Die Geometrie des Straßennetzes aus der Karte ist als dunkelgrauer Polygonzug dargestellt. Fahrspurobjekte sind als graue Quadrate visualisiert. Je nach Phase sind irrelevante Bereiche in hellgrau gekennzeichnet. Quelle: [Unterholzner, 2016, Fig. 3.9]

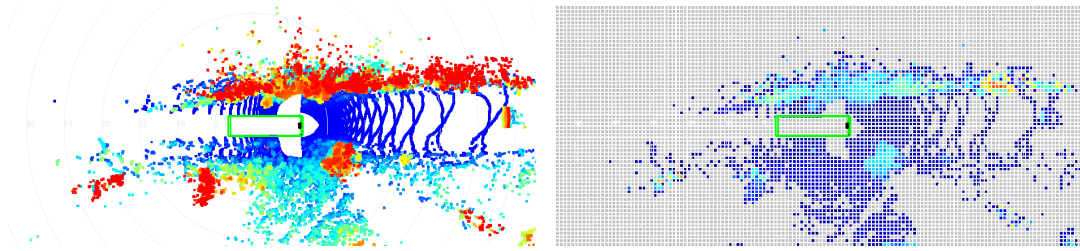
welche zwar pro Kamera nur ein begrenztes Sichtfeld bieten, aber aufgrund der aktuierten Kameraplattform trotzdem einen breiten Bereich vor dem Fahrzeug abdecken können. Eine kurze Vorstellung der Hardware findet sich in Abschnitt 3.2.3.3, für weitergehende Details zu Aufbau und Regelung sei auf [Unterholzner und Wuensche, 2009, 2010, 2012] sowie auf die Dissertation von Unterholzner [2016] verwiesen.

Um mit dem begrenzten Sichtfeld die für die aktive Verhaltensweise relevanten Bereiche der Umgebung abzudecken, wurden mehrere Varianten entwickelt, die von extern ausgewählt werden können:

Lenkwinkelabhängig Besteht keine weitere Information aus der Umgebungserfassung, so wird die Kameraplattform auf einen imaginären Punkt in einer einstellbaren Vorausschauweite ausgerichtet, welche auf einer dem aktuellen Lenkwinkel entsprechenden Klothoide liegt. Dieser Modus wird primär bei manueller Fahrt genutzt, z. B. bei der Aufzeichnung von Datensätzen für die spätere Evaluierung im Labor.

Externe Winkelvorgabe Alternativ kann die Blickrichtung von der aktiven Verhaltensweise direkt durch Winkelvorgaben eingestellt werden.

Aufmerksamkeitssteuerung Eine am Institut entwickelte Aufmerksamkeitssteuerung [Unterholzner, 2016] dient zur automatischen Bestimmung der optimalen Blickrichtung. Hierzu werden die bereits getrackten Objekte mitsamt ihren Schätzfehlerkovarianzmatrizen ebenso betrachtet wie potentiell neu in den Sichtbereich kommende, sog. *unentdeckte artifizielle Objekte*, die an mithilfe von Kartenwissen bestimmten Orten – beispielsweise auf den in Abschnitt 7.3.1 dargestellten Kreuzungsästen – erscheinen können. Zusätzlich werden sog. *Fahrspurobjekte* betrachtet. Durch Simulation möglicher Blickrichtungen und deren Auswirkung auf die Entwicklung der Schätzfehler wird die optimale Blickrichtung für die kommenden Zyklen bestimmt. Diese Arbeiten wurden von Pellkofer et al. [2003] am Vorgängerlehrstuhl von Prof. Dickmanns inspiriert.



(a) 3D-Punktwolke auf Basis zweier Velodyne HDL-32E LiDAR-Sensoren an den vorderen Fahrzeugecken des TULF (grünes Rechteck). Die Höhe (z -Koordinate) ist als Heatmap farbkodiert.

(b) Aus (a) resultierende Belegungskarte. Die Belegungswahrscheinlichkeit ist als Heatmap farbkodiert. Zellen, auf die weniger als zwei Punkte abgebildet wurden und für die deshalb keine Belegtheit bestimmt werden kann, sind grau dargestellt.

Abbildung 7.4:

Beispiele für einfache, ego-zentrierte Hinderniskarten und ihre Erzeugung aus LiDAR-Messungen. Quelle: [Lüttel et al., 2014, Abb. 51].

7.3.2 Rasterkarten

In Anhang B.1.2 werden die metrischen 2D-Rasterkarten, engl. Grid Maps, eingeführt. In der Robotik kommt diese Kartenart häufig als Abstraktionsebene für die Rohdaten hochauflösender 3D-Sensoren wie LiDAR, Stereokameras oder Radar (Radio Detection And Ranging) zum Einsatz. Diese Rasterkarte dient nachgelagerten Wahrnehmungs- und Navigationsalgorithmen als Eingang, bspw. in Form einer Hinderniskarte (engl. Obstacle Map) oder Belegungskarte (engl. Occupancy Map). Die Abstraktion bzw. Vorverarbeitung dient zwei Zielen: Einerseits ist diese Darstellungsart unabhängig vom verwendeten Ausgangssensor, wodurch oft auf Anpassungen in den nachgelagerten Algorithmen verzichtet werden kann. Andererseits ermöglicht sie eine kontrollierte Datenreduktion der großen Punktwolken in eine Rasterkarte, deren Auflösung und Größe dem Anwendungsfall angepasst werden kann.

In der klassischen Belegungskarte wird pro Zelle eine Hinderniswahrscheinlichkeit gespeichert [Thrun, 2002]. Die Rasterkarten kommen oft auch dergestalt zum Einsatz, dass pro Gitterzelle mehrere Werte gespeichert werden, z. B. Hinderniswahrscheinlichkeit, Rauigkeit und Farbe. Als 2½ D-Belegungskarten werden hier Karten bezeichnet, die nicht nur eine Belegungswahrscheinlichkeit enthalten, sondern die dritte Dimension durch die Angabe von kleinster und größter z -Koordinate approximieren (vgl. Abbildung 7.7b). Damit kann eine Hindernishöhe abgebildet werden, was für die Bahnplanung von Landfahrzeugen meist ausreicht, statt die wesentlich aufwendigeren 3D-Voxelkarten zu verwenden.

Im Folgenden werden zwei Varianten der Kartenerzeugung vorgestellt – einfache und akkumulierte Rasterkarten.

7.3.2.1 Einfache Rasterkarten

Zur Erzeugung der einfachen Rasterkarten kommen Messungen aus einem Zeitschritt zum Einsatz, bspw. die 3D-Punktwolke aus einer vollständigen Umdrehung von einem oder mehreren LiDAR-Sensoren oder aus einem synchronen Bildpaar eines Stereokamerasystems.

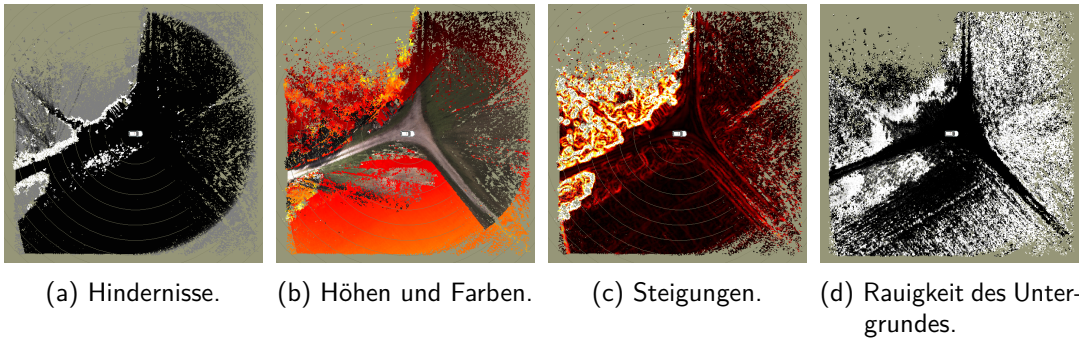
Zur Erzeugung der 3D-Punktwolke aus den Messungen gibt es zwei unterschiedliche Vorgehensweisen, die abhängig vom Verwendungszweck der Rasterkarte gewählt werden: Für den Einsatz in enger, statischer Umgebung wird die Eigenbewegung des Roboters bzw. des Sensors ggü. der Welt kompensiert – die sog. *inertiale Korrektur* –, bspw. für die Tentakel-Navigation in Abschnitt 7.3.6. Auf die Eigenbewegungskompensation wird dagegen häufig verzichtet, wenn der Fokus eher auf dem Tracking eines vorausfahrenden Objekts liegt, z. B. für Konvoi-Anwendungen, siehe Abschnitt 7.3.3.3. Jeder 3D-Punkt der Punktwolke wird anhand seiner (x, y) -Koordinaten einer Zelle der Rasterkarte zugeordnet, anschließend werden für jede Zelle die Merkmale berechnet. Wichtigstes Merkmal ist die Belegungswahrscheinlichkeit, die sich aus den Höhendifferenzen ergibt [von Hundelshausen et al., 2008, Kap. 2] und initial auf 0,5 gesetzt wird [Jaspers et al., 2017].

Die Abbildungen 7.4a und 7.4b zeigen die Erzeugung einer einfachen Rasterkarte am Beispiel des TULF (Technologieträger Unbemanntes Landfahrzeug), bei der auf die inertielle Korrektur verzichtet wurde.

Einfache Rasterkarten haben jedoch mehrere Nachteile: So besteht schon in der 3D-Punktwolke des LiDAR-Sensors ein mit dem Fahrzeug mitbewegter „blinder Fleck“ oder Totbereich rund um den Sensor, der mit der Einbauhöhe des Sensors wächst. Weitere blinde Flecken entstehen durch Selbstverdeckung, bspw. durch die Karosserie. Diese Totbereiche pflanzen sich in die Rasterkarte fort (siehe Abbildung 7.4). Ein probates Mittel, diese Totbereiche zu verkleinern, ist der Einsatz zusätzlicher Sensoren, wie die vier Ecksensoren am TULF oder die Ouster-Totwinkel-LiDAR-Sensoren an MuCAR-3, MuCAR-5 und dem StreetScooter im MORE-Projekt.

7.3.2.2 Akkumulierte Rasterkarten

Akkumulierte Rasterkarten begegnen dem Problem der blinden Flecken auf andere Art: viele Scans der LiDAR-Sensoren werden unter Berücksichtigung der Fahrzeugbewegung iterativ über der Zeit in einer Karte integriert. Aus dieser reichhaltigen Information lassen sich wesentlich mehr Informationen als aus den einfachen Rasterkarten gewinnen, sodass die akkumulierten Rasterkarten als gemeinsame Repräsentation der statischen Umgebung zum Einsatz kommen: für die Wahrnehmungskomponenten *Erkennung und Tracking von Wegen und Kreuzungen* für die Fahrt im nicht-urbanen Raum, sowie für die *Tentakel* und das *Gridplanning* für die Navigation in unstrukturierter Umgebung.

**Abbildung 7.5:**

Mehrere Ebenen aus dem multimodalen Umgebungsmodell, jeweils rückprojiziert in eine Ego-zentrierte Rasterkarte. Die Szene zeigt eine T-Kreuzung am Waldrand. In (a) korrespondiert die Helligkeit mit der Hinderniswahrscheinlichkeit. Unbeobachtete Bereiche sind grau-beige eingefärbt. (b) ist entsprechend der Kamerainformationen eingefärbt, wo diese verfügbar sind, andernfalls wird die Höhe der Hinderniszellen als Heatmap farbig kodiert. In (c) sind die Steigungen als Heatmap farbig kodiert. In (d) korrespondiert die Helligkeit mit der Rauigkeit des Untergrundes. Quelle: [Jaspers et al., 2017, Fig. 3].

Kartenerzeugung: Akkumulation und Wrapping Die Eingangsdaten entsprechen Abschnitt 7.3.2.1, jedoch aus mehreren Zeitschritten. Zusätzlich zu diesen 3D-Informationen können Informationen aus diversen Kameras integriert werden, bspw. RGB-Farbinformationen (Rot-Grün-Blau), Intensitäten aus dem NIR- (Nahes Infrarot, engl. *Near-Infrared*) oder Temperaturinformationen. Die extrinsische Kalibrierung der Sensoren zueinander [Himmelsbach et al., 2011] und ihre Synchronisierung [Schneider et al., 2010] sind essentiell, siehe auch Abschnitt 4.2.

Ähnlich zur Erzeugung der einfachen Rasterkarte wird bei der akkumulierten Rasterkarte die 3D-Punktwolke zuerst inertial korrigiert, anschließend werden die 3D-Punkte den Zellen der Rasterkarte zugeordnet [Himmelsbach et al., 2009b, Sec. IV.A] [Himmelsbach et al., 2011, Sec. 2]. Für die Behandlung überhängender Hindernisse wie Äste oder negativer Hindernisse wie Löcher sei auf [Jaspers et al., 2017, Sec. III.B.1-3] verwiesen. Um nicht eine stetig mit der Bewegung des Ego-Fahrzeugs wachsende akkumulierte Rasterkarte zu erhalten, wird die Kartengröße mittels *Wrapping* begrenzt [Himmelsbach et al., 2009b, Sec. IV.B]. Für die Nutzung in den nachgelagerten Algorithmen wird die Karte in eine Ego-zentrierte Karte transformiert.

Ebenen Im multimodalen Umgebungsmodell verfügt jede Zelle über unterschiedliche Informationen, die, wie in Abbildung 7.5 gezeigt, als *Ebenen* der Rasterkarte bereitgestellt werden.

Für detaillierte Informationen zur Erzeugung dieser weiteren Ebenen wird auf [Jaspers et al., 2017] und [Himmelsbach, 2015] verwiesen. Neben dem eigentlich Kartierungsmodul *environment-mapping* können optional andere Module weitere Kartenebenen beisteuern, z. B. Straßenwahrscheinlichkeiten [Bayerl, 2015, Kap. 4.1] oder in die Rasterkarte projizierte globale Karteninformationen [Heinrich et al., 2018, Sec. 4.1.2].

Verwendung und Einschränkungen Grundsätzlich ist die Akkumulation von 3D-Punkten über der Zeit eher für statische Szenarien geeignet. In dynamischen Szenen hat die Akkumulation allerdings Probleme, da dann dasselbe Objekt über der Zeit an unterschiedlichen Orten gesehen wird und einen „Schweif“ hinterlässt – die Messungen fallen in unterschiedliche Zellen und erhöhen dort die Hinderniswahrscheinlichkeit. Diese Hindernisse verschwinden entweder durch Freimessen, durch Vergessen oder durch gezielte Rückführung von Listen getrackter Objekte in die Kartierung [Himmelsbach, 2015, Kap. 3.7] [Forkel et al., 2021, Kap. 3].

7.3.3 Tracking dynamischer Objekte

Einführung Die rekursive Schätzung der Zustände eines dynamischen Objekts, basierend auf Sensordaten resp. den darauf beruhenden Messungen sowie einem dynamischen Modell für die Veränderung der Zustände über der Zeit, wird als *Objekttracking* bezeichnet.

Die Bezeichnung *4D-Ansatz* sowie die Ideen für diese räumlich-zeitliche Modellierung zur dreidimensionalen Erfassung von Bewegungen mit 2D-Kerasensoren gehen auf Konzepte von Dickmanns [1980, 1981] zu Beginn der 1980er Jahre zurück und haben nicht nur in der Robotik und dem Automobilbereich starke Verwendung gefunden. In seiner Dissertation nutzte Wünsche [1987] dieses Konzept der Zustandsschätzung u. a. für ein Andockmanöver eines Satellitenmodells. Dickmanns [2007] fasst sämtliche Nutzungen im Bereich des autonomen Fahrens am Vorgängerinstitut ISF zusammen.

Im Objekttracking können neben der 6-DOF-Pose eines Objekts weitere statische Parameter wie seine Geometrie sowie dynamische Zustände wie seine Geschwindigkeit, die Gierrate oder der Lenkwinkel anhand einer Folge von Messungen geschätzt werden. Diese Werte bilden zusammen den Zustandsvektor \mathbf{x} . Neben dem Zustandsvektor wird auch die Kovarianzmatrix des Schätzfehlers \mathbf{P} berechnet, welche sich aus der Kovarianzmatrix der Messungen \mathbf{R} und der Kovarianzmatrix des Systemfehler \mathbf{Q} ergibt und anhand derer eine Aussage über die Genauigkeit der Schätzung getroffen werden kann.

Als Eingangsdaten für das Objekttracking seien hier unverarbeitete 3D-Punktwolken aus einem LiDAR-Sensor oder einer Stereokamera, aber auch daraus abgeleitete nicht-akkumulierte 2D- oder 2½ D-Rasterkarten aus Abschnitt 7.3.2.1 genannt. Neben LiDAR-Daten werden auch Messungen im 2D-Kamerabild oder Radar-Detektionen verwendet. Weitere Sensoren liefern bereits intern vorverarbeitete Daten und damit Objekthypothesen.

Abhängig vom Einsatzzweck und den zur Verfügung stehenden Sensoren am jeweiligen Versuchsfahrzeug ergibt sich eine Vielzahl an möglichen Trackingverfahren, von denen einige in den Abschnitten 7.3.3.1 bis 7.3.3.4 vorgestellt werden.

Rückblick: Rekursive Schätzverfahren Für die rekursive Zustandsschätzung im Objekttracking kommen oft Derivate des KF (Kalman-Filter) wie das EKF (Erweitertes Kalman-Filter) oder das UKF (Unscented Kalman-Filter) zum Einsatz. Diese wurden im Abschnitt 5.4 „Schätzverfahren“ bereits für eine andere Anwendung eingeführt.

Ein anderer oft verwendeter Filtertyp ist das Partikel-Filter (PF), eine sequentielle Monte-Carlo-Methode, die insbesondere bei multimodalen, nicht-gauß'schen Verteilungen ihre Stärken hat und mit nichtlinearen System- und Messmodellen umgehen kann. Für weitere Details zum PF sei hier auf das Lehrbuch von Thrun et al. [2005] verwiesen.

Allen Filtern gemeinsam ist die Aufteilung in zwei Schritte, Prädiktion und Korrektur, die abwechselnd ausgeführt werden. Für die Prädiktion des Zustandsvektors \mathbf{x}_k vom Zeitpunkt t_k der letzten Messung bis zum Zeitpunkt t_{k+1} der nächsten Messung \mathbf{y}_{k+1} wird ein dynamisches Modell $f(\cdot)$ für das getrackte Objekt benötigt, mit dem dessen Bewegung vorhergesagt werden kann:

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k) .$$

Für die Modellierung der Bewegung eines Führungsfahrzeugs kommen häufig Einspurmodelle zum Einsatz (vgl. Abschnitt 5.5.3), für die Bewegung einer Person beispielsweise allgemein gehaltene Punktmassenmodelle wie CTRV (Konstante Gierrate und Geschwindigkeit, engl. *Constant Turn Rate and Velocity*) oder CTRA (Konstante Gierrate und Beschleunigung, engl. *Constant Turn Rate and Acceleration*) (vgl. Abschnitt 5.5.2). Andere Parameter wie die Geometrie sind oft als konstant modelliert. Im Korrekturschritt werden die Sensordaten (oder daraus abgeleitete Merkmale) als Messung

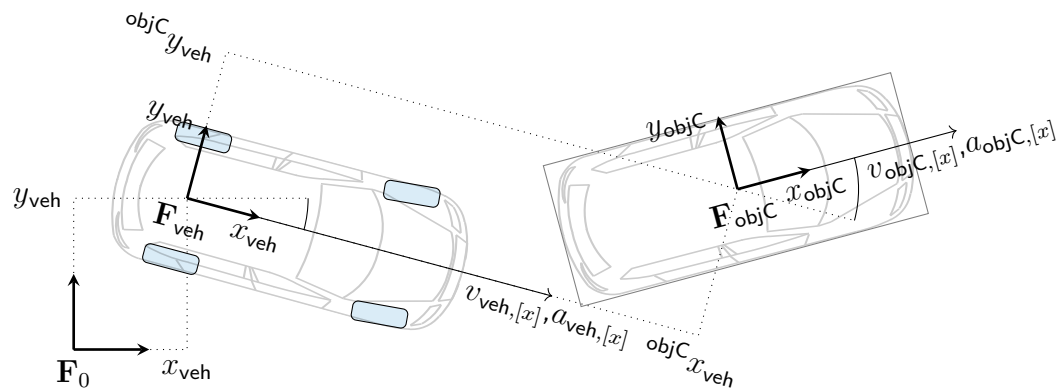
$$\mathbf{y}_k = g(\mathbf{x}_k, \mathbf{u}_k)$$

verwendet, um den Zustand zu aktualisieren.

3D-Objektschnittstelle Die unterschiedlichen Trackingalgorithmen kommunizieren ihre jeweiligen Ergebnisse – unabhängig von der Implementierung des Algorithmus und dem Aufbau des internen Zustandsvektors \mathbf{x} – über eine im Rahmen dieser Arbeit definierte einheitliche 3D-Objektschnittstelle, die in Tabelle 7.1 definiert ist. Wichtigster Parameter sind die Pose des Objekts sowie dessen Geschwindigkeit und Geometrie. Abbildung 7.6 visualisiert die Koordinaten und am häufigsten verwendeten Parameter, reduziert auf den 2D-Fall.

7.3.3.1 LiDAR-basiertes Objekttracking

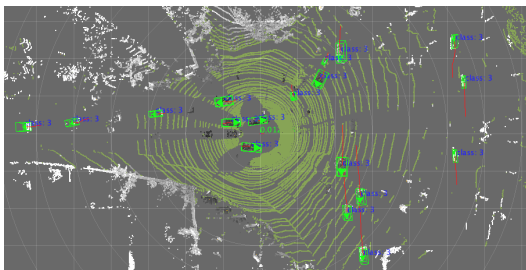
Die 3D-Punktewolke von LiDAR-Sensoren wie dem Velodyne HDL-64E S2 bietet auf den meisten Versuchsträgern eine Rundumsicht um das Fahrzeug und ist gut geeignet, um andere Objekte zu detektieren und zu verfolgen. Die Verarbeitungskette besteht hier im Regelfall aus einer Segmentierung der Punktewolke [Himmelsbach

**Abbildung 7.6:**

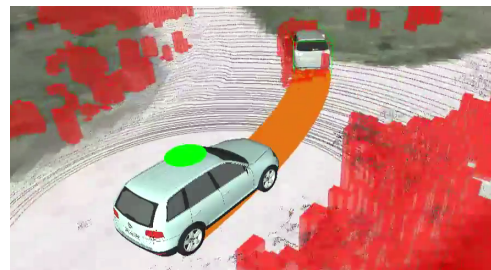
Übersicht über die Koordinaten in der Objektschnittstelle, vereinfacht auf den 2D-Fall. Links das Ego-Fahrzeug, rechts das getrackte Objekt.

Tabelle 7.1: Definition der Objektschnittstelle.

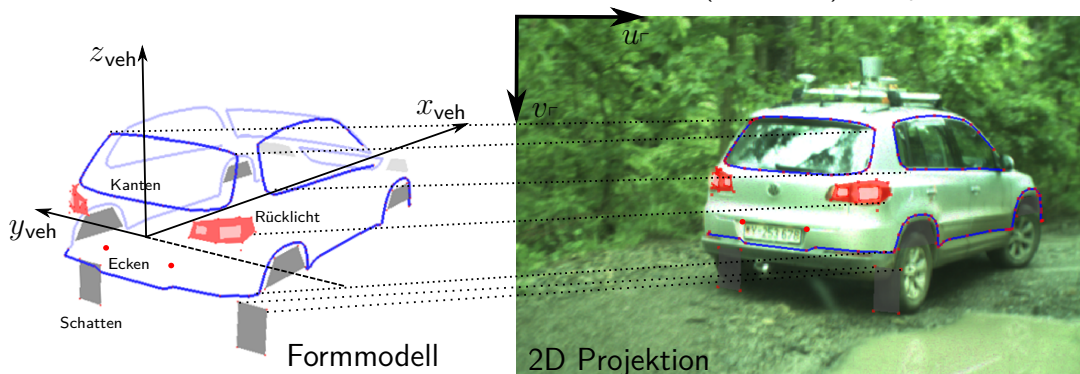
Bezeichnung	Variable	Bemerkung
6-DOF-Pose ${}^{\text{objC}}\mathbf{p}_{(\cdot)}$	$(x, y, z, \Phi, \Theta, \Psi)^T$	Pose des Objektreferenzpunktes (Zentrum der 3D-Bounding-Box)
Bezugssystem	–	Bezugssystem der 6-DOF-Pose: \mathbf{F}_{veh} , \mathbf{F}_{DR} , \mathbf{F}_0 oder andere.
6-DOF-Geschwindigkeit	$(v_{[x]}, v_{[y]}, v_{[z]}, \dot{\Phi}, \dot{\Theta}, \dot{\Psi})^T$	in \mathbf{F}_{objC} .
6-DOF-Beschleunigung	$(a_{[x]}, a_{[y]}, a_{[z]}, \ddot{\Phi}, \ddot{\Theta}, \ddot{\Psi})^T$	in \mathbf{F}_{objC} . Selten gefüllt, da a in den Schätzern entweder nicht als Zustand modelliert ist oder nur Rauschterm ist. Von der Ground-Truth gefüllt.
Geometrie	$(l, w, h)^T$	Abmessungen der Bounding-Box (Länge, Breite, Höhe)
Unsicherheiten	\mathbf{P}^{Pose} , \mathbf{P}^{Vel} , \mathbf{P}^{Acc} , σ_l , σ_w , σ_h	meist nur Diagonalelemente der Kovarianzmatrizen gefüllt
Klassifikation	–	Angabe über den Objekttyp (wenn verfügbar) und dessen Sicherheit



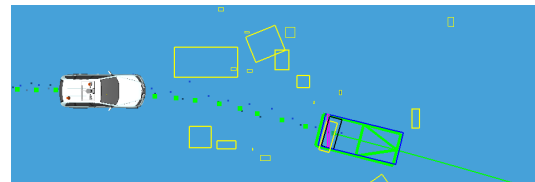
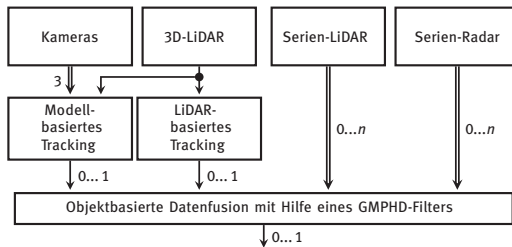
(a) LiDAR-basiertes Tracking vieler Fahrzeuge an einer Kreuzung (grüne Boundingboxen mit Richtungspfeil, rote Linien visualisieren den zurückgelegten Weg der letzten Sekunden). Bildquelle: TAS/Himmelsbach



(b) Verwendung der LiDAR-Daten im Fahrzeugtracking: 3D-Punktwolke (schwarz), 2½ D-Belegungskarte (rot, die Höhe der Säulen entspricht der maximal gemessenen Höhe in einer Zelle) und geschätzte Pose (3D-Modell). Bildquelle: TAS



(c) Projektion aller charakteristischen Merkmale aus dem 3D-Merkmalmodell in die Bildebene
Bildquelle: [Manz, 2013, Seite 120]



(d) Schematische Übersicht der OBDF. Bildquelle: [Luettel et al., 2017, S. 500]

(e) Messungen und Fusionsergebnis der OBDF. Bildquelle: [Luettel et al., 2017, S. 500]

Abbildung 7.7:

Beispiele für unterschiedliche Varianten des Objekttrackings: (a) zeigt ein LiDAR-basiertes Tracking. (b) und (c) zeigen Ergebnisse und Funktionsweise des fusionierten modellbasierten Fahrzeugtrackings. (d) und (e) zeigen die objektbasierte Datenfusion des Kollegen Jan Kallwies. In der schematischen Übersicht (d) stellt jeder Pfeil die Anzahl der bereitgestellten Objekthypothesen dar. (e) zeigt das Fusionsergebnis (grün, mit Historie und prädikierter Fahrtrichtung) und Messungen der einzelnen Sensoren bzw. Tracker: LiDAR-Tracking (schwarz), Fusioniertes modellbasiertes Tracking (blau), 8-Ebenen-Serien-LiDAR (gelb, extrem viele Tracks unterschiedlichster Objekttypen vom Grasbüschel bis zum Fahrzeug), Serien-Radar (magenta).

et al., 2010], einer Klassifikation der Segmente [Himmelsbach et al., 2009a] und darauf aufbauend einem Multi-Objekt-Tracking [Himmelsbach und Wuensche, 2012], wie in Abbildung 7.7a gezeigt.

Auf diesen Ergebnissen aufbauend, wurde das Tracking um eine Konturschätzung der Objekte erweitert [Naujoks und Wuensche, 2018, Naujoks et al., 2018, 2019]. Aber auch im Bereich der Detektion gibt es Weiterentwicklungen [Reich und Wuensche, 2022], wie die Detektion mithilfe von dynamischen Partikeln. Die Objektschnittstelle wurde jeweils beibehalten.

Für Anwendungen im Konvoi, bei denen der Fokus auf dem robusten Tracking eines Führungsfahrzeugs liegt, erfolgt häufig eine andere Parametrierung der Algorithmen als für die Multi-Objekt-Tracking-Anwendung.

7.3.3.2 Visuelles modellbasiertes Fahrzeugtracking

Neben dem LiDAR-Sensor eignen sich auch passive Kamera-Sensoren zur Detektion und zum Tracking anderer Objekte. Das modellbasierte monokulare Fahrzeugtracking nutzt eine einzelne Farbkamera in einem Partikelfilter-Framework [Manz et al., 2011a] [Manz, 2013, Kapitel 5], um ein spezielles Fahrzeug, welches durch einzelne Features (siehe Abbildung 7.7c) beschrieben wird, robust auch in unstrukturierten Offroad-Umgebungen zu erkennen und zu verfolgen. Eine entscheidende Einschränkung für das rein visuelle Tracking ist der begrenzte horizontale Öffnungswinkel der Kameras, die durch die Blickrichtungssteuerung nur teilweise kompensiert werden kann, sodass es in engen Kurven leicht zu Sicht- und damit Trackingverlusten kommt.

7.3.3.3 Fusioniertes modellbasiertes Fahrzeugtracking

Manz [2013] erweiterte seinen rein kamerabasierten Ansatz um zusätzliche Partikel-Bewertungen anhand einer LiDAR-basierten nicht-akkumulierten $2\frac{1}{2}$ D-Rasterkarte, wie in Abbildung 7.7b gezeigt. Durch diese Sensordatenfusion konnte die Robustheit des Trackers in den für die visuelle Wahrnehmung problematischen Bereichen, z. B. bei Überblendung durch Gegenlicht oder bei wechselnden Lichtverhältnissen im Wald oder auch in engen Kurven, massiv gesteigert werden. In [Fries et al., 2013] und [Fries und Wuensche, 2014] wurde dieser Tracker um eine verbesserte Reinitialisierung mithilfe von 2D-Bild-Templates erweitert und in [Fries und Wuensche, 2015] eine Wärmebildkamera für das verbesserte Tracking bei Nacht integriert.

An dieser Stelle sei auf die Bedeutung synchronisierter Sensordaten von bewegter Kamera und LiDAR-Sensor verwiesen. Hier kommen die gemeinsam mit Schneider et al. [2010] erzielten Ergebnisse, die in Abschnitt 4.2 zusammengefasst werden, zum Einsatz.

7.3.3.4 Objektbasierte Datenfusion im Fahrzeugtracking

Im Rahmen der StrAsRob-Studie (Straßentransport mit Assistenzfunktionen von Robotern) [Bundesamt für Wehrtechnik und Beschaffung, 2012] wurde vom Kollegen Jan Kallwies ein Verfahren zur *OBDF* (*Objektbasierte Datenfusion*, engl. *Object Based Data Fusion*) entwickelt, welches zusätzlich zu den Ergebnissen der oben beschriebenen Trackingverfahren Messdaten von seriennaher LiDAR- und Radar-Sensorik integriert und in Abb. 7.7d und 7.7e dargestellt ist. Ziel ist ein möglichst präzises und gleichzeitig robustes Tracking eines Führungsfahrzeugs im Konvoi. Die OBDF basiert auf dem GMPHD-Filter (engl. *Gaussian Mixture Probability Hypothesis Density*) [Vo und Ma, 2006], einem Multi-Hypothesen-Tracker mit impliziter Datenassoziation. Details zur Modellierung, vor allem der sehr sensor- und fahrzeugabhängigen Messungen, sowie Ergebnisse finden sich in [Luettel et al., 2017, Lüttel et al., 2015].

7.3.3.5 INS-basierte Objekt-Ground-Truth

Um die Entwicklung von Verhaltensweisen, die ein Tracking dynamischer Objekte benötigen, unabhängig von Algorithmen zum sensorbasierten Tracking vorantreiben zu können und Closed-Loop-Tests zu ermöglichen, wurde im Rahmen dieser Arbeit eine Rückfallebene in die Versuchsfahrzeuge integriert.

In den Versuchsträgern MuCAR-3/4/5 kommt jeweils ein INS (Inertiales Navigationssystem, engl. *Inertial Navigation System*) vom Typ OxTS RT3003 v1 oder vergleichbares Gerät desselben Herstellers (siehe Abschnitt 3.2.3.2) zum Einsatz, das unter optimalen Bedingungen (Korrekturdaten von eigener RTK-Referenzstation (Echtzeitkinematik, engl. *Real Time Kinematic*) oder mittels NTRIP-Modems (*Networked Transport of RTCM via Internet Protocol*)) eine Zentimeter-genaue globale Lokalisierung bereitstellt. Weitere Fahrzeuge können für kurzfristige Messkampagnen ebenfalls mit diesen Sensoren ausgerüstet werden.

Die rohen Sensordaten werden zwischen den Versuchsfahrzeugen latenzarm mittels Long-Range-WLAN (Drahtloses lokales Netzwerk, engl. *Wireless Local Area Network*) übertragen und anhand des GNSS-Zeitstempels synchronisiert. Nach der Konvertierung beider Lokalisierungen von \mathbf{Y}_{LLH} nach \mathbf{F}_{UTM} wird die Relativlage des Führungsfahrzeugs im Ego-Koordinatensystem \mathbf{F}_{veh} berechnet. Die objektumhüllende Boundingbox wird aus den bekannten Fahrzeugparametern des jeweiligen Führungsfahrzeugs sowie der extrinsischen Einbaulage des INS bestimmt. Mit diesen Informationen wird die in Tabelle 7.1 definierte Objektschnittstelle befüllt.

Die Nutzung für Closed-Loop-Tests ist jedoch nur ein Nebenprodukt, die Hauptnutzung der hier vorgestellten INS-basierten Objektposition ist die Verwendung als sog. Ground-Truth zur Evaluierung der anderen Trackingverfahren.

7.3.4 Fahrspuren

Einführung Der menschliche Fahrer lässt sich beim Autofahren im Regelfall von den Straßen- oder Wegbegrenzungen leiten und wählt geeignete Lenkeingriffe, um das Fahrzeug parallel zur Spur und möglichst in deren Mitte zu führen. Diese optisch erkennbaren Leitstrukturen sind auch für autonome Fahrzeuge geeignet und können in den Sensordaten sowohl auf markierter Straße als auch auf Feldwegen erkannt und verfolgt werden. Abschnitt 7.3.4.1 geht auf die visuelle Erkennung und das Tracking von Wegen ein.

Stehen keine Fahrbahnbegrenzungen zur Verfügung, so nutzt der Mensch oft das Verhalten anderer Verkehrsteilnehmer und folgt ihnen – in der Annahme, dass diese den richtigen Weg bspw. durch eine Wiese wählen. Visuell könnten der Mensch – oder das autonome Fahrzeug – nun versuchen, den am Boden hinterlassenen Spuren zu folgen, oder sich den Weg anhand der Positionen des anderen Objekts über der Zeit zu merken. Diese Fahrspurgenerierung anhand von Objekten nutzt die Informationen aus Abschnitt 7.3.3 und wird in Abschnitt 7.3.4.2 weiter detailliert.

Im Folgenden wird zunächst die Fahrspur \mathcal{L} als Schnittstellenbeschreibung zwischen Wahrnehmung und Fahrzeugführung eingeführt, bevor neben den zwei aufgeführten Verfahren auch eine GNSS- & kartenbasierte Spurgenerierung als Rückfallebene vorgestellt wird (siehe Abschnitt 7.3.4.3).

Schnittstelle In der Schnittstelle zwischen Wahrnehmung (bzw. Planung) und Regelung werden Klothoidensegmente zur Beschreibung der Geometrie der Fahrspur sowie der Relativlage zwischen Fahrzeug und Fahrspur verwendet. In Abschnitt 6.2.1 wurden die Grundlagen der Klothoide bereits eingeführt, die Relativlage von Fahrzeug und Klothoide ist in Abbildung 6.3 visualisiert.

Tabelle 7.2 zeigt die vollständige, im Rahmen dieser Arbeit definierte Fahrspur-Schnittstelle \mathcal{L} , allerdings werden von vielen Wahrnehmungsmodulen nicht alle Parameter gefüllt.

In den folgenden Abschnitten werden unterschiedliche Wahrnehmungs- und Navigationsmodule präsentiert, die jeweils eine Klothoide bereitstellen.

7.3.4.1 Erkennung und Tracking von Wegen

Fahrspurschätzung unter Verwendung von 2D-Kamerabildern Die Algorithmen des Vorgängerinstituts ISF (Institut für Systemdynamik und Flugmechanik) von Dickmanns [2007] nutzten nur das 2D-Kamerabild zur visuellen Fahrspurdetektion, primär auf markierter Straße. Der sog. 4D-Ansatz von Dickmanns [1988] erfordert dabei keine vollständige Auswertung der Kamerabilder, sondern es können geeignete Methoden zur gezielten Messung an erwarteter Stelle verwendet werden, beispielhaft richtungsselektive ternäre Masken zur Kantenextraktion [Kuhnert, 1988, Mysliwetz, 1990]. Auch erste Arbeiten des Kollegen Manz [2008] zur Erkennung und dem

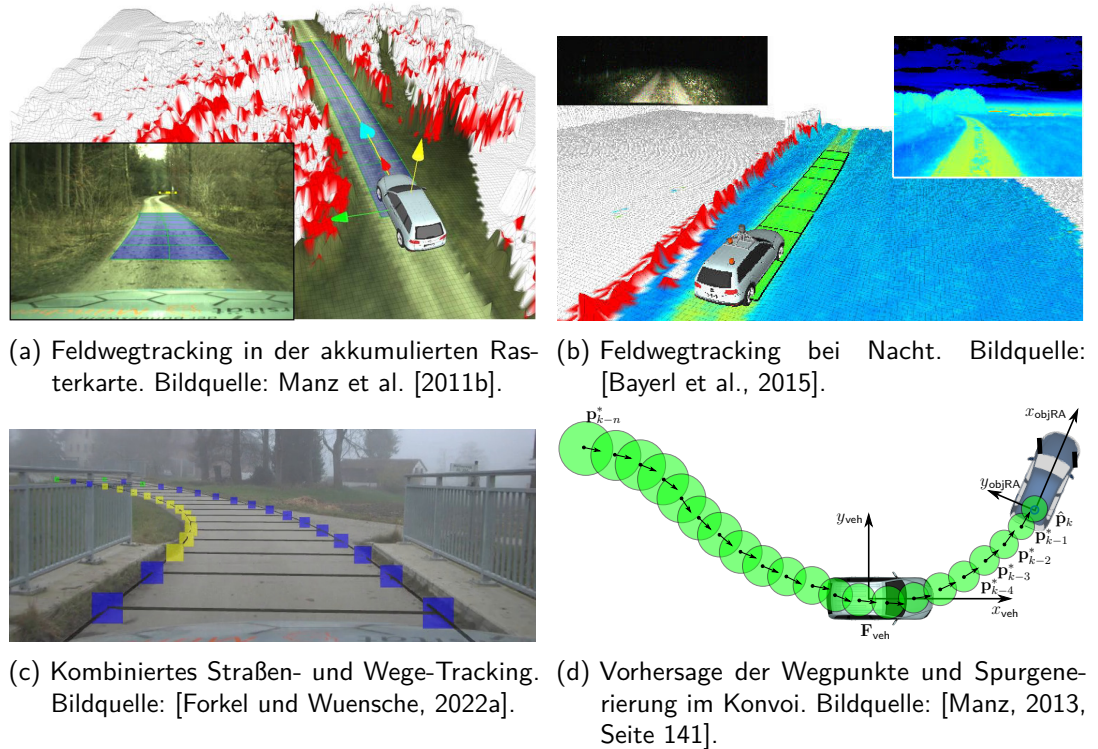
Tabelle 7.2:

Definition der Fahrspur-Schnittstelle \mathcal{L} . Die zwingend von allen Wahrnehmungsmodulen zu füllenden Parameter sind **fett** gedruckt. Unter „Änderung“ ist hier jeweils die Ableitung nach der Lauflänge zu verstehen, nicht nach der Zeit.

Bezeichnung	Variable	Bemerkung
Abstand des Ego-Fahrzeugs relativ zur Spur	d	
Winkel des Ego-Fahrzeugs relativ zur Spur	$\Delta\Psi$	
Horizontale Krümmung	c_0	
Horizontale Krümmungsänderung vorn	$c_{f,1}$	
Horizontale Krümmungsänderung hinten	$c_{b,1}$	
Vertikale Krümmung	$c_{\text{vertical},0}$	immer 0
Vertikale Krümmungsänderung	$c_{\text{vertical},1}$	immer 0
Breite	w_0	
Breitenänderung	w_1	
Trackingqualität	–	[0...1]
Spur-ID	–	0: rechte Spur, nach links hochzählend
Wunschgeschwindigkeit gültig	–	[0, 1]
Wunschgeschwindigkeit	v_{desired}	
Segmentlänge	l_{segment}	
Distanz entlang Spur zum ersten Objekt	$d_{\text{firstObject}}$	0, wenn kein Objekt existiert
Entfernung Spur-Startpunkt von Ego-X	–	normalerweise 0
Unsicherheit in d	σ_d^2	
Unsicherheit in $\Delta\Psi$	$\sigma_{\Delta\Psi}^2$	
Unsicherheit in c_0	$\sigma_{c_0}^2$	
Unsicherheit in c_1	$\sigma_{c_1}^2$	
Unsicherheit in w_0	σ_w^2	

Tracking von Straßen und Feldwegen in einem Kalman-Filter-basierten Trackingverfahren beruhen allein auf gerichteten Kantenmessungen im 2D-Kamerabild mithilfe des CRONOS-Operators [Fürst, 2001] – der C-Implementierung des ursprünglichen KRONOS-Operators [Mysliwetz, 1990, Dickmanns, 1992] – und lieferten eine Klothoidenschätzung für die Fahrspurschnittstelle \mathcal{L} .

Hybrider Ansatz zur Erkennung im 2D-Kamerabild Feld- und Waldwege verfügen im Gegensatz zu asphaltierten Straßen über stärkere Krümmungen und einen unregelmäßigeren Krümmungsverlauf, für das in [Manz et al., 2010] ein zweistufiges Verfahren vorgestellt wurde. Ein Partikelfilter schätzt die Klothoide in der Vorausschau, und auf Basis der statischen Wegsegmente wird mithilfe eines Kalman-Filters die Fahrspur \mathcal{L} für die Fahrzeugführung geschätzt (analog zu Abschnitt 7.3.4.2). Details finden sich in [Manz, 2013, Kap. 4.3.1].

**Abbildung 7.8:**

(a) bis (c) unterschiedliche Verfahren zum Tracking von Straßen und Wegen, (d) zeigt die Fahrspurgenerierung anhand eines getrackten Objekts.

Erkennung in akkumulierten Rasterkarten Neben den oben behandelten horizontalen Krümmungen treten auf Feld- und Waldwegen auch stärkere vertikale Krümmungen auf, die das Tracking beeinflussen. Von Manz et al. [2011b] wurde daher – auch als Grundlage für die in Abschnitt 7.3.5.1 vorgestellte Kreuzungserkennung – ein Verfahren entwickelt, das die Fahrspuren nicht mit Messungen aus dem Kamerabild schätzt, sondern, wie in Abb. 7.8a gezeigt, Messungen in einem 3D-Geländemodell (siehe Abschnitt 7.3.2.2) verwendet. Das Trackingverfahren ist identisch zum hybriden Ansatz, Details finden sich in [Manz, 2013, Kap. 4.4].

Dieses Verfahren wurde von [Bayerl et al., 2015] für den Einsatz bei Dämmerung und Nacht weiterentwickelt und mit Messdaten weiterer Sensoren (hochempfindliche Farbkamera, Wärmebildkamera) erweitert. Abb. 7.8b zeigt eine beispielhafte Szene.

Kombiniertes Straßen- und Wege-Tracking Neuere Arbeiten [Forkel et al., 2021a, Forkel und Wuensche, 2022a] greifen die effizienten Messungen im Bild mithilfe des CRONOS-Operators wieder auf und kombinieren sie mit Höhenmessungen aus einem Umgebungsmodell [Forkel et al., 2021b, Forkel und Wuensche, 2022b] sowie Messungen aus einer segmentierten LiDAR-Punktwolke [Burger und Wuensche, 2018]. Zu Modellierung des Straßen- und Wegeverlaufs mit größerer Vorausschau wird ein komplexes Straßenmodell verwendet, welches rekursiv über der Zeit optimiert wird (siehe Abb. 7.8c). Hier kommt keine Fahrspurnschnittstelle \mathcal{L} zur direkten

Fahrzeugführung zum Einsatz, sondern das Straßenmodell ist Eingang für einen Planungsalgorithmus.

7.3.4.2 Fahrspurgenerierung anhand von Objekten

Die in Abschnitt 7.3.3 vorgestellte Fähigkeit, ein Objekt auf Basis von Sensordaten zu verfolgen, bedeutet noch nicht, dass der Roboter diesem Objekt selbstständig spurtreu folgen kann. Hierzu sind weitere Schritte erforderlich: Als Eingangsdaten werden die aktuelle Schätzung der Objektposition sowie Informationen über die Lokalisierung und Dynamik des eigenen Fahrzeugs verwendet. Hieraus wird der Pfad des Führungsfahrzeugs erzeugt, anschließend das Ego-Fahrzeug auf diesem Pfad lokalisiert und abschließend ein Klothoidensegment abgeleitet, um die Fahrspurschnittstelle \mathcal{L} zu befüllen. Abbildung 7.8d visualisiert diesen Prozess.

7.3.4.3 GNSS- & kartenbasierte Spurgenerierung

Im Rahmen dieser Arbeit wurde eine GNSS- & kartenbasierte Spurgenerierung als Rückfallebene entwickelt.

In Anhang B.3.3.2 wird erläutert, wie aus der Punktsequenz der globalen Route $\mathcal{P}_{\text{routeAnnotated}}$ eine geometrische Beschreibung in Form von kubischen Splines $\mathcal{S}_{\text{route}}$ erzeugt wird. Um daraus eine für die Fahrzeugführung geeignete Fahrspur \mathcal{L}_{map} abzuleiten, erfolgt zunächst die Lokalisierung des Ego-Fahrzeugs \mathbf{F}_{veh} auf dem Spline an der Stelle s . Aus der Relativlage ergeben sich Ablage d und Gierwinkel $\Delta\Psi$, während die Krümmung c_0 direkt aus dem Spline berechnet werden kann. Neben den geometrischen Daten werden für \mathcal{L}_{map} weitere Parameter benötigt: Die zulässige Maximalgeschwindigkeit wird aus den Punkten der annotierten Route $\mathcal{P}_{\text{routeAnnotated}}$ übernommen und ggf. auf Basis der Fahrzeugparameter und aktuellen Krümmung limitiert. Andere Werte wie die Breite werden mit konstanten Standardwerten befüllt.

Die Fahrspur \mathcal{L}_{map} kann zur unmittelbaren Fahrzeugführung in der Verhaltensweise Spurfolgen verwendet werden, bspw. als Rückfallebene bei Ausfall der visuellen Spurerkennung. Dabei ist zu beachten, dass dies nur unter der Voraussetzung präziser Karten und einer präzisen Lokalisierung nutzbar ist – meist also unter guten GNSS-Bedingungen wie offenem Himmel. Sind diese Bedingungen nicht erfüllt, sollte \mathcal{L}_{map} nur als Hypothese, z. B. zur Initialisierung des visuellen Wegetrackings, genutzt werden.

7.3.5 Kreuzungen

Einführung Damit ein autonomes Fahrzeug nicht nur „bis zum Ende der Welt“ einem Weg folgen kann, sondern in die Lage versetzt wird, auf einem Straßennetz zu navigieren, muss es abbiegen können. Eine Grundvoraussetzung hierfür ist die explizite Modellierung und Erkennung von Kreuzungen.

Tabelle 7.3:

Definition der Kreuzungs-Schnittstelle \mathcal{C} (vereinfacht). Alle Winkelangaben beziehen sich auf das in der Schnittstelle definierte Bezugssystem.

Bezeichnung	Variable	Bemerkung
Kreuzungsmittelpunkt x	$\mathbf{p}_{\text{crossing},[x]}$	
Kreuzungsmittelpunkt y	$\mathbf{p}_{\text{crossing},[y]}$	
UTM-Zone	–	
Bezugssystem	–	$\mathbf{F}_{\text{veh}}, \mathbf{Y}_{\text{LLH}}, \mathbf{F}_0, \mathbf{F}_{\text{DR}}$
Distanz entlang der Route	d_{crossing}	
Kreuzungs-ID	–	eindeutig entlang der Route
Anzahl der Äste	n_{branches}	
Winkel der Äste	$\Theta = \{\Psi_i\}$	$i = 0 \dots n_{\text{branches}}$
Array an Fahrspuren	$\mathcal{L} = \{\mathcal{L}_i\}$	$i = 0 \dots n_{\text{branches}}$, Definition \mathcal{L} in Tabelle 7.2
Array an Flags	$\mathbf{f} = \{f_i\}$	$i = 0 \dots n_{\text{branches}}$
Unsicherheit in $\mathbf{p}_{\text{crossing},[x]}$	$\sigma_{\mathbf{p}_{\text{crossing},[x]}}^2$	
Unsicherheit in $\mathbf{p}_{\text{crossing},[y]}$	$\sigma_{\mathbf{p}_{\text{crossing},[y]}}^2$	

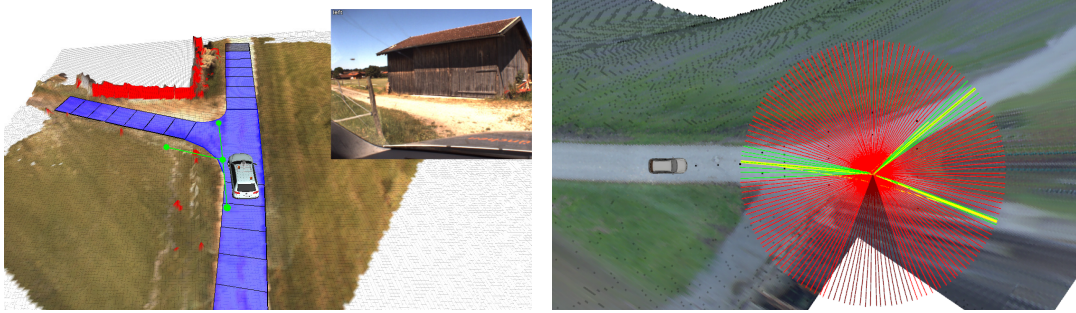
Im Folgenden wird zunächst die Kreuzung \mathcal{C} als Schnittstelle zwischen Wahrnehmung und Fahrzeugführung eingeführt. In Abschnitt 7.3.5.1 wird die auf Daten der Umfeldsensorik beruhende Kreuzungserkennung vorgestellt. Eine GNSS- & kartenbasierte Kreuzungsgenerierung dient zur Hypothesengenerierung für die Wahrnehmung und als Rückfallebene (siehe Abschnitt 7.3.5.2).

Schnittstelle Die Repräsentation einer Kreuzung \mathcal{C} besteht im Kern aus dem Kreuzungsmittelpunkt $\mathbf{p}_{\text{crossing}}$, der die Lage der Kreuzung global oder ggü. dem Ego-Fahrzeug angibt, sowie einer Menge an gerichteten Fahrspuren \mathcal{L}_i , den sog. Kreuzungsästen. Tabelle 7.3 zeigt die vollständige, im Rahmen dieser Arbeit definierte Kreuzungs-Schnittstelle \mathcal{C} .

In den folgenden Abschnitten werden unterschiedliche Wahrnehmungs- und Navigationsmodule präsentiert, die jeweils eine Kreuzung bereitstellen.

7.3.5.1 Erkennung und Tracking von Kreuzungen

Kreuzungen sind im Kamerabild aufgrund ihrer räumlichen Ausdehnung, die sich auch quer zur Fahrtrichtung erstreckt, nicht immer vollständig sichtbar. Die Nutzung einer aktiven Kameraplattform kann dies nur teilweise kompensieren. Ein weiterer Nachteil ist, dass die im Bild horizontal verlaufenden Kreuzungsäste aufgrund der perspektivischen Abbildung oft nur wenige Pixel hoch sind. Daher sind die für eine stabile Zustandsschätzung notwendigen Messungen nicht gleichzeitig für alle Kreuzungsäste und nicht in der notwendigen Qualität möglich.



(a) Erweitertes Kreuzungstracking mit dem verbesserten Formmodell, die Hypothese aus der Karte ist grün dargestellt. Bildquelle: [Bayerl, 2015, S. 41].
 (b) Ohne Kartenwissen erkannte Kreuzung aus gebündelten Strahlen. Bildquelle: [Forkel et al., 2023].

Abbildung 7.9: Erkennung und Tracking von Kreuzungen

Partikelfilter-basiertes Tracking in der akkumulierten Rasterkarte Zusammen mit der Fahrspurerkennung (vgl. Abschnitt 7.3.4.1) wurde die Kreuzungserkennung in akkumulierten Rasterkarten vorgestellt [Manz et al., 2011a] [Manz, 2013]. Im Partikelfilter-basierten Tracking kommt als Zustandsvektor ein – ggü. \mathcal{C} vereinfachtes – Kreuzungsmodell zum Einsatz. Auf Basis einer Kreuzungshypothese aus der Karte wird das Tracking initialisiert. Das Filter konvergiert auch bei Karten- oder Lokalisierungsfehlern im Bereich mehrerer Meter problemlos. Auf Basis der bestbewerteten Partikel wird die Schnittstelle \mathcal{C} befüllt.

Bayerl [2015] erweitert diese Arbeiten um die Erkennung auch bei Dämmerung und Nacht sowie eine verbesserte Berücksichtigung von Verdeckungen. Die Robustheit der Wahrnehmung konnte durch Modellierung einer expliziten Freifläche zwischen den Klothoidensegmenten, die analog zu den Ästen im Merkmalsraum ausgewertet wird und in die Partikelbewertung einfließt, weiter verbessert werden, wie in Abbildung 7.9a dargestellt.

Detektion ohne Vorwissen Um die Abhängigkeit einer kartenbasierten Kreuzungshypothese zu beseitigen, wurde in [Bayerl und Wuensche, 2014] ein Verfahren zur Kreuzungsdetektion ohne Vorwissen vorgestellt. [Forkel et al., 2023] hat diese Arbeiten wiederaufgegriffen und mit neuen Messverfahren verbessert (Abb. 7.9b). Diese detektierten Kreuzungen sind weniger für die direkte Verwendung in der Fahrzeugführung geeignet, sondern als Hypothese für die zuvor beschriebenen Schritte, zur Kartierung oder für Lokalisierungsverfahren [Mueller et al., 2011].

7.3.5.2 GNSS- & kartenbasierte Kreuzungsgenerierung

In Abschnitt 7.3.4.3 wird die Erzeugung der Fahrspur \mathcal{L}_{map} auf Basis der geplanten Route vorgestellt. Analog dazu wurde im Rahmen dieser Arbeit ein Verfahren entwickelt, dass die nächste Kreuzung $\mathcal{C}_{\text{next}}$ entlang der Route, wie in Anhang B.3.3.2 beschrieben, bereitstellt.

Diese Kreuzungshypothese wird von der visuellen Kreuzungserkennung in Abschnitt 7.3.5.1 benötigt, um das Partikelfilter mit der richtigen Anzahl an Kreuzungsästen und einer groben Position und Orientierung initialisieren zu können. In den Tentakeln mit Berücksichtigung von Kartenwissen (siehe Abschnitt 7.3.6) dient sie zur Erhöhung der Gewichtung der kartenbasierten Lokalisierung.

Eine weitere Anwendung von C_{next} ist die direkte Verwendung zum Abbiegen während einer Navigation auf einem Straßennetz. Ähnlich wie bei den Fahrspuren gilt hier die Anforderung an präzises Kartenmaterial und gute GNSS-Bedingungen.

7.3.6 Tentakel

Nicht in jeder Umgebung sind klar erkennbare Leitstrukturen wie Straßen, Feldwege oder Kreuzungen vorhanden, die ein autonomes Fahrzeug nutzen kann, um zum Ziel zu gelangen. Um auch eine Navigation in unstrukturiertem, engem Gelände zu ermöglichen, wurden am Institut die sog. *Tentakel* entwickelt, ein reaktives Navigationsverfahren, das unterschiedliche Fahrmöglichkeiten anhand der Sensordaten bewertet. Als weitere Herausforderung sind hier Waldumgebungen mit anspruchsvollen GNSS-Bedingungen zu nennen: hier kann es aufgrund von Dämpfung und Multipath-Effekten zu großen Störungen bis hin zu kompletten Ausfällen der Positionsbestimmung kommen.

Ursprünglicher Tentakel-Ansatz Diese Fragestellungen wurden am Institut erstmals durch den Tentakel-Ansatz zur Roboternavigation behandelt [von Hundelshausen et al., 2008]. Die dort verwendeten *Tentakel* bezeichnen integrierte Wahrnehmungs- und Bewegungsplanungsstrukturen („Integral Structures for Sensing and Motion“). Kreisbogensegmente definierter Breite werden anhand einer Ego-zentrierten Rasterkarte (siehe Abschnitt 7.3.2.1) nach unterschiedlichen Kriterien bewertet: Fahrbarkeit bzw. Hindernisfreiheit, Entfernung zur ersten Hinderniszelle, Flachheit und optional zielführende Eigenschaften.

Erweiterung um visuelle Merkmale Zur Unterscheidung „flacher Weg“ und „flacher Wiese“ helfen die Informationen der Rasterkarte nicht weiter, wohl aber visuelle Merkmale aus dem Farbbild [Manz et al., 2009]. Dieses Verfahren kam auf der C-ELROB 2009 in Oulu erfolgreich zum Einsatz und konnte sich hier besonders bei der Fahrzeugführung auf breiten Wegen bewähren [Luettel et al., 2009, Himmelsbach et al., 2009b]. Die akkumulierten Rasterkarten mit einer Farbinformationsebene (vgl. Abb. 7.5b) und daraus abgeleiteten Merkmalen kommen seit der M-ELROB 2012 auch für die Bewertung der Tentakel zum Einsatz.

Berücksichtigung von Kartenwissen Im ursprünglichen Ansatz der Tentakel wird der Sollpfad aus der globalen Karte immer mit gleichem Gewicht gewertet. Bei GNSS-Störungen und Kartenfehlern ergibt sich daraus oftmals eine Fahrt am Wegrand oder sogar abseits des Weges, um dem Sollpfad näher zu kommen. Verzichtet man

stattdessen wie in der C-ELROB 2007 ganz auf eine Gewichtung des Sollpfads, so ist keine zielführende Navigation mehr gewährleistet.

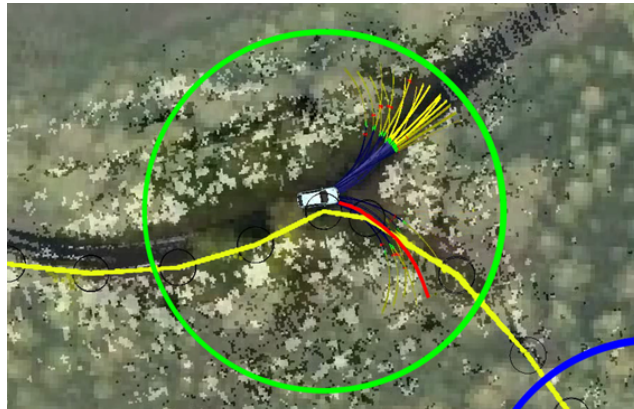
In [Luettel et al., 2009] wurde durch den Autor gezeigt, dass für eine Fahrt im offenen Gelände oder entlang von Wegen eine auf 20 % reduzierte Gewichtung des Sollpfads sehr gut geeignet ist. So werden die GNSS-beeinflussten Anteile von den reaktiven Anteilen überstimmt, und die Merkmale der Umgebung dominieren die Auswahl des Tentakels.

An Orten wie Kreuzungen oder Abzweigungen, an denen eine Richtungsentscheidung notwendig ist, ist diese Gewichtung jedoch schlecht geeignet. In deren „Anziehungsradius“ (empirisch auf Basis der erwarteten GNSS- und Karten-Abweichung zu $r_{\text{crossing}} = 20$ m gewählt) wird die Gewichtung so erhöht, dass sie gleichberechtigt zu den anderen Kriterien ist. Zusätzlich wird hier die Fahrzeuggeschwindigkeit reduziert, sodass auch stärker gekrümmte Tentakel zur Verfügung stehen und die Manövrierfähigkeit im Kreuzungsbereich verbessert wird. Ein Beispiel wird in Abbildung 7.10a gezeigt.

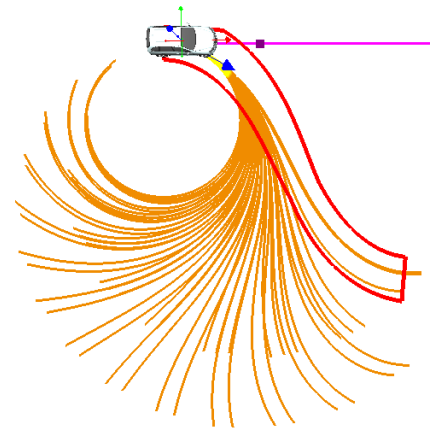
Verbesserte Tentakel-Geometrie Um die Manövrierfähigkeit des Roboters zu erhöhen, wurde das Design der Tentakel unter Berücksichtigung der Geometrie und Dynamik des Roboters erweitert. Diese Neuerungen sind unter Mitwirkung dieses Autors entstanden und wurden im Rahmen von [Himmelsbach et al., 2011] veröffentlicht.

Bei niedrigen Geschwindigkeiten kann die Dynamik eines vorderachsgelenkten Fahrzeugs durch das kinematische Einspurmodell (siehe Abschnitt 5.5.3.1) approximiert werden. Für die Geometrie der Tentakel wurden daher statt Kreisbogensegmenten Klothoidensegmente für die Vorwärts- und Rückwärtsfahrt gewählt, die jeweils an der Hinterachse des Roboters beginnen und deren Breite krümmungsabhängig ist. Abbildung 7.10b zeigt ein Beispiel. Die Bewertung der Tentakel erfolgt wie gehabt, allerdings wird das ausgewählte Tentakel nicht weiter als Vorsteuerung für den Lenkwinkel des Fahrzeugs verwendet, sondern als Sollspur für eine höherfrequente Regelung. Für den Fall, dass kein befahrbares Tentakel existiert und das Fahrzeug steht, werden zusätzlich „simulierte“ Tentakel untersucht, deren Startkrümmung nicht mit dem aktuellen Lenkwinkel korrespondierenden muss.

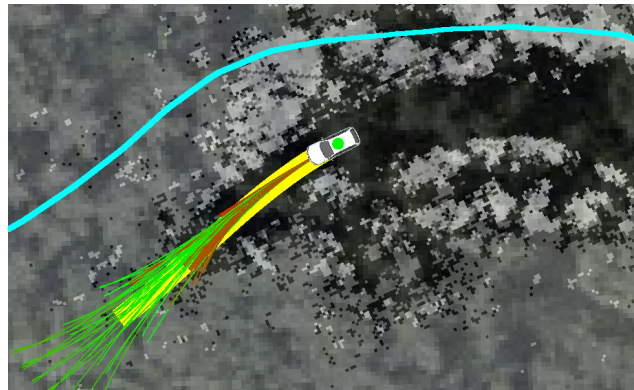
Bewertung extern vorgegebener Klothoidensegmente Für eine gezielte Bewertung von Klothoiden, die nicht Teil eines der vorberechneten Geschwindigkeitssets \mathcal{T}_i sein müssen, wurde eine externe Schnittstelle zu den *Fahrspuren* integriert. Aus der Parameterbeschreibung eines Klothoidensegments wird ein Tentakel erzeugt und dieses mit den bekannten Bewertungsverfahren untersucht. Neben der Bewertung wird auch die Distanz bis zum ersten Hindernis ermittelt und zurückgemeldet.



(a) Nutzung der um Kartenwissen erweiterten Tentakel auf der C-ELROB 2009. Rasterkarte der Hinderniswahrscheinlichkeit, überlagert auf Luftbild dargestellt. Aufgrund von GNSS-Störungen und Kartenfehlern ist der Sollpfad (gelb) nicht direkt fahrbar und wird im Normalfall nur mit geringem Gewicht genutzt. Obwohl die meisten Tentakel dem breiteren linken Weg folgen, wird in der dargestellten Situation ein Tentakel im rechten Weg ausgewählt (rot). Dies wird durch eine gezielte höhere Gewichtung des Sollpfads im Bereich von Kreuzungen (grün: aktuelle Kreuzung, blau: andere Kreuzungen) erreicht. Bildquelle: [Luettel et al., 2009]



(b) Realisierbare Tentakel (*orange*) für $v = 2 \text{ m/s}$ und $\lambda = -0,5 \text{ rad}$, mit visualisierter Stop-Distanz (*gelb*), aktuellem Lenkwinkel (*blauer Pfeil*) und willkürlich gewähltem Sollpfad (*magenta*). Für ein ausgewähltes Tentakel ist der Klassifikationsbereich dargestellt (*rot*). Bildquelle: [Himmelsbach et al., 2011].



(c) Tentakel bei langsamer Fahrt auf einem Waldweg im Mule-Szenario der M-ELROB 2010. Befahrbare Tentakel sind entsprechend ihrer Bewertung in der Rasterkarte (*grau-transparent*) eingefärbt, wobei *grün* einer besseren Bewertung als *rot* entspricht. Das für die Ausführung selektierte Tentakel ist als *gelbe* Klothoide dargestellt. Man beachte den für Waldszenarien typischen, durch GNSS-Störungen verursachten Lokalisierungsfehler zwischen zuvor aufgezeichneter Spur (*türkis*) und aktueller Positionsbestimmung.

Abbildung 7.10:

Nutzung der erweiterten Tentakel auf der C-ELROB 2009 und M-ELROB 2010.

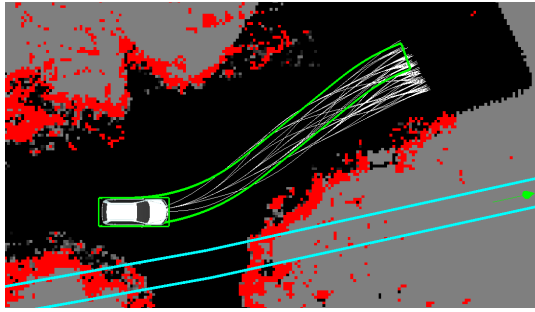


Abbildung 7.11:

Nutzung des *Gridplanning* für die Navigation durch ein Waldgebiet. Befahrbare Bereiche sind schwarz dargestellt, Hindernisse rot und verdecktes, noch nicht beobachtetes Gebiet grau. Aufgrund von GNSS- und/oder Kartenfehlern liegt die globale Route (cyan) in nicht befahrbarem Gebiet. Die 100 kostengünstigsten Trajektorien im Suchbaum des A*-Planers sind hier weiß dargestellt, die ausgewählte beste in Grün. Bildquelle: [Luettel et al., 2017]

7.3.7 Gridplanning

Beim *Gridplanning* handelt es sich um ein globales Navigationsverfahren mit größerem Planungshorizont, anstelle der reaktiven *Tentakel-Navigation*. Der Einsatzzweck ist weiterhin das unstrukturierte Umfeld und enge Gelände.

Bei der Pfadplanung entlang einer geplanten Sollspur $\mathcal{P}_{\text{routeAnnotated}}$ kommt ein auf Hybrid A* [Dolgov et al., 2010] basierender Algorithmus zum Einsatz. Dieser liest akkumulierte $2\frac{1}{2}$ D-Belegungskarten des Umgebungsmodells ein (vgl. Abschnitt 7.3.2.2), deren Zellen Informationen wie Hinderniswahrscheinlichkeit, Steigung, Vegetationswahrscheinlichkeit und Straßenwahrscheinlichkeit enthalten. Anhand dieser Merkmale sucht das *Gridplanning* nun nach der kostengünstigsten hindernisfreien Trajektorie, die das Fahrzeug in Richtung Ziel führt. In jedem Zeit- bzw. Planungsschritt werden dazu mehrere Klothoidensegmente definierter Länge zu einer Trajektorie aneinander gereiht, auch mit unterschiedlichem Geschwindigkeitsvorzeichen. Ein simuliertes Fahrzeugmodell stellt hierbei sicher, dass die kinematischen und dynamischen Beschränkungen des Ego-Fahrzeugs (maximaler Lenkeinschlag sowie maximale Quer-/Längsbeschleunigung) beachtet werden.

Details finden sich in den Veröffentlichungen des Kollegen Fassbender et al. [2014, 2016].

7.4 Koordination der Einzelverhalten durch hierarchischen Zustandsautomat

Zur Koordinaten der einzelnen Verhaltensweisen kommt ein Hierarchischer Zustandsautomat, engl. *Hierarchical State Machine* (HSM) zum Einsatz, dessen Funktionalität maßgeblich im Rahmen dieser Arbeit entwickelt wurde.

HSMs sind in der Robotik und im Bereich des autonomen Fahrens ein oft verwendetes Mittel, z. B. bei den Teilnehmern der DARPA Urban Challenge [Kammel et al., 2008, Urmsen et al., 2008, Montemerlo et al., 2008], bei der Bertha-Benz-Fahrt [Ziegler

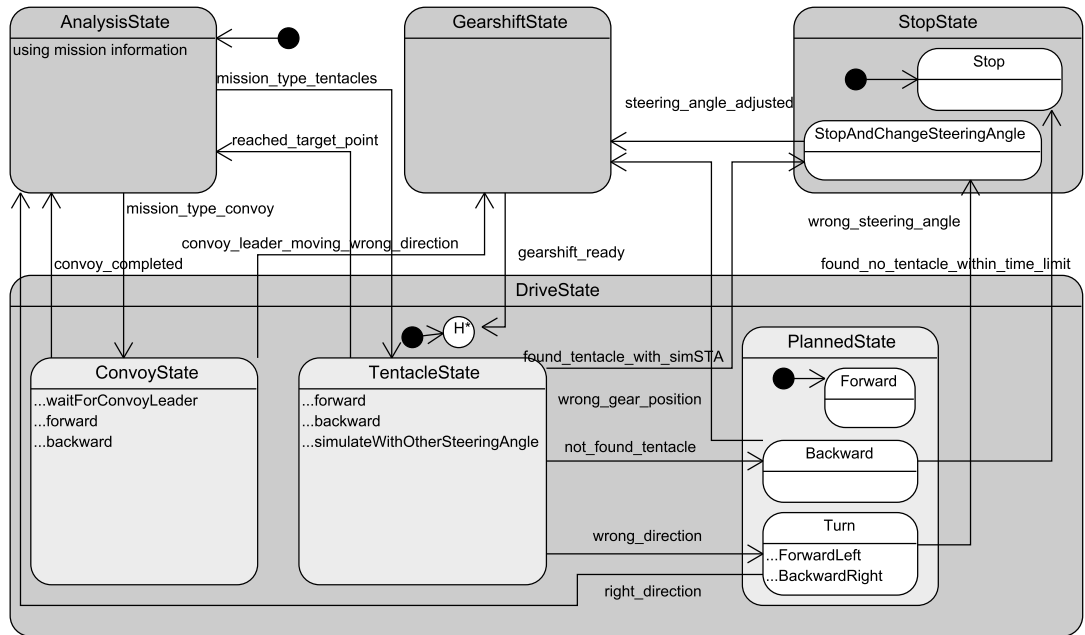


Abbildung 7.12:

Hierarchischer Zustandsautomat: reduzierte Darstellung aus dem Mule-Szenario der M-ELROB 2010. Bildquelle: [Luettel et al., 2011a].

et al., 2014] oder der Grand Cooperative Driving Challenge 2016 [Tas et al., 2018, Aramrattana et al., 2018].

In einem Zustandsautomaten bildet jeder *Zustand* ein generelles Verhaltensmuster wie *Fahren* oder *Stehen* ab. In abgeleiteten *Unterzuständen* der HSM werden diese Zustände weiter spezialisiert, gemeinsamer Programmcode verbleibt dabei im übergeordneten Zustand. Die Zustandsübergänge werden über *Transitionen* beschrieben, die bei eintretenden Bedingungen einen Wechsel des Zustands veranlassen. In jedem Zyklus werden die Bedingungen der vom aktiven Unterzustand und allen übergeordneten Zuständen ausgehenden Transitionen geprüft. Die Komplexität der HSM wächst exponentiell durch Hinzufügen neuer Zustände und den damit einhergehenden hohen kombinatorischen Möglichkeiten der möglichen und zu definierenden Zustandsübergänge.

Im Gegensatz zu „gelernten“ Verhalten mit neuronalen Netzen handelt es sich bei HSMs um aus menschlicher Sicht vorhersagbare Systeme.

Abbildung 6.5 aus dem vorherigen Kapitel zeigt die Verknüpfung der HSM mit dem Reglerframework sowie den Datenfluss. Die HSM nutzt als Eingangsdaten alle von den aktiven Wahrnehmungs- und Navigationsmodulen bereitgestellten Informationen zu Fahrspuren, Kreuzungen und dynamischen Objekten. Zusätzliche Eingänge liefern u. a. die aktuelle *Aufgabe*, die geplante globale Route $\mathcal{P}_{\text{routeAnnotated}}$ und die aktuelle Lokalisierung des Ego-Fahrzeugs auf der Route $\mathbf{p}_{\text{egoOnTrack}}$.

Beispiel Abbildung 7.12 zeigt eine auf die Konvoi- und Mule-Szenarien der M-ELROB 2010 reduzierte Version der für die Untersuchungen entwickelten HSM, an der im Folgenden beispielhaft die Funktionalität beschrieben wird.

Beim Start wird zunächst im *AnalysisState* die aktive *Aufgabe* geprüft, damit sinkt die Zahl der erreichbaren Unterzustände. Für die Aufgabe *Konvoi* ist dies nur der *ConvoyState* mit seinen Unterzuständen. Solange noch kein Führungsfahrzeug im Sichtbereich vor dem Ego-Fahrzeug ist, bleibt der *ConvoyWaitForLeaderState* aktiv, danach erfolgt ein Zustandsübergang nach *ConvoyForwardState*, und die Fahrt beginnt. Jeder Zustand nutzt unterschiedliche Fahrzeugfähigkeiten, die sog. Manöver aus Abschnitt 6.4, in diesem Fall das Spurfolgemanöver.

Falls das Führungsfahrzeug zurücksetzt, um ein Hindernis zu umfahren, muss auch das Folgefahrzeug zurücksetzen, und die Bedingung für die Transition in den *GearShiftState* wird wahr. Dieser aktiviert ein Manöver für die Ansteuerung der Gangschaltung und überwacht den Fortschritt. Nach Erreichen des Zielgangs erfolgt eine erneute Transition in den *DriveState*. Da der *GearShiftState* oft benötigt und daher über viele Transitionen aus den Unterzuständen erreichbar ist, wäre eine gleich große Menge an Rücktransitionen erforderlich. Zur Vereinfachung kommt nun die *StateHistory H** zum Einsatz, in der der *DriveState* den beim Verlassen aktiven Unterzustand speichert. Nach der Rückkehr in den *ConvoyState* wird der *ConvoyBackwardState* aktiviert.

Das auf Basis des Programmcodes erzeugte Zustandsübergangsdiagramm in Abbildung 7.13 gibt einen vollständigen Überblick über die Zustände.

Transitionen Viele Transitionen prüfen dieselbe Menge an Bedingungen. Zur Vereinfachung und Erhöhung der Wartbarkeit werden in den sog. *StateConditions* solche Untermengen an Bedingungen zusammengefasst, die häufig gemeinsam geprüft werden. Ein Beispiel ist der Spurwechsel: Hier muss geprüft werden, dass sowohl die Nachbarspur vom Tracking bereitgestellt wird, keine anderen dynamischen Objekte auf der Nachbarspur durch den Spurwechsel behindert werden, und zusätzlich die Strecke resp. Zeit vor der nächsten Kurve oder Einmündung für einen Spurwechsel ausreicht. Für letzteres werden die in Abschnitt 6.4 erwähnten Verfügbarkeitsabfragen der Manöver genutzt.

Zustände Jeder Zustand nutzt unterschiedliche Fahrzeugfähigkeiten, die sog. Manöver aus Abschnitt 6.4, bspw. das Spurfolgemanöver oder das Gangschalten. Diese Manöver werden innerhalb der einzelnen Zustände parametrisiert (Geschwindigkeit, IDs der für die Querführung relevanten Fahrspur sowie des für die Längsführung mittels ACC zu beachtenden Objekts), gestartet und ihr Fortschritt zyklisch überwacht.

Die Aufgabe *Navigation auf einem Straßennetz* verwendet den hier nicht dargestellten *LaneState* und *TurnOffState*. Hier wird innerhalb der Zustände bei Vorliegen mehrerer geeigneter Fahrspuren bzw. Kreuzungen entschieden, welche priorisiert genutzt wird. Der Fokus liegt hier auf den auf Basis von Sensordaten erkannten Leitstrukturen,

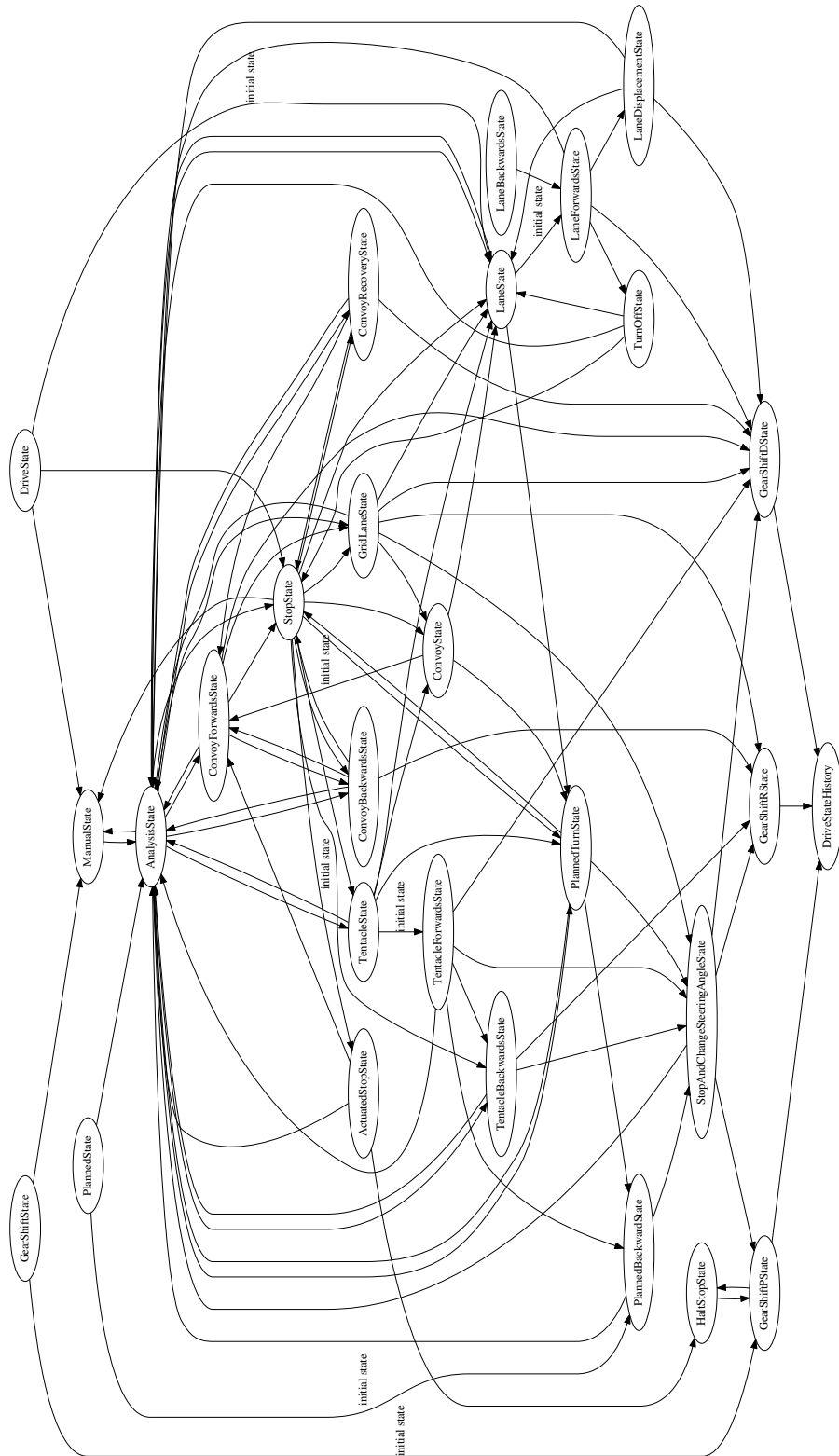
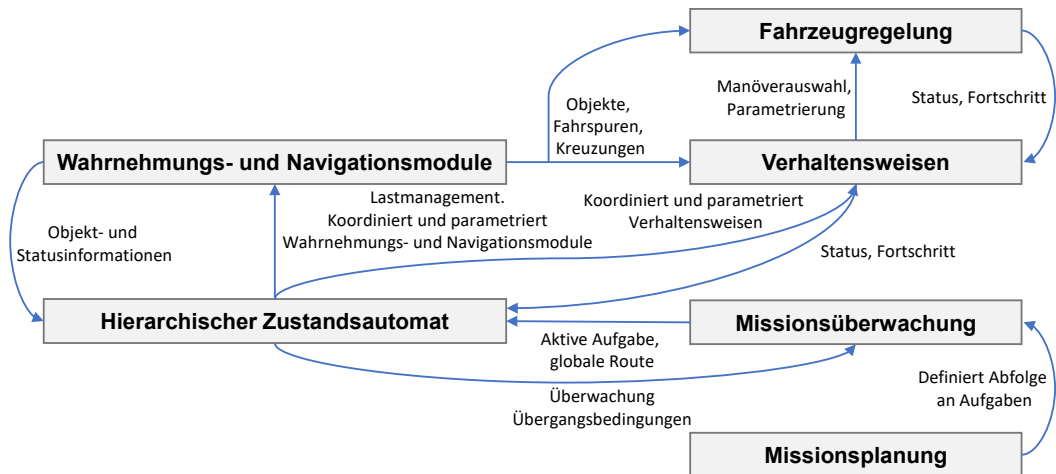


Abbildung 7.13: Zustandsübergangsdiagramm des hierarchischen Zustandsautomaten (automatisch erzeugt). Die Transitionen können aufgrund der Komplexität nicht automatisch beschriftet werden.

**Abbildung 7.14:**

Koordination der Verhaltensweisen, Wahrnehmungs- und Navigationsmodule mithilfe des Hierarchischen Zustandsautomaten und der Missionsüberwachung.

die GNSS- und kartenbasierten Fahrspuren bzw. Kreuzungen dienen lediglich als Rückfallebene.

In den Zuständen ist nicht nur die Funktionalität zur Verwaltung der aktiven Manöver hinterlegt, sondern auch Nebenaufgaben. So erfolgt aus der HSM heraus eine Parametrierung der Aufmerksamkeitssteuerung von Unterholzner [2016], die die optimale Blickrichtung für die aktive Kameraplattform MarVEye-8 bestimmt. Um diese an die jeweilige Aufgabe anzupassen, wird im Konvoi die Blickrichtung permanent der Position des Führungsfahrzeugs nachgeführt, während bei der Navigation auf dem Straßennetz zwischen einer Fahrspur-zentrierten Blickrichtung und einem zyklischen Blickrichtungswechsel an Kreuzungen unterschieden wird. In der hindernisvermeidenden Navigation erfolgt dagegen eine permanente Schwenkbewegung, um eine größtmögliche Fläche der Rasterkarte einzufärben. Ebenso erfolgt aus der HSM heraus eine Aktivierung und Deaktivierung einzelner, für die aktuelle Aufgabe nicht notwendiger Wahrnehmungsmodule, um die Systemressourcen zu schonen.

Rückmeldung an die Missionsüberwachung Ebenfalls innerhalb der HSM erfolgt die Prüfung von Übergangsbedingungen (engl. conditions) c^* , die für die überlagerte Missionsplanung von Bedeutung sind: *Zielkoordinate erreicht*, *Timeout*, *Langer Stillstand* oder *Blockade*. Die Erfüllung wird als sog. MissionEvent an die Missionsüberwachung kommuniziert, Details hierzu finden sich in Abschnitt 7.5.2.1.

Die Koordination zwischen HSM, Missionsüberwachung und den Wahrnehmungs- und Verhaltensweisen ist in Abb. 7.14 dargestellt.

7.5 Missionen

Wie bereits zu Beginn von Kapitel 7 eingeführt, handelt es sich bei einer Mission um eine Sequenz von *Aufgaben* für das autonome Fahrzeug. Die eben vorgestellten *Wahrnehmungs- und Navigationsmodule* dienen zur Umsetzung von *Verhaltensweisen*, welche wiederum für die Erfüllung einer *Aufgabe* dienen.

Das Konzept der Missionen beruht auf eigenen Arbeiten, die teilweise als Konferenzbeitrag auf der IEEE Intelligent Transportation Systems Conference präsentiert wurden [Luettel et al., 2011a].

Die Missionsplanung definiert für jede Aufgabe folgende Bestandteile:

Aufgabentyp Jede Aufgabe wird primär durch ihren Typ charakterisiert, welcher die Nutzung aufgabenabhängiger Wahrnehmungs- und Fahrzeugfähigkeiten definiert. Mögliche Aufgabentypen sind hier beispielsweise die Navigation auf einem Straßennetz oder das Verfolgen eines anderen Fahrzeugs im Konvoi.

Missionsziele Missionsziele können durch unterschiedliche Kriterien definiert werden, die den Abschluss einer Aufgabe anzeigen. Es handelt sich beispielsweise um das Erreichen einer globalen Zielkoordinate, um das Überschreiten eines Zeitlimits bzw. einer Wartezeit, oder um eine Signalisierung von extern.

Parametrierung der Wahrnehmungsfähigkeiten Über die Missionsplanung können für einige Wahrnehmungsmodule die essentiellen Parameter vorgegeben werden, die sich zwischen den Aufgaben ändern können. Ein Beispiel ist die Definition des Führungsfahrzeuges für das modellbasierte Fahrzeugtracking sowie ein eventuell verwendeter artifizieller Marker.

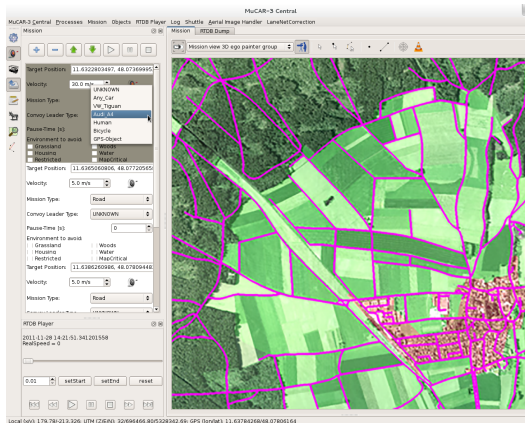
Parametrierung der Fahrzeugfähigkeiten Grundlegende Parameter für die Fahrzeugfähigkeiten werden ebenfalls an dieser Stelle festgelegt. Für nahezu alle Aufgaben von Bedeutung ist die Maximalgeschwindigkeit. Zusätzlich wird für Szenarien mit dynamischen Objekten auf der eigenen Fahrspur ein Wunschabstand für das ACC benötigt.

Parametrierung der Routenplanung Für besondere Anwendungen kann es wünschenswert sein, bspw. Waldflächen oder bebauten Gebiete zu vermeiden. Diese auszuschließenden Flächen-Layer können in der Missionsplanung gezielt markiert werden. Die globale Routenplanung zur Zielkoordinate erfolgt auf Basis der Straßen- und Wegenetze aus Anhang B und vermeidet dabei die markierten Bereiche.

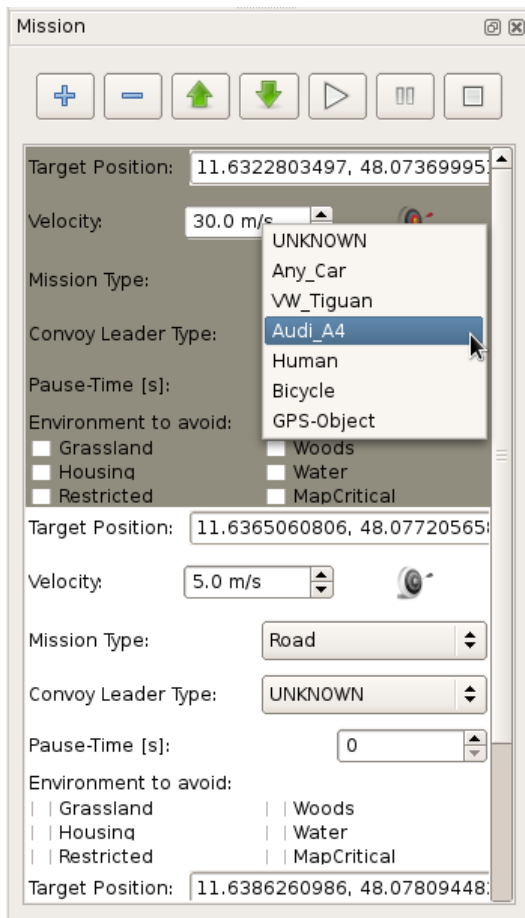
7.5.1 Erstellung von Missionen

Für die Erstellung von Missionen, also einer Verkettung von Aufgaben, wurden zwei Varianten entwickelt: einerseits ist die GUI-gestützte (Grafische Benutzerschnittstelle,

7 Missionen, Aufgaben und Verhaltensweisen



(a) MuCAR-Central mit visualisiertem Wegenetz (magenta) und aktiver Mission.



(b) GUI-basierte Erstellung der Mission: Einzelne Aufgaben können einfach hinzugefügt oder umsortiert werden (+−), die Zielkoordinaten werden per Maus im Wegenetz aus (a) ausgewählt. Alle Typ-Auswahlen wie z. B. für das Führungsfahrzeug sind über DropDown-Felder, Einschränkungen für die zulässige Umgebung über Checkboxes zugänglich.

```
#####
# Global configuration
#####
```

```
[area]
# Defines the road networks global reference system,
# either GPS or UTM
referenceSystem = GPS
# The coordinates that define the mission boundary, 3 at min.
coord0 = 11.610,48.080
coord1 = 11.610,48.060
coord2 = 11.650,48.060
coord3 = 11.650,48.080
```

```
[roadnet]
# Road network files, may start with road0 or road1
road0 = $HOME/maps/AP010L.shp
road1 = $HOME/maps/AP030L.shp
road2 = $HOME/maps/AP050L.shp
road3 = $HOME/maps/AP501L.shp
road4 = $HOME/maps/AQ065L.shp
road5 = $HOME/maps/AQ080L.shp
# Default road network velocity
globalVelocity = 10
```

```
[areas]
agriculture0 = $HOME/maps/EA010F.shp
water0 = $HOME/maps/BH015F.shp
water1 = $HOME/maps/BH030F.shp
woods0 = $HOME/maps/EC015F.shp
housing0 = $HOME/maps/flight_housing.shp
vision0 = $HOME/maps/flight_bad_vision.shp
mapcritical0 = $HOME/maps/flight_map_critical.shp
```

```
[images]
# Path to the image files
path = $HOME/aerial_images/
# Flag that defines that ALL images found in this folder
# will be used automatically and dynamically, depending
# from ego vehicle's position,
all = 1
# Flag that defines if the files listen below have to be
# loaded ALWAYS automatically,
always = 1
# Image names, may start with 0 or 1, i.e. image0 or image1
image0 = dop40c_4472_5326.bmp
```

```
#####
# The mission itself
#####
```

```
[mission]
# Mission subtask definition, may start with ...0 or ...1
```

```
[mission0]
# Target GPS position on road network
target0 = 11.6322803497,48.073699951
# Type of mission
type = CONVOY
# Type of Convoy Leader
convoyleader = PASSENGER_CAR
# Subtype of Convoy Leader
convoyleadersubtype = AUDI_A4
# Parameters
distToLeader = 20.0
# Max. velocity during task execution
maxVel = 30.0
# forbidden areas
```

```
[mission1]
target0 = 11.6365060806,48.07720565
type = ROAD
maxVel = 5.0
```

(c) Textbasierte Missionsbeschreibung: Der obere Block legt die zu ladenden Kartenausschnitte für Wegenetz, Flächenlayer und Orthofotos fest. Der untere Block definiert die Mission an sich, hier bestehend aus zwei Aufgaben.

Abbildung 7.15:

Alternative Darstellungen einer Mission, bestehend aus zwei Aufgaben: Zuerst „Konvoi“ (CONVOY) hinter einen Audi A4, anschließend „Navigation auf einem Straßennetz“ (ROAD).

engl. *Graphical User Interface*) Erstellung durch den Bediener möglich, andererseits die Verarbeitung von vorab erstellten Missionsbeschreibungsdateien in einem strukturierten Format.

Für die GUI-gestützte Missionserstellung ist die auch zur Visualisierung verwendete, am Institut gemeinschaftlich entwickelte Software MuCAR-Central erweitert worden. Zusätzlich zu den Orthofotos, die zur Darstellung und leichten Einordnung der aktuellen Situation dienen, können auch digitale Straßenkarten (vgl. Anhang B.3) eingelesen und visualisiert werden. Der Benutzer kann einzelne Aufgaben zur Mission hinzufügen, löschen oder die Reihenfolge anpassen. Die meisten Aufgabentypen benötigen eine Zielkoordinate, diese wird per Maus in der Karte (digitale Orthofotos und Wegenetz) gesetzt. Weitere Parameter für die Aufgabe, z. B. der Typ des Führungsfahrzeugs oder Bereichs-Beschränkungen für die Routenplanung, werden ebenfalls hier konfiguriert. In Abbildung 7.15a ist eine Darstellung von MuCAR-Central zu finden, Abbildung 7.15b zeigt einen detaillierteren Ausschnitt.

Die zweite Möglichkeit ist die Verarbeitung von vorab erstellten Missionsbeschreibungsdateien. Abbildung 7.15c zeigt eine exemplarische Mission mit zwei Aufgaben. Zusätzlich zur Parametrierung der einzelnen Aufgaben werden in der Missionsbeschreibungsddatei auch die zu verwendenden Kartenebenen definiert: einerseits das Straßen- und Wegenetz sowie Flächenlayer für die Berechnung der Routen, andererseits digitale Orthofotos für die Visualisierung. Die geladene Missionsdatei wird in die GUI übernommen, dort können die Aufgaben anschließend modifiziert werden.

7.5.2 Überwachung und Kontrolle der Mission

Ziel der Missionsüberwachung ist die Ausführung aller in der Mission enthaltenen Aufgaben in der gegebenen Reihenfolge. Zu Beginn jeder Aufgabe erfolgt anhand der Karte eine Routenplanung zum Zielpunkt, diese wird in Anhang B.3.2 näher beschrieben.

Zur Optimierung des Ressourceneinsatzes können einzelne, für die aktuelle Aufgabe nicht benötigte Komponenten in einen Standby-Modus geschickt werden. Ebenso erfolgt eine Signalisierung des Aufgabentyps an die Wahrnehmungskomponenten beim Wechsel der aktuellen Aufgabe. Beispielsweise wird für die *Tentakel-Navigation* das modellbasierte Fahrzeugtracking (siehe Abschnitt 7.3.3.3) nicht benötigt und ist auch nicht einsetzbar, da kein Führungsfahrzeugtyp bekannt ist.

Neben diesen einmaligen Aktionen zum Start der Aufgabe erfolgt eine zyklische Überwachung der Übergangsbedingungen.

7.5.2.1 Übergangsbedingungen für den Wechsel zwischen Aufgaben

Der Übergang von einer Aufgabe zur nächsten wird auf der Basis von Übergangsbedingungen (engl. *conditions*) ausgelöst. Die Prüfung dieser Übergangsbedingungen erfolgt hochfrequent in der HSM zusammen mit den Zustandsübergangsbedingungen

(siehe Abschnitt 7.4). In dieser Arbeit genutzte Übergangsbedingungen c^* sind vom Typ *Zielkoordinate erreicht*, *Timeout*, *Langer Stillstand* und *Blockade*, weitere Typen sind denkbar. Von der HSM werden sog. *MissionEvents* an die Missionsüberwachung kommuniziert, diese prüft zyklisch die eintreffenden *MissionEvents* und reagiert entsprechend, wie im Folgenden detaillierter ausgeführt wird:

Zielkoordinate erreicht Diese Bedingung kommt in den Aufgabentypen *Konvoi*, *Navigation auf einem Straßennetz* und *Hindernisvermeidende Navigation* zum Einsatz.

Als Bedingung wird das Erreichen einer globalen Zielkoordinate $\mathbf{p}_{\text{target}}$ überprüft. Da sowohl die eigene GNSS-gestützte Lokalisierung \mathbf{p}_{ego} als auch die Karte bzw. die daraus abgeleitete Zielkoordinate $\mathbf{p}_{\text{targetOnTrack}}$ fehlerbehaftet sind, wird hier jedoch kein exaktes Erreichen gefordert. Aufgrund der Ungenauigkeiten wird nicht die Fahrzeugposition \mathbf{p}_{ego} selbst, sondern ihr korrespondierender Punkt $\mathbf{p}_{\text{egoOnTrack}}$ auf der Sollspur gegenüber dem letzten Punkt $\mathbf{p}_{\text{targetOnTrack}}$ auf der Sollspur überprüft:

$$c_{\text{targetReached}} = \begin{cases} 1 & \|\mathbf{p}_{\text{targetOnTrack}} - \mathbf{p}_{\text{egoOnTrack}}\| < d_{\text{minOnTrack}} \\ & \wedge \|\mathbf{p}_{\text{targetOnTrack}} - \mathbf{p}_{\text{target}}\| < d_{\text{minTargetToTrack}} \\ 0 & \text{sonst} \end{cases} \quad (7.8)$$

Die tolerierte Abweichung in der 2D-Position wurde empirisch gewählt zu $d_{\text{minOnTrack}} = 1$ m, die zulässige Abweichung zwischen originalem Zielpunkt und Punkt auf dem Straßennetz zu $d_{\text{minTargetToTrack}} = 10$ m. Diese Übergangsbedingung löst als Aktion den Übergang zur nächsten Aufgabe aus der Mission aus.

Timeout Im Aufgabentyp *Pausieren* wird die gewünschte Wartezeit $\Delta t_{\text{desiredTimeout}}$ als Parameter in der Missionsplanung vorgegeben. Für diese Aufgabe ist die Verhaltensweise *Anhalten und Not-Halt* zuständig, sodass in der Übergangsbedingung die aktive Zeit im korrespondierenden Zustand überwacht wird:

$$c_{\text{timeoutReached}} = \begin{cases} 1 & \Delta t_{\text{StopState}} \geq \Delta t_{\text{desiredTimeout}} \\ 0 & \text{sonst} \end{cases} \quad (7.9)$$

Diese Übergangsbedingung löst als Aktion ebenfalls den Übergang zur nächsten Aufgabe aus der Mission aus.

Langer Stillstand Im Aufgabentyp *Mule Teach-In* kommt die Übergangsbedingung *Langer Stillstand* zum Einsatz. Bewegt sich das Führungsfahrzeug bzw. die Person über einige Sekunden nicht weiter, so wird in der HSM die Verhaltensweise *Anhalten und Not-Halt* aktiviert, um eine eigene Bewegung auch ohne dauerhaften Aktorikeinsatz zu verhindern. In dieser Übergangsbedingung wird die aktive Zeit im

korrespondierenden Zustand überwacht, wobei die minimale Stillstandszeit empirisch zu $\Delta t_{\text{stoppingLongTime}} = 20 \text{ s}$ bestimmt wurde.

$$c_{\text{stoppingLongTime}} = \begin{cases} 1 & \Delta t_{\text{StopState}} \geq \Delta t_{\text{stoppingLongTime}} \\ 0 & \text{sonst} \end{cases} . \quad (7.10)$$

Bei erfüllter Übergangsbedingung $c_{\text{stoppingLongTime}}$ wird zuerst der aufgezeichnete Weg aus dem *Mule Teach-In* als Karte abgespeichert. Danach erfolgt der Übergang auf die nächste Aufgabe *Mule Shuttle*.

Blockade Die Erkennung von Blockaden kommt in den Aufgabentypen *Navigation auf einem Straßennetz*, *Hindernisvermeidende Navigation* und *Mule Shuttle* zum Einsatz. Für die Übergangsbedingung werden die Erkennung einer blockierten eigenen Fahrspur oder die Tatsache, dass das Fahrzeug über längere Zeit dem Zielpunkt $p_{\text{targetOnTrack}}$ nicht mehr näher kommt, untersucht:

$$c_{\text{blocked}} = \begin{cases} 1 & \text{Fahrspur blockiert} \vee \text{kein Fortschritt} \\ 0 & \text{sonst} \end{cases} . \quad (7.11)$$

Ist die Übergangsbedingung c_{blocked} erfüllt, so wird das aktuelle Wegsegment in der globalen Karte mit extrem hohen Kosten belegt, anschließend wird die Routenplanung aus Anhang B.3.2 erneut ausgeführt. Diese liefert nun – wenn vorhanden – einen alternativen Sollpfad zum Zielpunkt. Der Aufgabentyp bleibt dabei aktiv, in der HSM werden die nötigen Entscheidungen für den Umgang mit dem geänderten Sollpfad getroffen.

Weitere denkbare Übergangsbedingungen Als Übergangsbedingungen sind auch Signale von extern denkbar, hier nur zwei Beispiele: Der Abschluss des *Mule Teach-In* könnte von der anlernenden Person über einen Schalter am Fahrzeug oder über Funk ausgelöst werden. Wenn die fahrzeugseitige Aufgabe *Pausieren* zu Aufklärungszwecken genutzt wird, sollte der Übergang auf die nächste Bedingung nicht nach festgelegter Zeit sondern durch ein Signal des Aufklärungsmoduls bzw. eines Operators erfolgen.

7.6 Zusammenfassung

Das im Rahmen dieser Arbeit entwickelte hierarchische Konzept der Missionen hat sich zur Verknüpfung unterschiedlicher *Aufgaben*, *Verhaltensweisen* sowie *Wahrnehmungs- und Navigationsmodule* bewährt. Für jede Aufgabe kann ein Navigationsverfahren der am besten geeigneten Rubrik ausgewählt werden (reaktive Navigation, globale Navigation oder geführte Navigation), sodass hier ein Vorwissen des Operators berücksichtigt werden kann.

Viele Wahrnehmungskomponenten kommen in verschiedener Zusammensetzung für unterschiedliche Verhaltensweisen zum Einsatz. Kernelement für das komplexe Zusammenspiel vieler unterschiedlicher Module, die noch dazu von verschiedenen Wissenschaftlern entwickelt wurden, sind sauber definierte Schnittstellen (hier für getrackte dynamische Objekte, Fahrspuren und Kreuzungen), die eine einfache Vergleichbarkeit und Austauschbarkeit von Komponenten ermöglichen. Damit lässt sich Forschungsfrage 6 „Wie können diese modular zu Verhaltensweisen und Aufgabentypen kombiniert werden, um Missionen mit unterschiedlichen Teilaufgaben abzubilden?“ beantworten.

Der Nachweis der Leistungsfähigkeit konnte anhand der ELROB-Szenarien erfolgen, die am Anfang dieses Kapitels die Modularisierung motiviert haben: Das vorgestellte System kam ab 2010 in den Wettbewerbsszenarien der M-ELROB und der euRathlon sowie für diverse Projektpräsentationen mit MuCAR-3, MuCAR-4 und TULF zum Einsatz. In Anhang C finden sich detaillierte Szenariobeschreibungen und Auswertungen zu zwei realen Szenarien.

Die HSM an sich ist im Nachgang jedoch zwiespältig zu betrachten. Einerseits konnten die geforderten Szenarien mit den dazu notwendigen Verhaltensweisen erfolgreich modelliert werden und über Jahre hinweg in unterschiedlichen Demonstrationen und Wettbewerbsteilnahmen gezeigt werden. Andererseits explodiert die Komplexität der HSM bei wachsender Zustandsanzahl, z. B. wenn weitere Verhaltensweisen wie Parkplatznavigation oder innerstädtischen Verkehr integriert werden sollen, durch die dazu notwendigen zusätzlichen Transitionen, welches die Fehlersuche im Realbetrieb aufwändig macht. Im Vergleich zu einem „flachen“ Zustandsautomaten wird die Explosion durch den Einsatz einer HSM leicht gedämpft, aber nicht verhindert. Die in dieser Arbeit gewählte Modellierung der HSM ist zudem kaum um Verkehrsregeln und ähnliches zu erweitern – beim Entwurf standen die eher militärisch geprägten Szenarien im Vordergrund.

Weitere Entwicklung bei TAS Parallel zu den Untersuchungen mit dem in dieser Arbeit aufgezeigten Gesamtkonzept entstand im Rahmen eines Kundenprojekts für die Firma StreetScooter GmbH (StS) eine Gesamtarchitektur für ein suburbanes Verkehrsszenario mit Paket-Zustellaufgaben [Heinrich et al., 2018]. Das dazu notwendige Verhalten wurden vom Kollegen Ebert ebenfalls als HSM modelliert, Verkehrsregeln wurden als zusätzliche Ebene der Rasterkarten erzeugt. Für die Bahnplanung kam ausschließlich eine in Fassbender et al. [2016] erweiterte Version der *Gridplanning-Navigation* zum Einsatz, die auch die Verkehrsregel-Rasterkarte und dynamische Objekte in die Bewertung einbezog. Im Vergleich zur alten Architektur bestand die neu entwickelte HSM aus wesentlich weniger Zuständen und Transitionen, da viele Aspekte unterlagert zwischen dem Bahnplanungs- und dem Bahnregelungsmodul [Heinrich et al., 2017][Heinrich et al., 2016] ausgehandelt wurden, zeigte jedoch bei den im Projektverlauf hinzukommenden Szenarien ebenfalls einen Trend zur hohen Komplexität.

Für die Mule-Szenarien der M-ELROB 2016 und 2018 wurde basierend auf diesen Erkenntnisse eine einfache HSM entworfen und erfolgreich genutzt [Ebert et al.,

2017]. Diese HSM modellierte allerdings eher die in dieser Arbeit als Missionsüberwachung bezeichnete Funktionalität (*Mule Teach-in* oder *Mule Shuttle?*), ohne flexibel weitere Aufgabentypen zu unterstützen.

Weitere Untersuchungen sollten aufgrund des Komplexitätsproblems der HSMs Alternativen betrachten, wie bspw. die *Behavior Trees* oder *Verhaltensbäume*, die ebenfalls deterministisch sind. Viel Potential ist bei der lernbasierten Verhaltensgenerierung zu erwarten, die allerdings nicht mehr deterministisch ist. Hier kommt die Disziplin der XAI (Erklärbare Künstliche Intelligenz, engl. *Explainable artificial intelligence*) ins Spiel, um die Entscheidungen des System jederzeit nachvollziehen zu können.

8 Fazit

Im Rahmen dieser Arbeit wurde untersucht, wie autonome Versuchsfahrzeuge für den universitären Forschungsbetrieb gestaltet werden können. Dazu wurden mehrere Versuchsfahrzeuge unterschiedlicher Größenklassen vom PKW bis zum LKW aufgebaut und in Betrieb genommen, teilweise Uni-intern, teilweise mit Partnern aus der Industrie.

Über die Hardware hinaus wurde untersucht, wie die Systemarchitektur für ein autonomes Straßen- und Geländefahrzeug gestaltet werden sollte, damit es in definierten militärisch geprägten Szenarien des Gütertransports eine komplexe Mission mit mehreren Teilaufgaben erledigen kann. Dieses Zusammenspiel konnte über viele Jahre hinweg experimentell nachgewiesen werden.

Zusammenfassend werden im folgenden Abschnitt noch einmal die in Abschnitt 1.4 aufgeworfenen Forschungsfragen betrachtet.

8.1 Beantwortung der Forschungsfragen

Forschungsfrage 1: Wie kann ein Versuchsfahrzeug mit seiner Sensorik-, Aktorik- und Rechner-Ausstattung gestaltet werden? Im Rahmen dieser Arbeit wurden mehrere Versuchsfahrzeuge auf der Basis von zivilen Serien-PKW aufgebaut und der Aufbau von Versuchsfahrzeugen auf der Basis militärischer LKW begleitet, um diese für die universitäre Forschung im autonomen Fahren nutzbar zu machen. Kernelemente waren hierbei die Ausstattung mit Sensorik, Aktorik und Rechentechnik, aber auch die Einbindung in das Basisfahrzeug für Schnittstellen und Energieversorgung.

Wie in Abschnitt 3.9 aufgezeigt, war dieses auf unterschiedlichen Wegen erfolgreich: der Umbau in Eigenregie ist zeitaufwändiger, aber nachhaltiger für ein tiefes Systemverständnis und langfristige Wart- und Nutzbarkeit. Beim Umbau mit Partnern ist die Wartbarkeit schwieriger, und während des Aufbaus sind oft Iterationen notwendig, um die Nutzbarkeit für die Forschung sicherzustellen.

Insgesamt lässt sich feststellen, dass das KISS-Prinzip (Halte es einfach, Du Dummkopf!, engl. *Keep it simple, stupid!*) auch beim Aufbau von Versuchsfahrzeugen seine Berechtigung hat: ein nachgerüstetes Fahrzeug wie MuCAR-3 mit separater Aktorik und an der Außenhaut nachgerüsteten Sensoren lässt sich im Forschungsumfeld leichter warten und erweitern als ein komplexes monolithisches System wie ein moderner PKW mit ab Werk integrierter Sensorik und Aktorik.

Gleiches gilt für Schnittstellen zwischen einem High-Level-System für die Autonomiefunktionen und einem Low-Level-System für die fahrzeugspezifische Regelung und Ansteuerung: Auf der unteren Ebene sollte Komplexität gekapselt werden, beispielsweise Initialisierungsprozeduren für die Aktorik, und das Auslesen der herstellerspezifischen Bus-Kommunikation übernommen werden. Die Schnittstellen zwischen

High- und Low-Level-System können so stark vereinfacht und, vor allem über unterschiedliche Fahrzeugtypen hinweg, vereinheitlicht und langfriststabil gestaltet werden. Dies ist wichtig, damit nicht jeder Forscher ein tiefes Systemverständnis für jedes Versuchsfahrzeug benötigt.

Insgesamt kann diese Frage also als beantwortet gelten.

Forschungsfrage 2: Wie sieht ein für die Entwicklungsphase geeignetes Sicherheitskonzept aus? Diese Frage lässt sich kurz beantworten: im universitären Umfeld mit Prototypenständen jeglicher Software ist ein Sicherheitsfahrer zusammen mit einem Operator das zentrale Element des Sicherheitskonzepts, da diese sowohl Autonomie als auch Umfeld überwachen und jederzeit eingreifen können. Mehr Details finden sich in Abschnitt 3.9.

Um hier zusätzliche technische Sicherheitsschichten einzuziehen, wird derzeit von Partnern im Projekt MORE die Integration eines separaten Nahfeldüberwachungssystems untersucht, das den zu befahrenden Bereich vor dem Fahrzeug mit separater Sensorik überwacht und eine Notbremsung auslösen kann. Dieses wurde in Abschnitt 3.8.5 kurz vorgestellt.

Forschungsfrage 3: Welche Hard- und Softwareumgebung ermöglicht einen vereinfachten Entwicklungsprozess für mehrere Wissenschaftler im Fahrzeug und im Labor? Als Hardwareumgebung haben sich x86-64-Systeme etabliert, die vom leichten Notebook über Desktop-PCs bis zu den Fahrzeugrechnern sehr gut skalieren. Algorithmen, die auf maschinellen Lernverfahren basieren, benötigen eine leistungsfähige GPU (Grafikprozessor, engl. *Graphics processing unit*), die zusätzliche Energie benötigt. Die Energieeffizienz wäre mit spezialisierter Hardware jedoch höher als mit den universell einsetzbaren x86-64-Systemen.

Im Robotik-Umfeld, zu dem auch das autonome Fahren gehört, ist die Verwendung einer Middleware zur Kommunikation zwischen verschiedenen Softwaremodulen nahezu unerlässlich. Neben der Inter-Prozess-Kommunikation ist hier eine Funktionalität zur Datenaufzeichnung und -wiedergabe erforderlich. Sowohl die KogMo-RTDB (Echtzeitdatenbank für kognitive Automobile, engl. *KogniMobil Real Time Data Base*) als auch ROS (Robot Operating System) haben sich im täglichen Einsatz bewährt, sie haben unterschiedliche Stärken und Schwächen. Der Trend geht aufgrund der großen Community und der aktiven Weiterentwicklung klar in Richtung ROS und bietet mit ROS2 viel Potential. Insgesamt kann diese Forschungsfrage als beantwortet gelten.

Forschungsfrage 4: Kann die Eigenbewegungsschätzung für autonome Geländefahrzeuge mit kostengünstiger Sensorik erfolgen? Das Ergebnis der *modularen Eigenbewegungsschätzung*, die auf günstigerer Sensorik aufbaut, ist für den Hauptanwendungsfall geeignet: die Wahrnehmungs- und Navigationsmodule des Gesamtsystems benötigen vor allem eine stabile, sprungfreie und möglichst

driftarme Dead-Reckoning-Pose, um Funktionalitäten wie Objekttracking oder Fahrspurtracking zu realisieren. Für den Konvoi-Betrieb in den Studien TULF und vor allem StrAsRob konnte der Regelkreis unter der Verwendung der Read-Reckoning-Pose geschlossen werden, sodass alle Demonstrations- und Abnahmefahrten mit dem hier vorgestellten System erfolgen konnten.

Diese Forschungsfrage kann aber insgesamt aus Sicht des Autor nur teilweise als beantwortet gelten: Einerseits besteht bei den Schätzung der dynamischen Zustände noch Verbesserungsbedarf, und andererseits liegt in der Modellierung von Versuchsträgern mit nicht-stationären Sensorpositionen noch viel Potential.

Weitere Details hierzu werden in Abschnitt 5.11 erläutert.

Forschungsfrage 5: Über welche Wahrnehmungs- und Navigationsmodule muss ein autonomes Fahrzeug für diese Szenarien verfügen? Auch wenn in dieser Arbeit (über die Eigenbewegungsschätzung hinaus) keine Module zur Umgebungswahrnehmung entwickelt wurden, besteht doch aus Gesamtsystemperspektive die Fähigkeit zur Beurteilung. Für den Gütertransport zwischen zwei Camps kann einerseits ein Konvoi zum Einsatz kommen, hier wird vor allem ein äußerst robustes Objekttracking benötigt, welches mit einem präzisen Spurfolgen kombiniert wird. Hier haben sich Trackingverfahren, die heterogene Sensoren wie Kamera und LiDAR fusionieren, ggü. Verfahren für einen einzelnen Sensortyp als überlegen herausgestellt.

Komplexer wird es bei Verzicht auf ein Führungsfahrzeug: hier muss das autonome Fahrzeug in der Lage sein, sowohl auf markierter als auch unmarkierter befestigter Straße sowie auf Feld- und Waldwegen die Fahrspur zu erkennen und dieser zu folgen. Zusätzlich wird eine Lokalisierung in Verbindung mit einer Kreuzungserkennung benötigt, um gezielte Abbiegeentscheidungen der globalen Route entsprechend umsetzen zu können. Auch hier hat sich eine Sensordatenfusion bewährt.

In schwierigerem, engerem Gelände sind oftmals keine klaren Leitstrukturen wie Wege zu erkennen. Hier wird zusätzlich eine hindernisvermeidende Navigation benötigt, um anhand einer grob bekannten Route einen Weg durch die mit der lokalen Sensorik erkannte Umgebung zu finden. Hier haben sich Planungsverfahren mit größerer Vorausschau wie das Gridplanning ggü. den zu Beginn verwendeten Tentakeln als überlegen herausgestellt.

Eine große Herausforderung ist weiterhin eine Klassifikation der Hindernisse und des Untergrunds anhand der Materialien, um dieses in einem Umgebungsmodell zu integrieren – dies würde sowohl der Wegeerkennung als auch den Planungsverfahren helfen.

Forschungsfrage 6: Wie können diese modular zu Verhaltensweisen und Aufgabentypen kombiniert werden, um Missionen mit unterschiedlichen Teilaufgaben abzubilden? Das im Rahmen dieser Arbeit entwickelte hierarchische

Konzept der Missionen (Kapitel 7) hat sich zur Verknüpfung unterschiedlicher *Aufgaben*, *Verhaltensweisen* sowie *Wahrnehmungs- und Navigationsmodule* bewährt. Für jede Aufgabe kann ein Navigationsverfahren der am besten geeigneten Rubrik ausgewählt werden (reaktive Navigation, globale Navigation oder geführte Navigation), sodass hier ein Vorwissen des Operators berücksichtigt werden kann.

Viele Wahrnehmungskomponenten konnten in verschiedener Zusammensetzung für unterschiedliche Verhaltensweisen zum Einsatz kommen. Kernelement für das komplexe Zusammenspiel vieler unterschiedlicher Module, die noch dazu von verschiedenen Wissenschaftlern entwickelt wurden, sind sauber definierte Schnittstellen (hier für getrackte dynamische Objekte, Fahrspuren und Kreuzungen), die eine einfache Vergleichbarkeit und Austauschbarkeit von Komponenten ermöglichten.

Insgesamt kann diese Frage mit „ja“ beantwortet gelten, auch wenn das System der Missionen und des hierarchischen Zustandsautomaten (HSM) mittlerweile eine Nachfolgelösung gefunden hat. Die HSM an sich ist im Nachgang zwiespältig zu betrachten, wie in Abschnitt 7.6 erläutert.

Forschungsfrage 7: Kann sich dieses System in einem Wettbewerbsszenario gegenüber anderen Systemen und Teams behaupten? Das Gesamtsystem von *Team MuCAR* konnte wiederholt in den Wettbewerbsszenarien der ELROB (European Land Robot Trial) von 2007 bis 2022 eine sehr gute Performance zeigen. Ein Teil ist den Versuchsfahrzeugen zuzuschreiben: umgerüstete handelsübliche PKWs sind ein guter Mittelweg zwischen kleiner Eigenbau-Plattform und großem LKW, was Fahrzeugrobustheit, Geschwindigkeit und Wendigkeit auch in Engstellen angeht. Sie bieten gute Voraussetzungen für die Ausrüstung mit Sensorik und Rechentechnik.

Der entscheidendere Teil liegt jedoch in der Software und deren Zusammenspiel, für das sich das im Rahmen dieser Arbeit entwickelte hierarchische Konzept über mehrere Jahre hinweg bewährt hat. Neue Algorithmen in den Wahrnehmungs- und Navigationsmodulen konnten dank langfristiger stabiler Schnittstellen problemlos in diese Systemarchitektur integriert werden.

Das Konvoi-Szenario konnte in allen Jahren mit Platz 1 oder 2 abgeschlossen werden. Hier spielt vor allem ein stabiles Fahrzeugtracking und ein spurtreues Folgen eine Rolle und weniger die Missionsplanung und das Verhalten. In den Mule-Szenarien konnte mit dieser Architektur ebenfalls sehr erfolgreich präsentiert werden, wie in Anhang C gezeigt. Hier liegt der Schwerpunkt einerseits auf der Modellierung des Verhaltens, welches mit der in dieser Arbeit gezeigten Architektur hinreichend möglich war, andererseits auf einer nicht immer einfachen Wahrnehmung der Umgebung – hier gibt es teilweise Hindernisse, die nur zu erkennen sind, wenn der Roboter direkt davor steht.

Aus Sicht des Autors bleibt im Bereich der autonomen Fahrzeuge und insbesondere im Anwendungsbereich abseits der Straße weiterhin Forschungsbedarf – packen wir's an!

A Posen und Homogene Koordinaten

A.1 6-DOF-Pose

Die 6-DOF-Pose ${}^A\mathbf{P}_B$ beschreibt die Lage eines rechtshändigen kartesischen Koordinatensystems \mathbf{F}_B gegenüber dem Koordinatensystem \mathbf{F}_A . ${}^A\mathbf{P}_B$ wird durch 6 Parameter beschrieben: Eine 3D-Translation entlang der Achsen x , y und z , sowie die drei Rotationen mit den Euler-Winkeln. In dieser Arbeit kommt die in der Fahrzeugtechnik übliche Rotationsreihenfolge zum Einsatz. Sie ist in DIN ISO 8855 [DIN Deutsches Institut für Normung e.V., 2013] bzw. ihrer Vorgängernorm DIN 70000 [DIN Deutsches Institut für Normung e.V., 1994] für den Automobilbau definiert und wird am Institut seit vielen Jahren verwendet:

1. Rotation mit dem Gierwinkel Ψ um die z -Achse,
2. Rotation mit dem Nickwinkel Θ um die neue y -Achse,
3. Rotation mit dem Rollwinkel Φ um die neueste x -Achse.

$${}^A\mathbf{P}_B = \begin{pmatrix} A_x \\ A_y \\ A_z \\ A_\Phi \\ A_\Theta \\ A_\Psi \end{pmatrix}. \quad (\text{A.1})$$

Eine Visualisierung ist in Abb. A.1a zu finden.

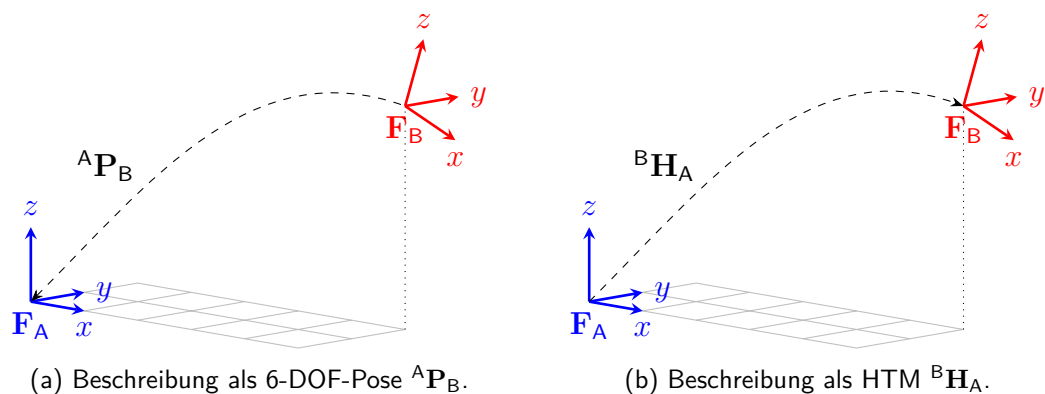


Abbildung A.1:

Die 6-DOF-Pose ${}^A\mathbf{P}_B$ beschreibt die Lage des Koordinatensystems \mathbf{F}_B im Koordinatensystem \mathbf{F}_A (hier mit den Parametern $x = 5, y = 2, z = 3, \Theta = 20^\circ$). Die HTM ${}^B\mathbf{H}_A$ beschreibt die Transformation vom Koordinatensystem \mathbf{F}_A zum Koordinatensystem \mathbf{F}_B und ist damit invers zur Pose.

A.2 Homogene Transformationsmatrizen in 3D

Homogene Koordinaten stammen ursprünglich aus der projektiven Geometrie. Sie werden häufig in der Robotik verwendet, um Punkte zwischen Koordinatensystemen zu transformieren. Um einen Punkt \mathbf{p} in homogenen Koordinaten darzustellen, wird ein Skalierungsfaktor ergänzt, hier immer 1.

Diese homogenen Punkte können anschließend mit sog. Homogenen Transformationsmatrizen (HTMs) transformiert werden. Eine Transformation von einem Koordinatensystem *from* in ein Koordinatensystem *to* ist gegeben durch:

$${}^{\text{to}}\mathbf{H}_{\text{from}} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix}. \quad (\text{A.2})$$

wobei \mathbf{R} eine 3×3 -Rotationsmatrix und \mathbf{t} ein Translationsvektor ist.

Eine 6-DOF-Pose ${}^A\mathbf{P}_B$ aus Anhang A.1 enthält dieselben Informationen und lässt sich alternativ als HTM beschreiben.

Diese HTM setzt sich aus vier Einzelmatrizen zusammen.

Rotation um die x -Achse mit dem Rollwinkel Φ :

$$\mathbf{R}_{[x]}(\Phi) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\Phi) & \sin(\Phi) & 0 \\ 0 & -\sin(\Phi) & \cos(\Phi) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (\text{A.3})$$

Rotation um die y -Achse mit dem Nickwinkel Θ :

$$\mathbf{R}_{[y]}(\Theta) = \begin{bmatrix} \cos(\Theta) & 0 & -\sin(\Theta) & 0 \\ 0 & 1 & 0 & 0 \\ \sin(\Theta) & 0 & \cos(\Theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (\text{A.4})$$

Rotation um die z -Achse mit dem Gierwinkel Ψ :

$$\mathbf{R}_{[z]}(\Psi) = \begin{bmatrix} \cos(\Psi) & \sin(\Psi) & 0 & 0 \\ -\sin(\Psi) & \cos(\Psi) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (\text{A.5})$$

Translation des Koordinatenursprungs um $\mathbf{t} = (t_{[x]} \ t_{[y]} \ t_{[z]})^T$:

$$\mathbf{T}(\mathbf{t}) = \begin{bmatrix} 1 & 0 & 0 & -t_{[x]} \\ 0 & 1 & 0 & -t_{[y]} \\ 0 & 0 & 1 & -t_{[z]} \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (\text{A.6})$$

Aus der Verkettung der drei Rotationen und der Translation ergibt sich die homogene Transformationsmatrix ${}^B\mathbf{H}_A$:

$${}^B\mathbf{H}_A ({}^A\mathbf{P}_B) = \mathbf{R}_{[x]}({}^A\Phi) \cdot \mathbf{R}_{[y]}({}^A\Theta) \cdot \mathbf{R}_{[z]}({}^A\Psi) \cdot \mathbf{T}({}^A\mathbf{t}). \quad (\text{A.7})$$

Bei der Transformation eines Punktes ${}^A\mathbf{p}$ von einem kartesischen Koordinatensystem \mathbf{F}_A in ein zweites Koordinatensystem \mathbf{F}_B wird zuerst die Translation $\mathbf{T}({}^A\mathbf{t})$ durchgeführt. Daran schließen sich die Rotation $\mathbf{R}_{[z]}({}^A\Psi)$ mit dem Gierwinkel ${}^A\Psi$ um die z -Achse, die Rotation $\mathbf{R}_{[y]}({}^A\Theta)$ mit dem Nickwinkel ${}^A\Theta$ um die neue y -Achse und abschließend die Rotation $\mathbf{R}_{[x]}({}^A\Phi)$ mit dem Rollwinkel ${}^A\Phi$ um die neueste x -Achse an. Damit ergibt sich ${}^B\mathbf{p}$ zu:

$${}^B\mathbf{p} = {}^B\mathbf{H}_A \cdot {}^A\mathbf{p}. \quad (\text{A.8})$$

Aus

$$\underbrace{\mathbf{T}^{-1}({}^A\mathbf{t}) \cdot \mathbf{R}_{[z]}^{-1}({}^A\Psi) \cdot \mathbf{R}_{[y]}^{-1}({}^A\Theta) \cdot \mathbf{R}_{[x]}^{-1}({}^A\Phi)}_{{}^B\mathbf{H}_A^{-1}} \cdot \underbrace{\mathbf{R}_{[x]}({}^A\Phi) \cdot \mathbf{R}_{[y]}({}^A\Theta) \cdot \mathbf{R}_{[z]}({}^A\Psi) \cdot \mathbf{T}({}^A\mathbf{t})}_{{}^B\mathbf{H}_A} = \mathbf{I} \quad (\text{A.9})$$

folgt für die Umkehrabbildung:

$${}^B\mathbf{H}_A^{-1}({}^A\mathbf{P}_B) = \mathbf{T}^{-1}({}^A\mathbf{t}) \cdot \mathbf{R}_{[z]}^{-1}({}^A\Psi) \cdot \mathbf{R}_{[y]}^{-1}({}^A\Theta) \cdot \mathbf{R}_{[x]}^{-1}({}^A\Phi) \quad (\text{A.10})$$

$$= \mathbf{T}(-{}^A\mathbf{t}) \cdot \mathbf{R}_{[z]}(-{}^A\Psi) \cdot \mathbf{R}_{[y]}(-{}^A\Theta) \cdot \mathbf{R}_{[x]}(-{}^A\Phi) \quad (\text{A.11})$$

$$= {}^A\mathbf{H}_B(-{}^A\mathbf{P}_B). \quad (\text{A.12})$$

A.3 Zusammenhang Pose vs. HTM

In diesem Abschnitt werden die beiden Begriffe *Heraustransformieren* und *Hineintransformieren* kurz an einfachen Beispielen erläutert.

Die Pose ${}^{\text{veh}}\mathbf{P}_{\text{obj}}$ beschreibt die Pose eines Objekts (obj) im Ego-Koordinatensystem \mathbf{F}_{veh} , also relativ zum eigenen Fahrzeug (vgl. Abschnitt 5.2.3.1). Eine weitere Pose

${}^0\mathbf{P}_{\text{veh}}$ beschreibt die Lage des Ego-Koordinatensystems \mathbf{F}_{veh} in globalen ebenen Koordinaten \mathbf{F}_0 (vgl. Abschnitt 5.2.2.3).

Heraustransformieren bedeutet nun, dass die Pose ${}^{\text{veh}}\mathbf{P}_{\text{obj}}$ aus dem Ego-Koordinatensystem \mathbf{F}_{veh} in die globalen ebenen Koordinaten \mathbf{F}_0 überführt wird, in denen das System \mathbf{F}_{veh} gegeben ist:

$${}^0\mathbf{P}_{\text{obj}} = {}^0\mathbf{P}_{\text{veh}} \cdot \text{out} \left({}^{\text{veh}}\mathbf{P}_{\text{obj}} \right) \quad (\text{A.13})$$

$$= {}^{\text{veh}}\mathbf{H}_0^{-1} \cdot {}^{\text{veh}}\mathbf{P}_{\text{obj}}. \quad (\text{A.14})$$

Hineintransformieren ist die Abbildung in der Gegenrichtung, d. h. eine in globalen ebenen Koordinaten \mathbf{S}_0 gegebene Pose ${}^0\mathbf{P}_{\text{obj}}$ soll in das Ego-Koordinatensystem überführt werden:

$${}^{\text{veh}}\mathbf{P}_{\text{obj}} = {}^0\mathbf{P}_{\text{veh}} \cdot \text{in} \left({}^0\mathbf{P}_{\text{obj}} \right) \quad (\text{A.15})$$

$$= {}^{\text{veh}}\mathbf{H}_0 \cdot {}^0\mathbf{P}_{\text{obj}}. \quad (\text{A.16})$$

A.3.1 Berechnung der Euler-Winkel aus einer HTM

Gegeben sei die Homogene Transformationsmatrix

$$\mathbf{H} = \begin{bmatrix} h_{[1,1]} & h_{[1,2]} & h_{[1,3]} & h_{[1,4]} \\ h_{[2,1]} & h_{[2,2]} & h_{[2,3]} & h_{[2,4]} \\ h_{[3,1]} & h_{[3,2]} & h_{[3,3]} & h_{[3,4]} \\ h_{[4,1]} & h_{[4,2]} & h_{[4,3]} & h_{[4,4]} \end{bmatrix}. \quad (\text{A.17})$$

Aus den Komponenten dieser Matrix lässt sich – in Anlehnung an eine Veröffentlichung von Slabaugh [1999], allerdings angepasst an die am Institut verwendete Rotationsreihenfolge – auf die drei Eulerwinkel schließen. Aufgrund der Mehrdeutigkeiten in den trigonometrischen Funktionen ist die Lösung nicht eindeutig, d. h. unterschiedliche Rotationen führen zur gleichen Lage im Raum. In dieser Arbeit wird folgende Berechnungsvorschrift verwendet:

$$\begin{aligned} \Theta &= -\arcsin\left(h_{[1,3]}\right), \\ \Phi &= \arctan2\left(\frac{h_{[2,3]}}{\cos(\Theta)}, \frac{h_{[3,3]}}{\cos(\Theta)}\right), \\ \Psi &= \arctan2\left(\frac{h_{[1,2]}}{\cos(\Theta)}, \frac{h_{[1,1]}}{\cos(\Theta)}\right). \end{aligned} \quad (\text{A.18})$$

Gleichung (A.18) gilt allerdings nicht für den Fall des „Gimbal-Locks“, bei dem Roll- und Gierachse parallel ausgerichtet sind. Der Gimbal-Lock tritt bei einem Nickwinkel von $\Psi = \pm 90^\circ$ auf, dieser resultiert in $|h_{[1,3]}| = 1$.

Da Einzeldrehungen um Roll- und Gierachse im Gimbal-Lock nicht separat beobachtbar sind, wird der Rollwinkel Φ in diesem Fall zu Null gesetzt (alle anderen Werte wären allerdings ebenfalls zulässig):

$$\begin{aligned}\Phi &= 0, \\ \Theta &= \operatorname{sgn}(h_{[1,3]}) \frac{\pi}{2}, \\ \Psi &= -\operatorname{sgn}(h_{[1,3]}) \Phi + \operatorname{arctan2}(h_{[3,2]}, h_{[3,1]}).\end{aligned}\tag{A.19}$$

Die Parameter der Translation lassen sich folgendermaßen bestimmen:

$$\begin{pmatrix} t_{[x]} \\ t_{[y]} \\ t_{[z]} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \mathbf{H}^{-1} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}.\tag{A.20}$$

B Karten

Dieses Kapitel beschäftigt sich mit der Definition von Karten im Allgemeinen, ihren Anwendungen im Kontext des autonomen Fahrens, sowie im Speziellen mit der Kartenverwendung in dieser Arbeit.

Hier kommt das Kartenwissen für mehrere Aufgaben zum Einsatz: Eine Nebenfunktion ist die Visualisierung von Sensor-, Objekt- und Lokalisierungsdaten überlagert auf einer Karte, um die Interpretation der Daten durch den Benutzer zu erleichtern und die Ergebnisse von Algorithmen einfach qualitativ bewerten zu können.

Die inhaltliche Hauptfunktion ist allerdings die gegenseitige Stützung von Karte und Wahrnehmung. Das bedeutet, dass die Karte einerseits zur Routenplanung genutzt wird und aus der Karte gemäß der geplanten Route und unter Verwendung einer satelliten- oder kartenbasierten Lokalisierung Hypothesen generiert werden, die die Wahrnehmungsmodule bei ihrer Arbeit unterstützen. Dies kann beispielsweise eine grobe Beschreibung einer Kreuzung, in Bezug auf Geometrie und Lage, sein.

Andererseits werden die von den Wahrnehmungsmodulen detektierten Abweichungen zwischen echter Umwelt und der Karte zurückgespeist, um später bessere Hypothesen generieren zu können, oder auch bei kurzfristigem Ausfall der Wahrnehmungsmodule das Fahrzeug anhand von höherwertigen Karteninformationen noch bewegen zu können.

Das Kapitel gliedert sich wie folgt: Zuerst wird in Anhang B.1 der Kartenbegriff definiert, und es werden unterschiedliche Arten von Karten vorgestellt.

Anschließend wird in Anhang B.2 das im Rahmen dieser Arbeit zur Verfügung stehende Kartenmaterial analysiert, bevor in Anhang B.3 die unterschiedlichen Nutzungsarten vorgestellt werden.

B.1 Definition

Unter einer Karte versteht man ein verebnetes, verkleinertes, maßstabsgerechtes und erläutertes Grundrißbild der Erdoberfläche in zeichnerischer Darstellung auf dem Papier oder – zunehmend – auf dem Bildschirm.

[Es] [...] werden nicht nur topographische und thematische Karten, sondern auch Pläne sowie Luft- und Satellitenbilder als Karten bezeichnet. Dem Kriterium der Lagegenauigkeit der Darstellungsinhalte kommt dabei je nach Kartenart und Verwendungszweck eine unterschiedliche Bedeutung zu. (Steingrube [1998, S. 74f])

Der Begriff der „Karte“ ist also mehrdeutig und wird von unterschiedlichen Personen je nach Kontext verschieden interpretiert: Die einen verstehen darunter Luftbilder, andere topographische Karten, und wieder andere topologische Karten. Gemeinsam ist ihnen, dass sie Beschreibungen der Umgebung sind, wobei Darstellungsart und Detaillierungsgrad je nach Anwendungszweck stark variieren können.



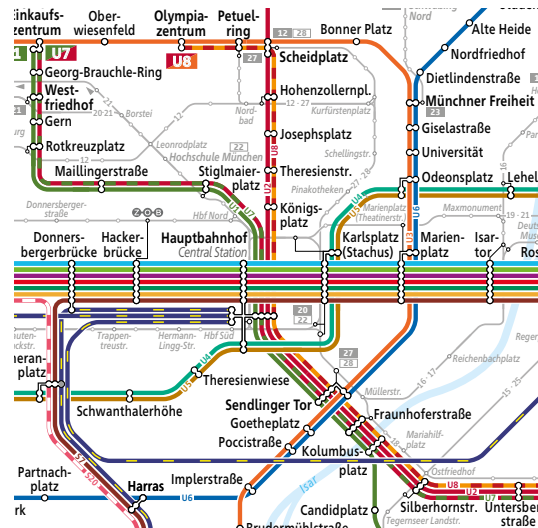
(a) Digitales Orthophoto. Quelle: ZGeoWw



(b) Straßenkarte als Beispiel für eine topographische Karte. Quelle: ADAC



(c) Wanderkarte als Beispiel für eine topographische Karte. Quelle: Kompass Karten



(d) Liniennetzplan des Münchner Verkehrs- und Tarifverbunds als Beispiel für eine topologische Karte. Auszug aus [Münchner Verkehrs- und Tarifverbund GmbH, 2018].

Abbildung B.1: Beispiele für menschenlesbare Kartentypen.

Karten enthalten einerseits geometrische Informationen, wie den Verlauf der Straßen, Wegen oder Grenzen in einem definierten Koordinatensystem. Andererseits enthalten sie aber auch semantische Informationen, beispielsweise ob es sich bei der Straße um eine Autobahn, eine Landstraße oder einen Feldweg handelt, deren Straßennamen oder Angaben zu zulässigen Geschwindigkeiten.

Die meisten Karten werden zur Routenplanung sowie Lokalisierung entweder vom Menschen oder von technischen Systemen genutzt.

B.1.1 Menschenlesbare Darstellungsarten

Betrachten wir zuerst die bekannten, menschenlesbaren Darstellungsarten:

Luftbilder werden entweder aus Satellitenaufnahmen oder aus Aufnahmen aus einem Flugzeug erzeugt. Bei Luftbildern unterscheidet man zwischen Schräg- und Senkrechtaufnahmen [Clerico et al., 2009]. Werden letztere entzerrt und auf eine orthogonale Projektion umgerechnet, so spricht man von Orthophotos (siehe Abbildung B.1a). Diese sind als Digitales Orthophoto (DOP) für die meisten Bereiche mit Auflösungen im Dezimeter-Bereich oder besser verfügbar [Landesamt für Digitalisierung, Breitband und Vermessung, 2021]. Neben dem gezeigten RGB-Bild (Rot-Grün-Blau) sind bei vielen Anbietern weitere Varianten verfügbar: Color-Infrarot-Bilder (CIR) verfügen über die Kanäle NIR (Nahes Infrarot, engl. *Near-Infrared*), Rot und Grün, während RGBI-Bilder alle 4 Kanäle kombinieren.

Topographische Karten sind mittel- bis großmaßstäbige Karten und dienen zur genauen Abbildung der Topografie (Geländeform) und anderer sichtbarer Details der Erdoberfläche. Das Gelände wird üblicherweise durch Höhenlinien visualisiert, zusätzlich sind markante Höhenpunkte (Berggipfel, Sättel, usw.) sowie der Verlauf von Gewässern (Seen, Flüsse, Gräben, usw.) verortet. Zum Inhalt solcher Karten gehören auch künstlich geschaffene Strukturen wie Straßen, Bahnlinien, größere Gebäude, die Umrisse von Orten, aber auch der Verlauf von Grenzen oder Stromleitungen. Alle diese geografischen Objekte werden entsprechend des Karten-Maßstabs lagerichtig und vollständig wiedergegeben, hierzu kommt ein System kartografischer Zeichen zum Einsatz.

Die Kartengrundlage bilden fast immer amtliche topographische Karten der Landesvermessungsämter oder – bei sehr kleinräumigen Projekten – die Karten des amtlichen Liegenschaftskatasters.

Außer den amtlichen Kartenwerken gibt es eine Vielzahl verschiedenster Kartenprodukte (Auto-, Wander-, Stadtkarten usw.), deren Herausgeber oftmals Verlage mit eigenen kartographischen Anstalten sind. Für die räumliche Planung besitzen diese nichtamtlichen Kartenwerke lediglich Ergänzungscharakter. Ihnen können ggf. in Bereichen, die die amtliche Statistik sehr schlecht abdeckt, Hinweise auf Einrichtungen entnommen

werden. So sind insbesondere im Bereich des Freizeit- und Erholungswesens die kommerziellen Kartenwerke sehr viel besser informiert und aktueller als die amtlichen. (Steingrube [1998, S. 74f])

Amtliche topographische Karten werden auf gesetzlicher Grundlage aus Steuermitteln hergestellt und dienen vor allem der öffentlichen Daseinsvorsorge und Sicherheit sowie der Landesverteidigung. Hier bezieht das Zentrum für Geoinformationswesen der Bundeswehr (ZGeoBw) sein Kartenmaterial für das Inland weitgehend aus den amtlichen topografischen Landeskartenwerken.

Zwei Beispiele für menschenlesbare topographische Karten sind in Abbildung B.1b sowie Abbildung B.1c gezeigt. Alternativ zu dieser gerasterten Darstellung sind die digitalen topographischen Karten auch in einer Vektordarstellung verfügbar, welche für den Einsatz in technischen Systemen besser geeignet ist.

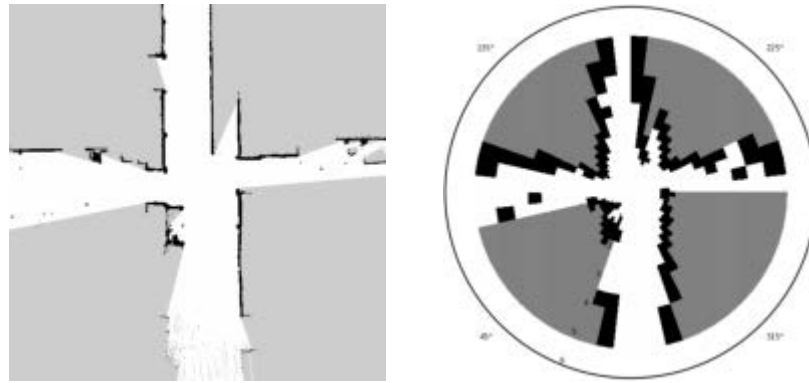
Topologische Karten verzichten im Gegensatz zu den topographischen Karten auf die präzise geometrische Verortung der einzelnen Kartenelemente. Wichtig ist hier vor allem die Relation zwischen den Elementen, also eine Wegebeziehung von einem Ort zum anderen. Eine Variante sind Topogramme, bekannte Beispiele hierfür sind Liniennetzpläne, wie sie für die öffentlichen Verkehrsmittel zum Einsatz kommen. Abbildung B.1d zeigt hier exemplarisch das Nahverkehrsnetz von München mit S-Bahn-, U-Bahn-, Regionalzug- und Tramlinien.

B.1.2 Technisch geprägte Darstellungsarten

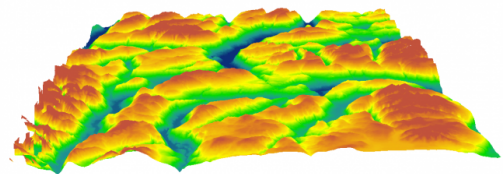
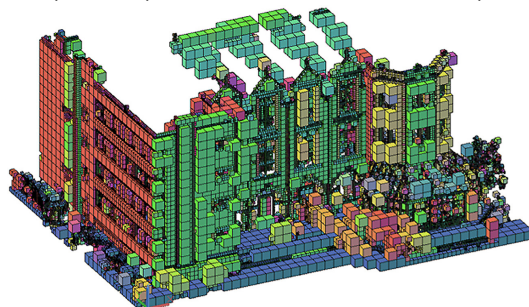
Im technischen Bereich existieren weitere Kartentypen, von denen einige in der Robotik und im autonomen Fahren gebräuchliche hier kurz vorgestellt werden.

2D-Rasterkarten (auch als 2D-Gitterkarten bezeichnet, engl. Grid Maps) sind seit langem eine weit verbreitete Kartenart [Elfes, 1989, Thrun, 2002] und dienen in der Robotik der abstrakten Darstellung unterschiedlicher Informationen, z. B. Hindernisse, Befahrbarkeiten oder Höhen für jede einzelne Zelle. Eine einfache Abstraktion ist das Karopapier, in dem einzelne Kästchen geschwärzt werden. Je nach Anwendungszweck kommen unterschiedliche Kartengrößen und Auflösungen zum Einsatz.

Abbildung B.2a zeigt dieselbe Szene in einem Gebäude sowohl in einer metrischen kartesischen Rasterkarte als auch in einer polaren Rasterkarte. In der metrischen Rasterkarte ist die Auflösung, d. h. Größe der Zellen, überall gleich, während die Auflösung in der polaren Rasterkarte nach außen abnimmt. Die polare Darstellung ist im Vergleich zur kartesischen wesentlich geringer verbreitet. Während andere Kartentypen meist in einem weltfesten Koordinatensystem wie F_{UTM} gegeben sind, kommen 2D-Rasterkarten sowohl weltfest als auch als mitbewegte Karten im Ego-Koordinatensystem F_{veh} zum Einsatz.

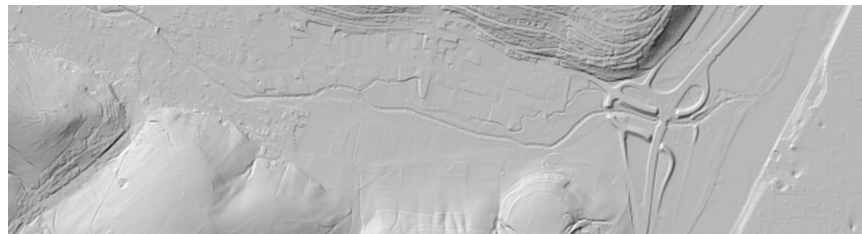


(a) 2D-Rasterkarten: links eine kartesische und rechts eine polare Gitterrepräsentation derselben Szene in einem Gebäude. Die Farbe bzw. Helligkeit kodiert hier Freiraum (weiß), Hindernisse (schwarz) und unbeobachteten Raum (grau). Quelle: [Pronobis und Rao, 2017, Fig. 3a].

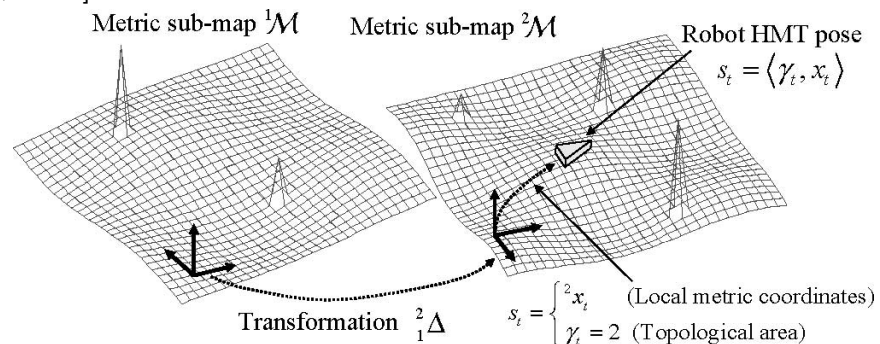


(b) 3D-Voxelkarte, hier am Beispiel eines sog. Ocotrees mit wechselnden Zellgrößen. Die Farbkodierung dieser Abbildung ergibt sich aus dem Normalenvektor. Quelle: [Vo et al., 2015, Fig. 3].

(c) Digitales Höhenmodell aus der Space Shuttle Radar Topography Mission (SRTM) im Jahr 2000. Die Höhe ist farblich als Heatmap kodiert (blau niedrig, rot hoch). Quelle: [GIS Geography, 2022].



(d) Digitales Geländemodell DGM5, Zellgröße $5\text{ m} \times 5\text{ m}$. Die unterschiedlichen Höhen sind anhand der Schattierungen erkennbar. Quelle: [Landesamt für Digitalisierung, Breitband und Vermessung, 2022a].



(e) Metrisch-topologische Karte: Zwei metrische 2D-Rasterkarten ${}^1\mathcal{M}$ und ${}^2\mathcal{M}$. Jede 2D-Rasterkarte hat ein lokales Koordinatensystem, welche über die Transformation ${}^2_1\Delta$ miteinander verbunden sind. Quelle: [Blanco et al., 2008, Fig. 2].

Abbildung B.2: Beispiele für technisch geprägte Kartentypen.

Die Orthophotos aus Abbildung B.1a sind eine besondere Art der kartesischen 2D-Rasterkarten – jedes Pixel repräsentiert eine Gitterzelle, in der ein Farbwert gespeichert ist.

3D-Voxelkarten sind die dreidimensionale Erweiterung der 2D-Rasterkarten und meist im kartesischen System gegeben. Aus der Gitterzellen werden so Würfel, die mit entsprechenden Informationen (z. B. Belegungswahrscheinlichkeit oder Farbe) annotiert sind. In der Robotik haben 3D-Voxelkarten den Vorteil, dass überhängende Hindernisse modelliert werden können.

Die sog. Octrees sind eine Besonderheit der 3D-Voxelkarten, da hier innerhalb der Karte mit unterschiedlichen Auflösungen gearbeitet wird, wie in Abbildung B.2b dargestellt. Im Bereich ausgedehnter Flächen genügt eine niedrigere Auflösung, während an Ecken und Kanten eine höhere Auflösung notwendig ist.

Digitale Höhenkarten sind eine weitere Unterart der metrischen kartesischen 2D-Rasterkarten, ein Beispiel ist in Abbildung B.2c gezeigt. Die Höhenkarten können einerseits aus Satellitendaten erzeugt werden – dann ist eine globale Verfügbarkeit möglich, allerdings nur mit geringerer Auflösung. Andererseits können sie aus luftgestützten LiDAR-Messungen (Light Detection And Ranging) im Rahmen von Befliegungen – oftmals gleichzeitig mit der Erfassung von Luftbildern (vgl. Anhang B.1.1) – generiert werden. Diese Karten verfügen über eine wesentlich höhere Auflösung und Genauigkeit der Höhe, sind allerdings nicht global verfügbar.

Das DHM (Digitales Höhenmodell) dient hier als Oberbegriff für Höhenkarten, ist aber wissenschaftlich nicht präzise definiert [Li et al., 2005] und unterscheidet sich bezüglich der repräsentierten Oberflächen. Konkreter sind die beiden folgenden Modelle definiert.

„Ein Digitales Geländemodell (DGM) beschreibt die Erdoberfläche ohne Vegetation und Bebauung als eine in der Lage und Höhe bekannte Punktwolke. Diese liegt als regelmäßiges Gitter oder in Form von Laserpunkten vor [Landesamt für Digitalisierung, Breitband und Vermessung, 2022a]“. Über die Vermessungsämter ist das DGM in unterschiedlichen Bodenauflösungen verfügbar, z. B. als DGM1 mit 1 m Bodenauflösung, DGM5 oder DGM25. Abbildung B.2d zeigt eine exemplarische Szene. Je höher die Bodenauflösung bzw. kleiner die Zelle, desto größer ist auch die mögliche Höhenauflösung – letztere ist vom Anwendungszweck getrieben und hat starke Auswirkungen auf Aufwand und Preis in der Modellerzeugung.

„Das DOM (Digitales Oberflächenmodell) zeigt die Erdoberfläche inklusive der darauf befindlichen Objekte (z. B. Vegetation und Gebäude) in Gitterform. Die [in Bayern] aktuelle Gitterweite beträgt 40 cm (entspricht 6,25 Pkt/m²) [Landesamt für Digitalisierung, Breitband und Vermessung, 2022b]“.

Metrisch-topologische Karten sind eine Variante der topologischen Karten, bei denen zusätzlich zu den Relationen auch metrische Teilkarten abgespeichert werden. Metrisch-topologische Karten können als Graph interpretiert werden: diese einzelnen Teilkarten sind die Knoten, welche über Kanten (die Relation zueinander) verbunden sind.

Ein Beispiel für diese Karten sind metrische 2D-Rasterkarten einzelner Räume eines Gebäudes [Thrun, 1998, Sec. 3], die von einem Roboter mittels SLAM-Verfahren (Simultaneous Localization and Mapping) ohne große Fehler kartiert werden können. Die Verbindung der Räume wird dann als Graph modelliert – so können auch größere Etagen und auch unterschiedliche Stockwerke abgebildet werden.

Diese Darstellung hat mehrere Vorteile gegenüber einer großen metrischen 2D-Rasterkarte: die schlanke Modellierung des Graphen ermöglicht eine effiziente schnelle globale Routenplanung – die metrischen Teilkarten kommen erst während der Lokalisierung und Navigation im jeweiligen Raum zum Einsatz. Zusätzlich wirken sich Integrationsfehler der Roboterposition während der Kartierung, bspw. durch Drift in der Odometrie, nicht auf die komplette Karte aus.

Abbildung B.2e zeigt exemplarisch eine metrisch-topologische Karte aus dem Outdoor-Bereich.

B.2 Analyse Kartenmaterial

In diesem Abschnitt werden mehrere Varianten von Vektorkarten für die Nutzung untersucht. Diese müssen mindestens Informationen zum Straßen- und Wegenetz enthalten und routingfähig sein.

B.2.1 Vektorbasiskarte 25

Die Vektorbasiskarte 25 (VBK 25) ist eine digitale topographische Vektorkarte im Maßstab 1:25 000 des ZGeoBw bzw. seiner Vorgängerorganisation AGeoBw (Amt für Geoinformationswesen der Bundeswehr), die im Shape-Datenformat bereitgestellt wird. Die VBK 25 enthält verschiedene Kartenebenen mit Linien-, Flächen- und Punktmerkmalen in WGS 84-Koordinaten (World Geodetic System 1984).

Die vom ZGeoBw bereitgestellten Shape-Daten wurden zuerst auf Genauigkeit und Detaillierungsgrad untersucht, um ihre Eignung für die Verwendung in den Autonomiemodulen festzustellen:

Linien-Layer

- In den Linien-Layern sind Merkmale diverser Kategorien enthalten. Neben Wegen unterschiedlicher Klassifizierung sind dies z. B. der Verlauf von Stromleitungen oder Gräben.

- Straßen und Feld- bzw. Waldwege sind fast ausnahmslos in den Linien-Layern enthalten.
- Die Linien-Layer sind sehr grob aufgelöst, d. h. Wegpunkte liegen oft im Abstand >100 m zueinander. Dadurch sind vor allem Kurven schlecht abgebildet, und geschlängelte Wege erscheinen in der Karte als Gerade.
- Vorhandene Wegpunkte und vor allem Kreuzungen liegen allerdings recht gut auf den tatsächlichen Wegen, angegeben ist eine Genauigkeit von 3m.
- Wege sind als einzelne Segmente, jeweils von Kreuzung zu Kreuzung, modelliert. Die Endpunkte mehrerer Segmente an derselben Kreuzung liegen im Zentimeterbereich beieinander.
- Das Shape-Datenformat erlaubt eine Annotation jedes einzelnen Elements pro Layer mit Attributen wie Spuranzahl, Spurbreite, Straßename, Fahrrichtung oder Höchstgeschwindigkeit. Allerdings sind diese Felder in den Daten von ZGeoBw zum Großteil nicht gefüllt, sodass auf ihre Nutzung bislang verzichtet wird.
- Die einzige Möglichkeit, auf die Beschaffenheit der Straße zu schließen, ist die Zugehörigkeit zu bestimmten Layern (z. B. Autobahn, Landstraße oder Waldweg).
- Folgende Layer der VBK 25 wurden für die Verwendung als Straßennetz in unserem Testgelände südlich von München ausgewählt. Namen und Beschreibung entsprechen der zugehörigen Dokumentation aus [Thelker Geoinformation, 2005], teilweise ergänzt mit Beschreibungen aus [AdV-Arbeitsgruppe ATKIS, 2003]. Der erste Buchstabe bezeichnet die Hauptkategorie, A steht für „Künstliche Bodenabdeckung“. Der zweite Buchstabe beschreibt die Subkategorie, P für „Transportsystem: Straße“, Q für „Zugehörige Transporteinrichtung“.

AP010L Fahrweg/Piste Ein nicht ausgebauter Verkehrsweg für nicht schienengebundene Fahrzeuge.

AP030L Straße Ein befestigter Verkehrsweg für nicht schienengebundene (Kraft-) Fahrzeuge.

AP501L Weg Ein befestigter oder unbefestigter Geländestreifen, der zum Befahren und/oder Begehen vorgesehen ist.

AQ065L Durchlass Ein Bauwerk zur Unterführung eines Wasserlaufs unter einer Straße, einem Schienenweg oder einer Aufschüttung.

Straßen höherer Ordnung wurden bisher bewusst ausgeschlossen, da sich dieses Projekt mit der Navigation auf Feld- und Waldwegen, nicht aber mit Bundesstraßen und Autobahnen beschäftigt. Ebenso zu schmale Wege, die nicht für eine Befahrung mit einem Kraftfahrzeug geeignet sind. Dies sind u. a. folgende Layer:

AP534L Straßenkörper Befestigter, in erster Linie dem Straßenverkehr dienender Verkehrsweg einschließlich der auf Über- und Unterführungen (Tunnel) verlaufenden Abschnitte ohne die Fahrbahn selbst.

AP537L Fahrbahn Befestigter und in Fahrstreifen unterteilter Bereich innerhalb eines Straßenkörpers, der in erster Linie dem fließenden Verkehr dient. Zu der Fahrbahn gehören auch befestigte Seiten- und kleinere Trennstreifen, Stand- und Kriechspuren. Begleitende Fuß- und Radwege werden ebenfalls der Fahrbahn zugerechnet, so weit sie nicht durch Zwischenräume von mehr als 3 m Breite von der Fahrbahn getrennt sind.

AP551L Straßenelement Beschreibt eine Strecke zwischen zwei Punkten, in denen eine verkehrliche Verbindung vorliegt. Das Linienelement beginnt und endet jeweils an einem Verbindungspunkt und enthält die Fahrbahnachse als Geometrie.

AP050L Pfad/Fußweg Ein von Menschen oder Tieren ausgetretener Pfad.

Flächen-Layer

- In den Flächen-Layern sind z. B. Siedlungen, Waldgebiete, Ackerflächen oder Gewässer verzeichnet.
- Die Konturen sind nicht auf den Meter exakt, geben aber einen guten Hinweis auf die Umgebung. Angegeben ist eine Genauigkeit von 10 m.
- Seit 2010 werden diese Informationen zur Routenplanung innerhalb der Teilmissionen sowie im Fähigkeitsnetz zur Parametrierung der Fähigkeiten genutzt.
- Folgende Layer wurden für die Nutzung in unserem Testgelände südlich von München ausgewählt. Namen und Beschreibung entsprechen der zugehörigen Dokumentation aus [Thelker Geoinformation, 2005], teilweise ergänzt mit Beschreibungen aus [AdV-Arbeitsgruppe ATKIS, 2003].
Der erste Buchstabe bezeichnet die Hauptkategorie, A steht für „Künstliche Bodenabdeckung“, B für „Hydrographie/Gewässerkunde“, E für „Bodenbewachsung“. Der zweite Buchstabe beschreibt die Subkategorie, A für „Landwirtschaftliche Anbaufläche“, B für „Weideland“, C für „Waldgebiete“.

AL020F Bebautes Gebiet Bebautes Gebiet enthält neben 'Wohnbau-', 'Industrie- und Gewerbefläche', 'Fläche gemischter Nutzung', 'Fläche besonderer funktionaler Prägung' auch die dazu in einem engen räumlichen und funktionalen Zusammenhang stehenden Flächen des Verkehrs, der Gewässer, der Flächen, die von 'Bauwerke und sonstige Einrichtungen' für Erholung, Sport und Freizeit sowie von 'Vegetationsflächen' belegt sind. Die Grenze des Bebauten Gebiets zur Feldlage oder zu Waldflächen wird in der Regel durch die Grenzen der bebauten Grundstücke unter Einbeziehung der Hofraumflächen und Hausgärten gebildet. Der Umring bildet einen geschlossenen Linienzug. Die Ortslage kann Objekte des Objektbereichs Vegetation als Inseln umschließen.

BH015F Moor Ein schlecht entwässertes oder periodisch überflutetes Gebiet mit Ausnahme von Marsch- und Wattgebieten, dessen Boden mit Pflanzenresten angereichert ist.

BH030F Graben Ein künstlich angelegter Wasserlauf ohne Strömung oder mit kontrolliertem Fluss zum Trockenlegen oder Bewässern von Gebieten.

BH080F See/Teich Eine von Land umgebene Wasserfläche.

BH095F Sumpf Eine Fläche mit wassergesättigtem, zeitweise unter Wasser stehenden Untergrund, zu deren Bewuchs auch Bäume gehören.

BH140F Fluss/Wasserlauf Ein natürliches, fließendes Gewässer.

EA010F Ackerland Eine Fläche für den Anbau von Feldfrüchten.

EB010F Grünland Eine Fläche mit natürlichem Pflanzenbewuchs mit geringem oder keinem Holzpflanzenanteil.

EC015F Wald/Forst Eine (baumbestandene) Fläche, die der forstwirtschaftlichen Nutzung oder der Erholung dient.

Punkt-Layer

- Als Punktmerkmale sind bspw. Strommasten oder besonders markante Bäume kartiert.
- Elemente aus den Punkt-Layern werden bislang noch nicht genutzt. Allerdings sind diese Elemente als potentielle Landmarken in den Folgeprojekten zu verwenden.

B.2.2 OpenStreetMap

Das OSM-Projekt (OpenStreetMap) [OpenStreetMap, o.J.] ist ein freies und nicht-kommerzielles Projekt, dessen Ziel es ist, freie Kartendaten bereitzustellen. Die Daten werden von beliebigen Benutzern zusammengetragen und aktualisiert. Diese Karten dürfen unter Angabe der Quelle kostenlos benutzt und verändert werden.

Über www.geofabrik.de werden kostenlos ins Shape-Format konvertierte Versionen der OSM-Daten angeboten. Diese lassen sich analog zu den VBK 25-Daten vom ZGeoBw im Softwareframework nutzen.

Folgende Layer sind hier vorhanden:

Linien-Layer

Roads Straßen und Wege unterschiedlichster Kategorie, hier zusammengefasst auf einen Layer.

Waterways alle Arten von Fließgewässern, Wasserwegen und dazugehörigen Elementen.

Railways Schienenwege.

Flächen-Layer

Natural beschreibt physische und geologische Landschaftselemente sowie natürliche Vegetationsformen.

Landuse beschreibt die überwiegende Nutzung von Gebieten bzw. Flächen.

Buildings Pro Gebäude ist in diesem Layer ein Polygon enthalten.

Punkt-Layer

Points Punktmerkmale, die unterschiedlich attributiert sind (z. B. Telefonzellen, ...)

Places Ortsnamen, Stadtteilnamen

Im Gegensatz zu den Daten in der VBK 25 sind die OSM-Karten von wechselhafter Qualität. In städtischen Gebieten ist die Genauigkeit und Detailliertheit meist besser, in ländlichen Gebieten oft schlechter.

Die OSM-Karten wurden daher im Rahmen dieser Arbeit nicht für das autonome Fahren verwendet.

B.2.3 Selbst erstellte Karten

Die Kartendaten sollen u. a. für das GNSS- (Globales Navigationssatellitensystem, engl. *Global Navigation Satellite System*) und Karten-basierte Fahren (d. h. ohne aktive Wahrnehmungsmodule) genutzt werden, hierfür ist eine Genauigkeit der Karte im Zentimeterbereich erforderlich. Die Genauigkeit der Linien-Layer aus der VBK 25 genügt nicht. Daher wurde im Testgebiet eine eigene Karte erstellt, hierzu kommen Aufzeichnungen aus mehreren Befahrungen zum Einsatz. In einem Geo-Tool wie QGIS können die als KML-Datei (Keyhole Markup Language) vorliegenden Aufzeichnungen per Hand zu einem Straßennetz verknüpft werden. Daraus ergibt sich ein zusätzliches Linien-Layer in der Karte, das an den Kreuzungen mit den Layern aus der VBK 25 verbunden werden kann.

Zusätzlich wurden weitere Flächen-Layer ergänzt, die ebenfalls leicht per Hand mit QGIS erstellt werden können. Diese werden für die Missionsplanung sowie die Parametrierung der Wahrnehmungsfähigkeiten genutzt (siehe Abschnitt 7.5).

No-Go-Areas Diese Bereiche sind bei der Routenplanung auf jeden Fall zu vermeiden.

Vision-Critical-Area In diesen Gebieten ist kein Einsatz der visuellen Spur- und Kreuzungserkennung erlaubt.

GNSS-Critical-Area In diesen Gebieten ist kein Rückfall auf die GNSS-basierte Spurgenerierung erlaubt.

Map-Critical-Area Innerhalb dieser Gebiete wird – basierend auf der mittels Konvoi oder visueller Spurerkennung gefahrenen Strecke – die Karte während der Durchfahrt aktualisiert. Diese Gebiete werden vom Bediener vor der Fahrt speziell markiert.

B.3 Nutzung Kartenmaterial

Das im vorherigen Abschnitt analysierte Kartenmaterial kommt für unterschiedliche Zwecke auf verschiedenen Ebenen zum Einsatz. In den folgenden Abschnitten werden diese top-down erläutert: von der globalen Routenplanung über Hypothesengenerierung bis zur Backup-Sollspurgenerierung.

B.3.1 Kartenimport

Die einzelnen Dateien der Karte (Linien- und Flächenlayer) werden in ein Instituts-eigenes Kartenformat für den Online-Betrieb umgewandelt. Diese Karte besteht aus einzelnen Linien-Segmenten. Deren Punkte können neben der GNSS-Koordinate mit weiteren Informationen annotiert werden.

Zur Netzgenerierung werden zuerst alle Linien-Layer aus den Shape-Files analysiert und jedes Liniensegment zu einem gemeinsamen Wegenetz hinzugefügt. Optional – und vor allem bei großen Karten sinnvoll – werden dabei nur die Liniensegmente hinzugefügt, die sich innerhalb eines gewünschten Bereichs befinden. Bei diesem ersten Wegenetz handelt es sich um einen ungerichteten Graphen.

Für die spätere Routenplanung sind gerichtete Kanten in diesem Graph erforderlich. Diese werden mittels Duplizierung aller Segmente im Wegenetz erzeugt, dabei wird die Reihenfolge der Wegpunkte innerhalb jedes Segments vertauscht sowie die Segmente lateral auseinander geschoben (Rechtsverkehr). Da im Normalfall keine Information über die Spurbreite vorliegt, wird von einspurigen Straßen ausgegangen, und die Spuren werden nur um wenige Zentimeter verschoben. Jede Spur bleibt dabei logisch mit ihrer Gegenspur verknüpft. Wenn in den Linien-Layern Einbahnstraßen-Informationen verfügbar wären, könnten diese durch eine Nicht-Duplizierung hier einfach berücksichtigt werden.

Anschließend werden die Flächen-Layer behandelt. Sämtliche Punkte der Karte werden überprüft, ob sie innerhalb eines der Polygone der Flächenlayer (oder innerhalb eines

einstellbaren Bereichs rund um dieses Polygon) liegen und ggf. mit dieser Information annotiert.

Abschließend wird eine Analyse der Segmente auf Kreuzungen und weitere Verknüpfungen durchgeführt. Liegt dabei ein Endpunkt des Segments innerhalb eines definierten Radius um einen Startpunkt eines anderen Segments, so werden diese Punkte logisch verknüpft und zusätzlich als „Kreuzung“ annotiert. Falls erforderlich, findet eine kleine Verschiebung der Start- und Endpunkte entlang des Segments von der Kreuzung weg statt, um an einer Kreuzung keine sich schneidenden Linien zu erhalten.

B.3.2 Globale Routenplanung

Eine typische Aufgabe für ein autonomes Fahrzeug ist die Navigation vom aktuellen Standpunkt p_{ego} zu einem definierten Zielpunkt p_{target} , der z. B. über GNSS-Koordinaten beschrieben ist. Um dabei eine geeignetere Route als „auf der Luftlinie querfeldein“ in Richtung Ziel fahren zu können, kann auf Kartenmaterial zurückgegriffen werden. Ziel ist die Ermittlung einer „optimalen“ Route, wobei sich optimal auf unterschiedlichste Kriterien beziehen kann. Handelsübliche PKW-Navigationssysteme (Personenkraftwagen) berechnen hier die kürzeste, die schnellste, die energieoptimale oder auch die landschaftlich reizvollste Route. Diese Unterscheidung wird in diesem Projekt nicht getroffen, optimal ist nach unserer Definition die kürzeste Route. Die anderen Typen wie „schnell“ oder „schön“ können anhand der vorliegenden Straßenkarte nicht berechnet werden, da in ihr keine Information über eine segmentabhängige Maximalgeschwindigkeit oder Sehenswürdigkeiten enthalten sind.

B.3.2.1 Berücksichtigung von Flächen-Layern

Anhand der vorliegenden Flächen-Layer können Gebietstypen festgelegt werden, die bei der Routenplanung zu umgehen sind. Folgende Layer können derzeit in der Missionsplanung (siehe Abschnitt 7.5.1) ausgewählt werden und werden bei der Routenplanung berücksichtigt:

Wald In diesen Gebieten ist mit reduzierter Helligkeit, die zu Problemen in den Bildverarbeitungsalgorithmen führen kann, sowie mit GNSS-Abschattung zu rechnen. Es ist kein Rückfall auf die GNSS-basierte Spurgenerierung erlaubt.

GNSS-Critical-Area In diesen Gebieten ist kein Rückfall auf die GNSS-basierte Spurgenerierung erlaubt.

Bebautes Gebiet In diesen Gebieten ist mit einer erhöhten Anzahl dynamischer Objekte auf dem Straßennetz zu rechnen, außerdem bei hoher Bebauung mit GNSS-Abschattung. Daher ist eine Vermeidung bei der Routenplanung zu präferieren.

Map-Critical-Area Innerhalb dieser Gebiete wird – basierend auf der mittels Konvoi oder visueller Spurerkennung gefahrenen Strecke – die Karte während der Durchfahrt aktualisiert. Diese Gebiete werden vom Bediener vor der Fahrt speziell markiert.

No-Go-Areas Diese Bereiche sind bei der Routenplanung auf jeden Fall zu vermeiden.

Vision-Critical-Area In diesen Gebieten ist kein Einsatz der visuellen Spur- und Kreuzungserkennung erlaubt.

B.3.2.2 Lösungsmöglichkeiten für das Shortest Path Problem

Die Routenplanung auf einem Wegenetz lässt sich zum bekannten *Shortest Path Problem* (Bestimmung des kürzesten Pfads) aus der Graphentheorie zurückführen: Ein Graph besteht aus Knoten (den Orten bzw. Kreuzungen) und Kanten (den Wegeverbindungen zwischen den Knoten), wobei jeder Kante ein Gewicht (Kosten, z. B. die Weglänge) zugeordnet ist. Ziel ist es, genau den Pfad zwischen zwei Knoten zu finden, bei welchem die Summe der Gewichte minimiert wird.

Weit verbreitete Algorithmen zur Lösung dieses Problems sind der Algorithmus von Dijkstra [1959], der A*-Algorithmus [Hart et al., 1968] oder der Bellman-Ford-Algorithmus [Bellman, 1958, Ford, 1956]. Da bei einem Wegenetz alle Wege eine positive Länge haben, sind alle Gewichte der Kanten positiv und somit die Voraussetzungen für den Einsatz des Dijkstra-Algorithmus erfüllt.

Jeder Weg eines Wegenetzes kann prinzipiell in beiden Richtungen befahren werden, die notwendigen Schritte wurden bereits beim Import der Karten in Anhang B.3.1 gezeigt. Das neue Wegenetz kann daher als gerichteter Graph betrachtet werden.

In der Routenplanung soll die aktuelle Position des Fahrzeugs \mathbf{p}_{ego} als Startpunkt verwendet werden, der Zielpunkt $\mathbf{p}_{\text{target}}$ ist ein Parameter der aktuellen Aufgabe. Da nicht gewährleistet ist, dass sich \mathbf{p}_{ego} und $\mathbf{p}_{\text{target}}$ direkt auf Knoten des Graphen befinden, werden zuerst korrespondierende Punkte auf den gerichteten Kanten bestimmt. Für das Fahrzeug wird hierbei sein Gierwinkel Ψ_{ego} mit der Richtung der beiden gerichteten Kanten verglichen. Die Kante mit der kleineren Winkeldifferenz wird ausgewählt, auf dieser wird der Startpunkt $\mathbf{p}_{\text{egoOnTrack}}$ bestimmt. Für den Zielpunkt wird die „Straßenseite“ betrachtet, also die Kante mit dem geringeren Abstand zu $\mathbf{p}_{\text{target}}$, und auf dieser der nächstgelegene Punkt des Straßennetzes $\mathbf{p}_{\text{targetOnTrack}}$ bestimmt. Damit sind erste und letzte Kante des Pfads bereits festgelegt. Die Routenplanung erfolgt nun ausgehend von dem auf die Startkante folgenden Knoten zum der Zielkante vorausgehenden Knoten, also von $\mathbf{p}_{\text{egoOnTrack}}$ zu $\mathbf{p}_{\text{targetOnTrack}}$.

Zusätzlich zum primären Gewicht der Kanten, das auf der Länge der Kante bzw. des Weges beruht, können die Wegsegmente mit Zusatzkosten belegt werden: sind in der aktuellen Aufgabe zu vermeidende Bereiche wie *No-Go-Areas* oder Waldflächen parametrisiert, so wird das Gewicht aller Kanten, die innerhalb solch eines Bereichs verlaufen, um einen empirisch bestimmten Wert vergrößert.

B.3.2.3 Ausgangsdaten der Routenplanung

Anhand der optimierten Route $\mathcal{P}_{\text{route}}$ werden aus dem Graph für jede enthaltene Kante die zugrundeliegenden semantisch annotierten Wegpunkte $\mathcal{P}_{\text{routeAnnotated}}$ extrahiert. Bei der Annotation handelt es sich bspw. um Geschwindigkeitsinformationen oder um Flächenzugehörigkeiten wie *Wald*. Zusätzlich wird aus dem Wegenetz für jeden passierten Knoten eine geometrische Beschreibung der korrespondierenden Abzweigung erzeugt. Die Sequenzen von Wegpunkten $\mathcal{P}_{\text{routeAnnotated}}$ und Kreuzungen $\mathcal{C}_{\text{route}}$ werden zur Weiterverwendung bereitgestellt, bspw. zur Hypothesengenerierung, für die GNSS-basierten Rückfallebenen zur Fahrspur- und Kreuzungsgenerierung (siehe Abschnitte 7.3.4.3 und 7.3.5.2) oder den hierarchischen Zustandsautomaten (siehe Abschnitt 7.4).

B.3.3 Hypothesen

Neben der globalen Routenplanung dienen die Kartendaten primär zur Generierung von Hypothesen für die Wahrnehmungsmodule.

In diesem Abschnitt werden zuerst mögliche Hypothesen erläutert, bevor auf deren Generierung eingegangen wird. Abschließend werden kurz die unterschiedlichen Nutzungen der Hypothesen aufgeführt.

B.3.3.1 Wegverlauf und Kreuzungen

Zwei Varianten von Hypothesen werden erzeugt, der Spurverlauf sowie Kreuzungen. Beide werden im Ego-Koordinatensystem \mathbb{F}_{veh} beschrieben.

Der Spurverlauf besteht aus einer Sequenz von zweidimensionalen Punkten mit Richtungen (x, y, Ψ) , die typischerweise 200 m lang und um das Ego-Fahrzeug zentriert ist.

Eine Kreuzungshypothese besteht aus einem Kreuzungsmittelpunkt (x, y) sowie n Kreuzungsästen. Der erste Ast ist der „eingehende“, also der, auf dem das Fahrzeug die Kreuzung erreicht. Der zweite Ast ist der „ausgehende“, auf dem das Fahrzeug nach Passieren der Kreuzung weiterfährt. Es folgen $n - 2$ weitere Äste, die für das Fahren an sich irrelevant, aber für die Wahrnehmungsmodule von Relevanz sind. Jeder Ast ist als Klothoide (Krümmung, Krümmungsänderung, Breite) modelliert, die in einem definierten „globalen“ Winkel die Kreuzung verlässt. Die vollständige Schnittstellenbeschreibung für die Kreuzungen ist in Tabelle 7.3 zu finden.

B.3.3.2 Hypothesengenerierung

Die Hypothese des Spurverlaufs wird in mehreren Schritten aus der globalen Karte erzeugt. Dazu erfolgt einmalig pro globalem Ziel:

Routenplanung Die Routenplanung liefert eine Sequenz von Wegsegmenten, beginnend an der Ego-Position und endend an der Zielkoordinate. Alle Punkte in $\mathcal{P}_{\text{routeAnnotated}}$ sind noch in globalen Koordinaten gegeben. Details siehe Anhang B.3.2.

Upsampling Da die Abstände der Wegpunkte aus der globalen Karte sehr unterschiedlich sind (von wenigen Metern bis hin zu hunderten Metern), werden Zwischenpunkte ergänzt. Das Upsampling dient zur Sicherstellung einer gleichmäßigen Punktdichte für den nachfolgenden Schritt.

Smoothing Hier kommt der Levenberg-Marquardt-Algorithmus als Werkzeug zur numerischen Optimierung zum Einsatz, um mögliches Rauschen in den Punkten sowie unnötige Krümmungsänderungen zu beseitigen. Dabei wird jedes Straßensegment einzeln optimiert, die Kreuzungsmittelpunkte werden konstant gehalten.

Spline-Fit Mithilfe der geglätteten Punkte wird pro Punktsequenz ein kubischer Spline als geometrische Beschreibung erzeugt.

Multi-Spline Um eine durchgängige geometrische Beschreibung zu erhalten, werden alle Splines zu einer gemeinsamen Beschreibung vereint.

Sampling Als abschließender Schritt in den globalen Koordinaten wird anhand der Splines eine komplette globale Punktsequenz $\mathcal{P}_{\text{routeHypothesis}}$ mit 1 m Punktabstand erzeugt.

Die nachfolgenden Schritte erfolgen zyklisch, maximal mit der Frequenz wie die globale Lokalisierung:

Lokalisierung Nach der Bestimmung des zum Ego-Fahrzeug nächstgelegenen Punktes auf $\mathcal{P}_{\text{routeHypothesis}}$ wird ein Ego-zentrierter Ausschnitt erzeugt, mit einer Gesamtlänge von 200 m.

Transformation Diese kurze Punktsequenz wird in das Ego-Koordinatensystem \mathbf{F}_{veh} transformiert und für weitere Anwendungen als ${}^{\text{veh}}\mathcal{P}_{\text{routeHypothesis}}$ bereitgestellt.

Für die Kreuzungen entfällt der einmalige Schritt, es erfolgen nur zyklisch:

Lokalisierung Aus der Sequenz von Kreuzungen entlang der Route $\mathcal{C}_{\text{route}}$ wird die nächste bzw. relevante Kreuzung $\mathcal{C}_{\text{next}}$ ausgewählt.

Transformation Neben der Transformation des Kreuzungsmittelpunkts von $\mathcal{C}_{\text{next}}$ in das Ego-Koordinatensystem \mathbf{F}_{veh} werden die Winkel der Kreuzungsäste rotiert.

B.3.3.3 Hypothesenverwendung

Die lokale Punktsequenz ${}^{\text{veh}}\mathcal{P}_{\text{routeHypothesis}}$ und die nächste Kreuzung $\mathcal{C}_{\text{next}}$ dienen in der hindernisvermeidenden Navigation mit den Tentakeln zur Integration von globalen Information (vgl. Abschnitt 7.3.6). In anderen Wahrnehmungsmodulen werden diese Hypothesen optional zur Stützung der Wahrnehmung verwendet, z. B. „Rechtskrümmung“ als Hinweis für die visuelle Wegerkennung.

Für andere Zwecke sind die Hypothesen zwingend erforderlich, z. B. für die visuelle Erkennung und das Tracking von Kreuzungen (siehe Abschnitt 7.3.5.1). Die Kreuzungserkennung, die als hochdimensionales und damit rechenintensives Partikelfilter aufgebaut ist, ist nur im Bereich von Kreuzungshypothesen aktiv, außerhalb verbraucht sie keine Rechenleistung. Durch die Hypothesen kann der Suchraum in der visuellen Kreuzungserkennung eingeschränkt werden, einerseits durch eine Geometriehypothese (Anzahl der Äste, Winkel zueinander) und andererseits durch eine Positionshypothese des Kreuzungsmittelpunktes. Für die Navigation ist eine präzise Schätzung der ersten beiden Äste wichtig, alle weiteren dienen einem stabileren Tracking der Kreuzung.

B.3.4 GNSS- & kartenbasierte Spur- und Kreuzungsgenerierung als Rückfallebene

Auf Basis der lokalen Punktsequenz ${}^{\text{veh}}\mathcal{P}_{\text{routeHypothesis}}$ sowie der nächsten Kreuzung C_{next} können zusätzlich Rückfallebenen für die autonome Fahrt realisiert werden. Abschnitt 7.3.4.3 geht auf die GNSS- & kartenbasierte Spurgenerierung ein, während die GNSS- & kartenbasierte Kreuzungsgenerierung in Abschnitt 7.3.5.2 vorgestellt wird.

Eine Voraussetzung für die Verwendung als Rückfallebene sind einerseits die passenden Annotation aus der Routenplanung (nicht „Map-Critical“, nicht „GNSS-Critical“) und andererseits eine präzise globale Lokalisierung mittels GNSS oder anderen Verfahren.

B.3.5 Visualisierung

Im in dieser Arbeit genutzten zentralen Visualisierungs- und Missionsplanungsmodul MuCAR-Central kommen Orthophotos für die Visualisierung der Routenplanung und bei der Überwachung des Missionsfortschritts zum Einsatz. Sie werden allerdings nicht in den Softwaremodulen für die Autonomie an sich verwendet. Den Orthophotos wird ein Ausschnitt einer digitalen topographischen Vektorkarte überlagert.

In derselben GUI (Grafische Benutzerschnittstelle, engl. *Graphical User Interface*) werden auch die erkannten Objekten, Fahrspuren und Kreuzungen unter Verwendung der Lokalisierung des Ego-Fahrzeugs visualisiert – daraus ergibt sich ein wichtiges Werkzeug für den Operator, um eventuelle Fehlfunktionen frühzeitig zu erkennen. So wirken sich bspw. Lokalisierungsfehler dergestalt aus, dass das Ego-Fahrzeug „virtuell“ durch die Wiese fährt – statt wie in der Realität auf dem Weg.

C Auswertung von Missionen

In diesem Abschnitt wird die Eignung des in Kapitel 7 entwickelten hierarchischen Konzepts aus Missionen, in denen unterschiedlichen Aufgabentypen, *Verhaltensweisen* sowie *Wahrnehmungs- und Navigationsmodule* verknüpft werden, in realen Szenarien untersucht.

Der erste Einsatz erfolgte bei der M-ELROB (Military European Land Robot Trial) 2010 in Hammelburg in den Szenarien *Transport-Convoy* und *Transport-Mule* – letzteres wird in Anhang C.1 genauer betrachtet. Für eine Projektabschlussdemonstration wurde eine komplexere Mission entworfen, in der eine größere Vielzahl von Aufgabentypen zum Einsatz kam. Diese wird in Anhang C.2 vorgestellt.

Zum Ranking bei der European Land Robot Trial (ELROB) sei angemerkt, dass in einigen frühen Veranstaltungen alle Fahrzeuge mit einem Sicherheitsfahrer von Beginn an außerhalb der Wertung geführt wurden – der Fokus lag auf vollkommen unbemannten Fahrzeugen. Da eine wachsende Anzahl an Teams mit Personenkraftwagen (PKWs) oder Lastkraftwagen (LKWs) teilnimmt und aus Zulassungs- und Sicherheitsgründen nicht auf einen Sicherheitsfahrer verzichten kann, wurde eine Kategorie *Supervised autonomy* eingeführt. Hier werden alle Eingriffe von einem mitfahrenden Schiedsrichter festgehalten.

C.1 M-ELROB 2010 – Transport Mule-Szenario

Die Aufgabe im *Transport-Mule*-Szenario der M-ELROB besteht darin, während einer Wettbewerbsdauer von 60 Minuten möglichst oft mit einem Roboter bzw. Fahrzeug zwischen zwei definierten Punkten bzw. Camps hin und her zu pendeln. Die Idee ist, zwischen diesen zwei Camps eine Art Shuttle-Service zu betreiben, der den automatisierten Transport von Gütern übernehmen kann. In der M-ELROB 2010 auf dem Truppenübungsplatz Hammelburg wurde als zusätzliche Herausforderung der sog. *Approach* integriert: alle Roboter wurden zu Beginn an einer unbekanntem Stelle ausgesetzt, von der sie zum ersten Camp navigieren mussten.

Die Teams und ihre Fahrzeuge kennen das Gebiet vor dem Start nicht und erhalten vom Veranstalter keine präzisen Karten – die Nutzung von öffentlich frei verfügbaren oder kommerziell erhältlichen Karten ist jedoch zulässig. Kurz vor dem Start werden die erforderlichen Wegpunkte bereitgestellt, die zusammen mit weiteren Informationen zu den Anforderungen im Szenario dazu dienen, die Aufgaben in einer Missionsbeschreibung verknüpfen zu können.

Mule Approach Das Fahrzeug wird vom Veranstalter zu P0 befördert. Nach dem Start der Software durch den Sicherheitsfahrer, der gleichzeitig der Operator ist, bestimmt das Fahrzeug seine Position P0 mithilfe von Globales Navigationssystem, engl. *Global Navigation Satellite System* (GNSS). Von P0 plant es unter Verwendung von Kartendaten die kürzeste Route zum ersten

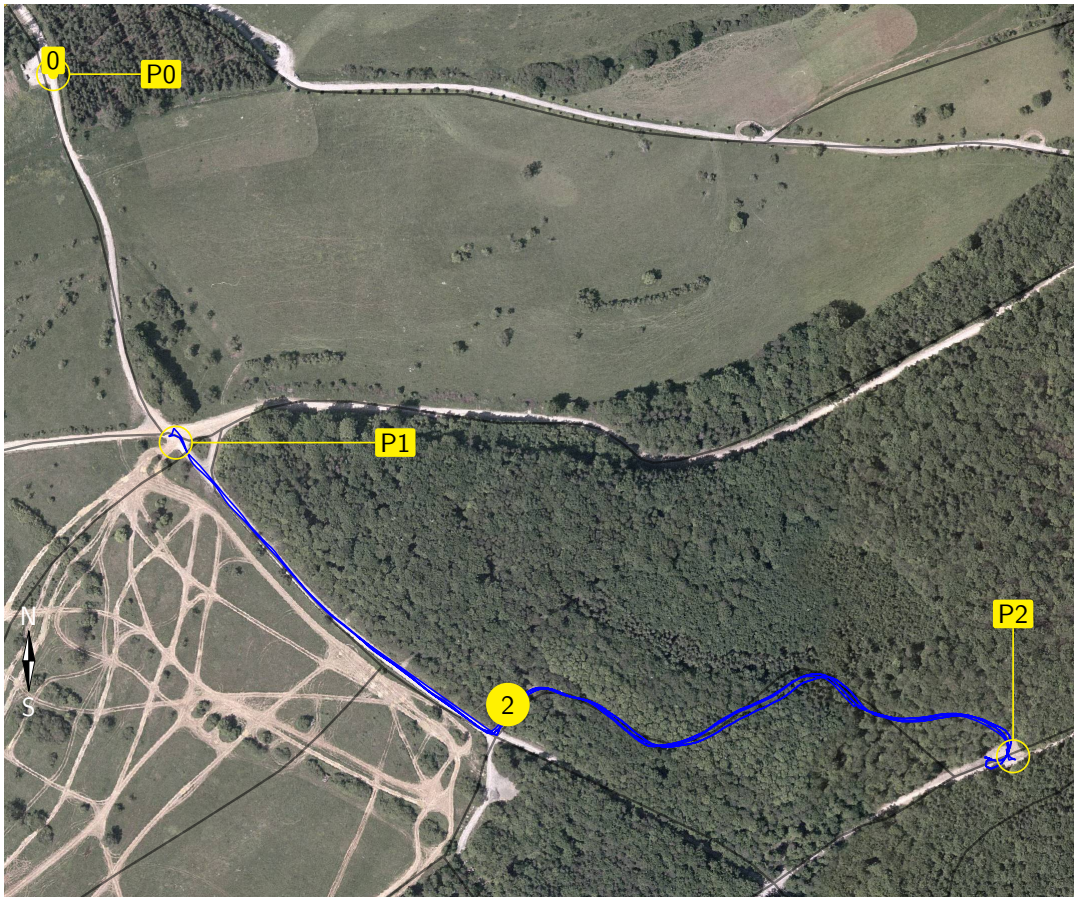


Abbildung C.1:

Darstellung des „Transport Mule“-Szenarios der M-ELROB 2010 in Hammelburg.

Camp P1. In unserer Missionsbeschreibung kommt hierfür der Aufgabentyp *Hindernisvermeidende Navigation* zum Einsatz.

Mule Teach-In Nach dem Erreichen von P1 stößt als weitere Person der *Teacher* hinzu, der die geforderte Ladung ins Fahrzeug lädt – ins diesem Fall einige Sandsäcke. Dem Fahrzeug ist die Zielkoordinate von Camp P2 anfangs nicht bekannt, daher wird die erste Fahrt von P1 zu P2 durch den Teacher angeleitet. Dieser läuft vor dem Fahrzeug entlang und „zeigt ihm den Weg“, während das Fahrzeug folgt und den Weg kartiert. Am Zielpunkt signalisiert der Teacher dem Fahrzeug, dass das Camp P2 erreicht ist, und beendet damit den Teach-In. In unserer Missionsbeschreibung kommt hierfür der Aufgabentyp *Mule Teach-In* mit dem Typ „Person“ für das Führungsobjekt zum Einsatz.

Mule Shuttle Der folgende Aufgabentyp *Mule Shuttle* sorgt dafür, dass das Fahrzeug mit wechselnden Zielpunkten P1 und P2 eine Route zwischen beiden Camps berechnet und so oft wie möglich pendelt.

Auswertung Abbildung C.1 gibt einen Überblick über die Wettbewerbsstrecke. Die folgenden Punkte basieren auf einer Auswertung der während des Szenarios

aufgezeichneten RTDB-Streams sowie den internen Aufzeichnungen aus dem INS vom Typ O_xTS RT3003 v1.

- Mule Approach:
 - Wegpunkt P1 erfolgreich auf Karte gematched und Route berechnet
 - Verwendung der *Tentakel* zur integrierten Wahrnehmung und Bewegungsplanung
 - Fahrtstrecke 327 m in 3:40 min zurückgelegt
 - 100 % autonom
- Mule Teach-In:
 - vorweg laufenden Menschen mit LiDAR-basiertem (Light Detection And Ranging) Tracker verfolgt (vgl.. Abschnitt 7.3.3.1)
 - Wegpunkte zwischen P1 und P2 aufgezeichnet
 - 100 % autonom
 - Beendigung nach $\Delta t_{\text{stoppingLongTime}}$
- Mule-Shuttle:
 - Verwendung der *Tentakel* zur integrierten Wahrnehmung und Bewegungsplanung
 - zweimal komplett entlang der aufgezeichneten Strecke hin- und zurück gependelt (P2 → P1 → P2 → P1 → P2), dann Abbruch aus Zeitgründen.
 - 4637 m gesamt in 48:25 min
 - 98,36 % der Strecke autonom
 - 97,06 % der Zeit autonom
 - Wenden an den Camps in 4 von 5 Fällen autonom bewältigt
 - teilweise Straßensperre autonom erkannt und lokal umfahren
 - Autonomes Zurücksetzen vor einem in die Fahrbahn hängenden Ast, um neue Fahroptionen zu erhalten.
 - Abbildung 7.10c zeigt exemplarisch eine Szene aus dem Wettbewerb.
- Manuelle Eingriffe:
 - Bei jeder der vier Passagen während des Mule-Shuttle wollte das Fahrzeug an der Kreuzung nahe Wegpunkt (2) nicht abbiegen sondern geradeaus fahren. Aufgrund des waldreichen Szenarios mit starken GNSS-Abschattungen wurde im Vorfeld die Gewichtung der Routenabweichung a_{target} sehr gering gewählt. Dadurch wählte das Fahrzeug den besser einsehbaren und daher besser bewerteten Weg geradeaus.

- Wende bei P2 einmal gescheitert, vermutlich falsche Lokalisierung auf die Route aufgrund starker GNSS-Abweichung.

In den offiziellen Ergebnissen¹ wurde das Team aufgrund des Sicherheitsfahrers nicht aufgeführt, liegt aber nach Zahlen (Eingriffszeit, Anzahl Handover, Gesamtzeit) vor dem Sieger. Ein Video² gibt zusätzlichen einen Eindruck über die gezeigte Performance sowie die Herausforderungen in diesem Szenario.

C.2 Projektabschlussdemonstration

In diesem Abschnitt wird eine umfangreiche Mission vorgestellt, die in der Abschlusspräsentation zur Studie E E810 9C006 9F148 „Robotsensor Umwelterkennung“ [WTD 81 - 320: Sensortechnologie und Robotik, 2009] für das Bundesamt für Wehrtechnik und Beschaffung (BWB) gezeigt und von einem autonomen Fahrzeug erfüllt wurde. Nach [Luettel et al., 2011c] wird diese als Abfolge von Aufgabentypen (A) unterschiedlicher Art definiert, die verschiedene Wahrnehmungs- und Navigationsmodule (W) benötigen:

- (A) Konvoi mit verschiedenen Führungsfahrzeugen
- (A) Navigation auf dem Straßennetz
- (W) Nutzung der Daten des „Kartenbasierten Fahrens“ als Hypothesen für die visuelle Fahrspurerkennung sowie Blickrichtungssteuerung
- (W) Nutzung eines sensorbasierten Fahrzeugtrackings (vgl. Abschnitt 7.3.3.3)
- (W) Integration der visuellen Fahrspur- und Kreuzungserkennung (vgl. Abschnitte 7.3.4.1 und 7.3.5.1),
 - Nutzung überall dort, wo es gemäß Karte sinnvoll und gemäß Umwelt möglich ist
- Nutzung diverser Fahrzeugfähigkeiten (Manöver, vgl. Abschnitt 6.4)
- Aktualisierung des Kartenwissens basierend auf den Ergebnissen der Wahrnehmungsexperten (Spurverlauf, Kreuzungen)

Die in Abbildung C.2 dargestellte Mission besteht aus mehreren Aufgabentypen:

0. In der ersten Aufgabe vom Typ *Konvoi* dient der VW Tiguan als Führungsfahrzeug, sie startet auf der Mehrzweckfläche und hat P0 im Norden der verbuschten Fläche zum Ziel. Beim Passieren von P0 erfolgt die Weiterschaltung in der Mission.

Verhaltensweisen: *Objektfolgen*.

Wahrnehmungs- und Navigationsmodule: *Visuelles modellbasiertes Fahrzeugtracking* (benötigt die im Folgenden nicht extra aufgeführten *Einfachen Rasterkarten*), *Fahrspurgenerierung anhand von Objekten*.

¹<https://www.elrob.org/elrob-2010-results>

²<https://www.youtube.com/watch?v=uBZRA6wf2Qc>

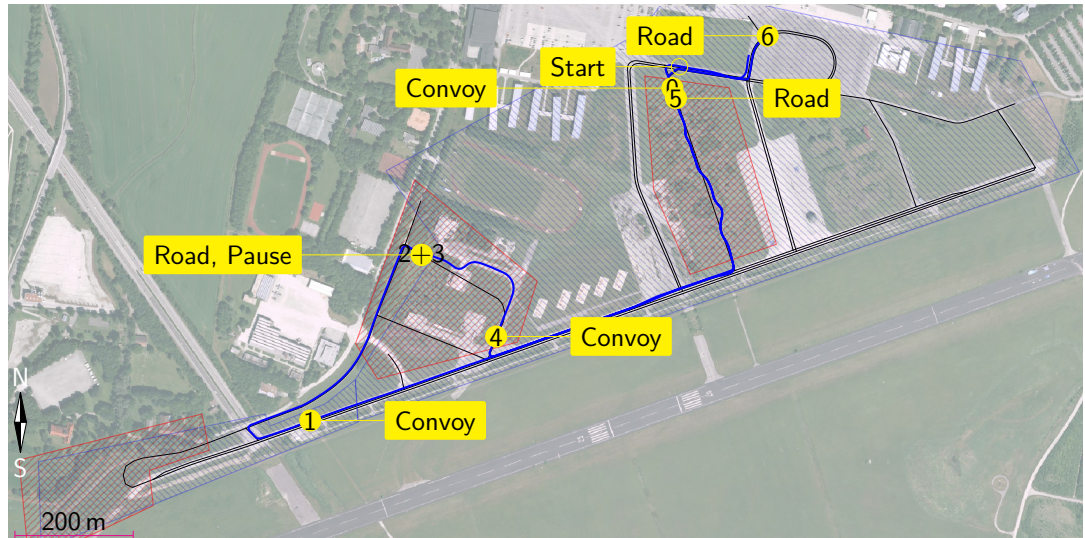


Abbildung C.2:

Szenariobeschreibung der Projektabschlussdemonstration: Die schwarzen Linien stellen das Straßen- und Wegenetz aus der Karte dar. *Rot schraffierte* Gebiete sind *Map Critical*, d. h. die Kartendaten sind von schlechter Qualität und sollen während der Mission verbessert werden. In *blau schraffierten* Gebieten ist der Einsatz von visueller Spurerkennung nicht möglich. Die gelben Punkte markieren die Zielpunkte der einzelnen Aufgaben. Die blaue Linie visualisiert den gefahrenen Track nach Abschluss der Mission.

1. Aufgrund gleichen Aufgabentyps geht es nahtlos weiter im *Konvoi* hinter dem Tiguan als Führungsfahrzeug. Dieser gibt nun einen geeigneten Weg durch die verbuschte Fläche vor, und biegt dann auf den Taxiway (lange Straße im Süden der Karte) nach Westen ab bis zum Passieren von P1. Währenddessen wird die im Bereich der Verbuschung schlechte und als *Map Critical* gekennzeichnete Karte (siehe Abb. C.4a) anhand des gefahrenen Weges aktualisiert.
Verhaltensweisen: *Objektfolgen*.
Wahrnehmungs- und Navigationsmodule: *Visuelles modellbasiertes Fahrzeugtracking, Fahrspurgenerierung anhand von Objekten*.
2. Während der Fahrt erfolgt der nahtlose Übergang in die *Navigation auf einem Straßennetz*, es wird eine Route zu P2 geplant.
Verhaltensweisen: *Spurfolgen, Abbiegen an Kreuzungen*.
Es kommen fünf Wahrnehmungs- und Navigationsmodule parallel zum Einsatz: *Visuelles modellbasiertes Fahrzeugtracking* liefert weiterhin die Position des Tiguans, zu dem mittels ACC der Sicherheitsabstand eingehalten wird.
Die Daten aus der visuellen *Erkennung und Tracking von Wegen* und der *Erkennung und Tracking von Kreuzungen* (beide benötigen als Grundlage *Akkumulierte Rasterkarten*) werden priorisiert, als Rückfallebene und zur Hypothesengenerierung dient die *GNSS- & kartenbasierte Spurgenerierung* und *GNSS- & kartenbasierte Kreuzungsgenerierung*. Aufgrund der Annotation in der Karte kann auf dem Taxiway keine visuelle Erkennung erfolgen, sondern erst nach zweimaligen Rechtsabbiegen auf den Feldweg (siehe Abb. C.3). Diesem wird visuell weiter nach Nordosten gefolgt, z. T. mit kurzfristiger kar-

tenbasierter Rückfallebene im Bereich der betonierten Fläche bei Tor 19. Die Kreuzungen werden visuell erkannt, nach dem automatisierten Abbiegen hält das Fahrzeug an P2.

3. Die nächste Aufgabe ist ein einminütiges *Pausieren*, bspw. für Aufklärungszwecke, währenddessen das Fahrzeug an P2/P3 verbleibt. Währenddessen positioniert sich ein Audi A4 vor Munich Cognitive Autonomous Robot Car 3rd Generation (MuCAR-3).

Verhaltensweisen: *Fahrstufenwechsel*.

Wahrnehmungs- und Navigationsmodule: keine.

4. Im Aufgabentyp *Konvoi* folgt MuCAR-3 nun dem Audi A4 als Führungsfahrzeug bis zu P4. Der zu befahrene Bereich ist in der Karte inkorrekt kartiert und daher als *Map Critical* markiert. Während des Objektfolgens wird der gefahrene Track aufgezeichnet (siehe Abb. C.4c) und die Karte damit nach dem Verlassen des *Map Critical*-Bereichs aktualisiert (siehe Abb. C.4d).

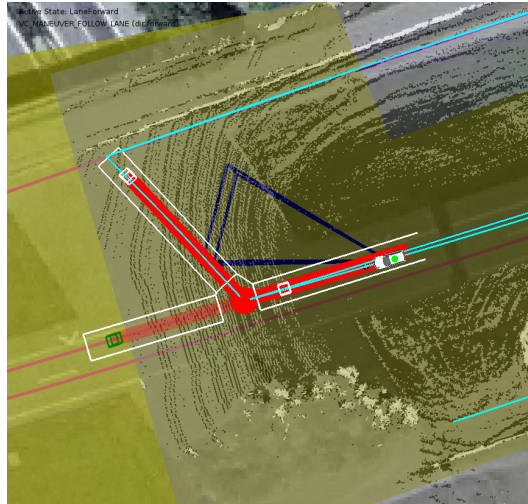
Verhaltensweisen: *Objektfolgen*.

Wahrnehmungs- und Navigationsmodule: *Visuelles modellbasiertes Fahrzeugtracking, Fahrspurgenerierung anhand von Objekten*.

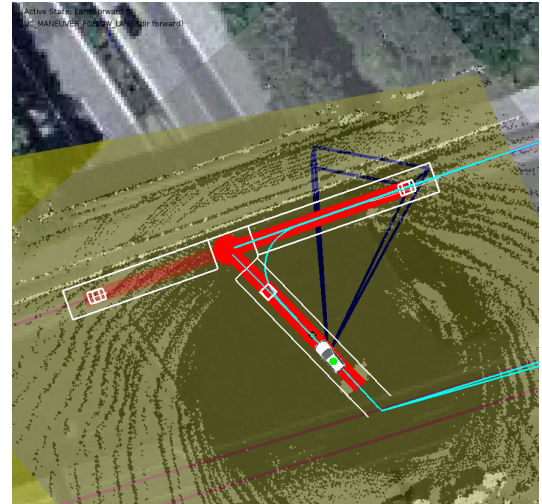
5. Beim Passieren von P4 erfolgt ein nahtloser Übergang in die *Navigation auf einem Straßennetz*, es wird eine Route zu P5 geplant. Dabei wird die zuvor neu kartierte Strecke im ehemaligen *Map Critical*-Bereich für die Routenplanung genutzt. Das Fahrzeug biegt auf Basis der kartenbasierten Rückfallebene links auf den Taxiway ab und folgt bis zur übernächsten Kreuzung, an der es links in den verbuschten Bereich abbiegt. Die während der Konvoi-Fahrt zu P1 aktualisierte Karte (siehe Abb. C.4b) ist präzise genug, um nun in der kartenbasierten Rückfallebene den Slalom um die Büsche zu ermöglichen. Verhaltensweisen: *Spurfolgen, Abbiegen an Kreuzungen*.

Wahrnehmungs- und Navigationsmodule: *Visuelles modellbasiertes Fahrzeugtracking, GNSS- & kartenbasierte Spurgenerierung, GNSS- & kartenbasierte Kreuzungsgenerierung*.

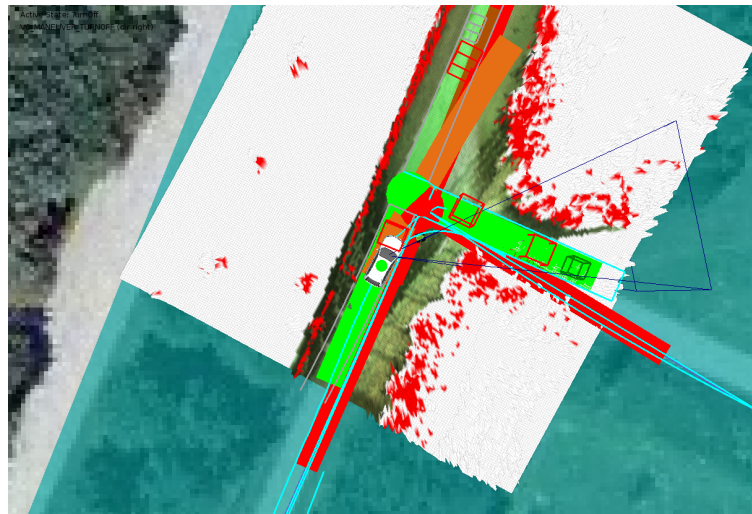
6. Nach dem Erreichen von P5 wird ein weiterer Routenabschnitt für die *Navigation auf einem Straßennetz* zum Zielpunkt P6 geplant. Es kommen dieselben Verhaltensweisen sowie Wahrnehmungs- und Navigationsmodule wie in Aufgabe 5 zum Einsatz.



(a) Nach Passieren von P1 fährt das Führungsfahrzeug geradeaus, MuCAR-3 folgt der aus der geplanten Route (cyan) mithilfe des GNSS generierten Fahrspur (rot) und biegt anhand des Kartenwissens (Kreuzung mit roten Ästen) rechts ab. Die Blickrichtungssteuerung (blauer Kegel) fokussiert abwechselnd die Kreuzungsäste.



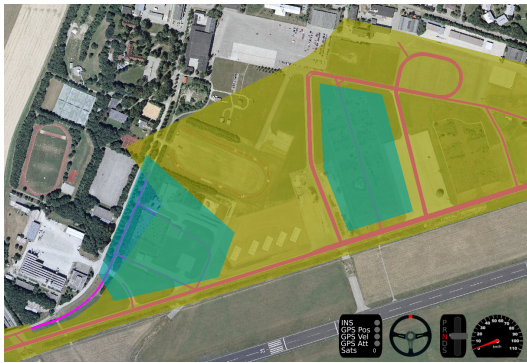
(b) Hier wurde bereits das Abbiegemanöver für die zweite Kreuzung aktiviert, zu erkennen an der Klothoidensequenz (cyan) zwischen ein- und ausgehendem Kreuzungsast. Der Fokus der Blickrichtungssteuerung (blauer Kegel) liegt permanent auf dem Ausgangsast.



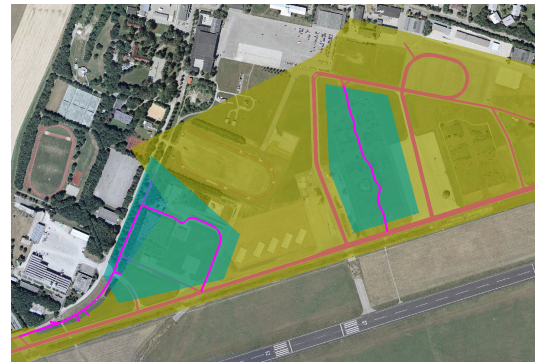
(c) Abbiegevorgang kurz vor P2 innerhalb des Map-Critical-Bereichs, daher dient das Kartenwissen (Kreuzung mit roten Ästen) nur als Hypothese für die visuelle Erkennung. Das Fahrzeug folgt dem in den Sensordaten getrackten Weg (orange) und hat die Kreuzung (grün) bereits erkannt. Durch die aktive Blickrichtungssteuerung konnte der Ausgangsast bereits beobachtet und damit Farbinformationen in der Rasterkarte akkumuliert werden.

Abbildung C.3:

Abbiegevorgänge während der Mission im Bereich zwischen P1 und P2. Solange die visuelle Erkennung nicht aktiv ist ((a) und (b)), ist im Hintergrund die höhenkodierte Rasterkarte (ocker) dargestellt. In (c) ist die visuelle Fahrspur- und Kreuzungserkennung aktiv, hier werden die Hindernis- und Farbebenen der akkumulierten Rasterkarte dargestellt.



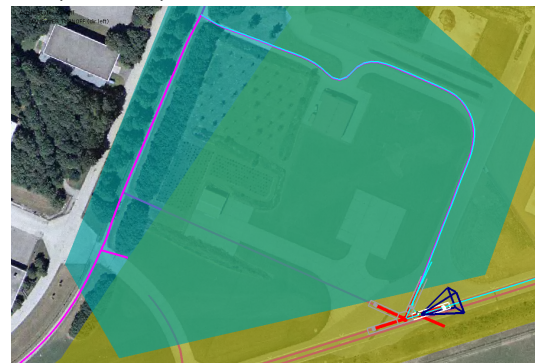
(a) Gesamtansicht des Straßennetzes (rot) im Ausgangszustand.



(b) Ursprüngliches Straßennetz (rot) und während der Mission verbesserte Kartensegmente (magenta).



(c) Während MuCAR-3 im Konvoi-Betrieb durch Map-Critical-Bereiche fährt, wird der gefahrene Weg im Dead-Reckoning-System ohne Nutzung von GNSS-Daten aufgezeichnet (cyan, global noch nicht korrekt lokalisiert).



(d) Nach dem Verlassen des Map-Critical-Bereichs werden die aufgezeichneten Segmente jeweils zwischen zwei Kreuzungen in die Karte eingepasst (magenta). Somit steht dort für spätere Fahrten ebenfalls die kartensbasierte Rückfallebene zur Verfügung.

Abbildung C.4:

Ablauf der Kartenaktualisierung während des Beispielmission. Im Gegensatz zu Abb. C.2 weicht die Farbkodierung ab: *grün hinterlegte* Gebiete sind *Map Critical*, in *gelb hinterlegten* Gebieten ist der Einsatz von visueller Spurerkennung nicht möglich.

Symbolverzeichnis

Abkürzungen

ABS	Anti-Blockier-System, engl. <i>Anti-lock Braking System</i>
ACC	Abstandsregeltempomat, engl. <i>Adaptive Cruise Control</i>
ADAS	Fahrerassistenzsystem, engl. <i>Advanced Driver Assistance System</i>
AGeoBw	Amt für Geoinformationswesen der Bundeswehr
AGV	Fahrerloses Transportfahrzeug, engl. <i>Automated Guided Vehicle</i>
AHS	Autonomes Transportsystem, engl. <i>Autonomous Haulage System</i>
AHVS	Automated Highway Vehicle System
API	Application Programming Interface
BAAINBw	Bundesamt für Ausrüstung, Informationstechnik und Nutzung der Bundeswehr
BAST	Bundesanstalt für Straßenwesen
BEV	Battery Electric Vehicle
BMWi	Bundesministerium für Wirtschaft und Energie
BWB	Bundesamt für Wehrtechnik und Beschaffung
C-ELROB	Civilian European Land Robot Trial
CAN	Controller Area Network
CBRNE	Chemisch, Biologisch, Radioaktiv, Nuklear und Explosiv
CMU	Carnegie Mellon University
CoG	Schwerpunkt, engl. <i>Center of Gravity</i>
CPU	(Haupt-)Prozessor eines Computers, engl. <i>Central processing unit</i>
CTRA	Konstante Gierrate und Beschleunigung, engl. <i>Constant Turn Rate and Acceleration</i>
CTRV	Konstante Gierrate und Geschwindigkeit, engl. <i>Constant Turn Rate and Velocity</i>
DARPA	Defense Advanced Research Projects Agency
DBD	Diehl BGT Defence GmbH & Co. KG
DbW	engl. <i>Drive-by-Wire</i>
DESM	Dynamisches Einspurmodell
DFG	Deutsche Forschungsgemeinschaft
DGM	Digitales Geländemodell
DHM	Digitales Höhenmodell
DOF	Freiheitsgrade, engl. <i>Degrees of Freedom</i>
DOM	Digitales Oberflächenmodell
DOP	Digitales Orthophoto
DOP	Dilution of Precision
DR	Koppelnavigation (engl. <i>Dead Reckoning</i>)
DSR	Fahrer-Lenkempfehlung, engl. <i>Driver Steering Recommendation</i>

Abkürzungen

dtec.bw	Zentrum für Digitalisierungs- und Technologieforschung der Bundeswehr
EGM 96	Earth Gravitational Model 1996
EKF	Erweitertes Kalman-Filter
ELROB	European Land Robot Trial
EME	Eigenbewegungsschätzung, engl. <i>Ego-Motion Estimation</i>
ENU	Osten-Norden-Oben, engl. <i>East-North-Up</i>
EOD	explosive ordnance disposal
ESP	Elektronisches Stabilitätsprogramm, engl. <i>Electronic Stability Control</i>
ETRS89	Europäisches Terrestrisches Referenzsystem 1989
F&T	Forschung und Technologie
FGPA	Field Programmable Gate Array
FOG	Faseroptischer Kreisel, engl. <i>Fibre-Optic Gyroscope</i>
FPFD	Vorwärts-Prädiktion mit Fusion und Dekorrelation, engl. <i>Forward-Prediction Fusion and Decorrelation</i>
FTS	Fahrerloses Transportsystem
GBAS	Ground Based Augmentation System
GIS	Geoinformationssystem
GLONASS	Globales Satellitennavigationssystem, russ. <i>Globalnaja nawigazionnaja sputnikowaja sistema</i>
GMPHD	engl. <i>Gaussian Mixture Probability Hypothesis Density</i>
GNSS	Globales Navigationssatellitensystem, engl. <i>Global Navigation Satellite System</i>
GPS	Global Positioning System
GPU	Grafikprozessor, engl. <i>Graphics processing unit</i>
GRS 80	Geodätisches Referenzsystem 1980
GUI	Grafische Benutzerschnittstelle, engl. <i>Graphical User Interface</i>
HaaS	engl. <i>Hardware as a Service</i>
HFOV	Horizontales Sichtfeld, engl. <i>Horizontal Field of View</i>
HSM	Hierarchischer Zustandsautomat, engl. <i>Hierarchical State Machine</i>
HT	Hyper-Threading
HTM	Homogene Transformationsmatrix
IAC	Indy Autonomous Challenge
IEEE	Institute of Electrical and Electronics Engineers
IITB	Fraunhofer-Institut für Informations- und Datenverarbeitung
IMU	Inertiale Messeinheit, engl. <i>Inertial Measurement Unit</i>
iMUGS	integrated Modular Unmanned Ground System
INS	Inertiales Navigationssystem, engl. <i>Inertial Navigation System</i>
InterRoC	Interoperable Robotic Convoy
ION	The Institute of Navigation
IPC	Interprozesskommunikation, engl. <i>Inter-Process Communication</i>
ISF	Institut für Systemdynamik und Flugmechanik
ITRF	International Terrestrial Reference Frame

KESM	Kinematisches Einspurmodell
KF	Kalman-Filter
KISS	Halte es einfach, Du Dummkopf!, engl. <i>Keep it simple, stupid!</i>
KIT	Karlsruher Institut für Technologie
KML	Keyhole Markup Language
KogMo-RTDB	Echtzeitdatenbank für kognitive Automobile, engl. <i>KogniMobil Real Time Data Base</i>
KogniMobil	Sonderforschungsbereich Transregio 28 – „Kognitive Automobile“
LCA	Spurwechselassistent, engl. <i>Lane Change Assist</i>
LDWS	Spurverlassenswarner, engl. <i>Lane Departure Warning System</i>
LHD	Fahrlader, engl. <i>Load Haul Dump</i>
LiDAR	Light Detection And Ranging
LKAS	Spurhalteassistenzsystem, engl. <i>Lane Keeping Assist System</i>
LKW	Lastkraftwagen
LWS	Lenkwinkelsensor
M-ELROB	Military European Land Robot Trial
MaaS	engl. <i>Mobility as a Service</i>
MarVEye-8	Multifokales aktives / reaktives Fahrzeug-Auge der 8. Generation, engl. <i>Multifocal active / reactive Vehicle Eye 8th Generation</i>
MDF	Mission Data File
MEMS	Mikrosystem, engl. <i>Micro-Electro-Mechanical System</i>
MGCS	Main Ground Combat System
MHEV	Mild Hybrid Electric Vehicle
MKF	Militärkraftfahrer
MORE	Munich Mobility Research Campus
MuCAR	Munich Cognitive Autonomous Robot Car
MuCAR-3	Munich Cognitive Autonomous Robot Car 3 rd Generation
MuCAR-4	Munich Cognitive Autonomous Robot Car 4 th Generation
MuCAR-5	Munich Cognitive Autonomous Robot Car 5 th Generation
NED	Norden-Osten-Unten, engl. <i>North-East-Down</i>
NeFiMMore	Near Field Monitoring System MORE
NIR	Nahes Infrarot, engl. <i>Near-Infrared</i>
NMEA	National Marine Electronics Association
NQE	National Qualification Event
NTP	Network Time Protocol
NTRIP	<i>Networked Transport of RTCM via Internet Protocol</i>
NWU	Norden-Westen-Oben, engl. <i>North-West-Up</i>
o.B.d.A.	ohne Beschränkung der Allgemeinheit
OBDF	Objektbasierte Datenfusion, engl. <i>Object Based Data Fusion</i>
OEM	Erstausrüster, engl. <i>Original Equipment Manufacturer</i>
OOSM	Verzögerte Messung, engl. <i>Out-of-Sequence Measurement</i>
OSM	OpenStreetMap
OSRF	Open Source Robotics Foundation
OxTS	Oxford Technical Solutions
PA	Parklenkassistent, engl. <i>Park Assist</i>

Abkürzungen

PARAVAN	PARAVAN GmbH
PATH	Partners for Advanced Transit and Highways
PDC	Parkdistanzkontrolle, engl. <i>Park Distance Control</i>
PDOP	Positional DOP
PF	Partikel-Filter
PKW	Personenkraftwagen
PLL	Phasenregelschleife, engl. <i>Phase-Locked Loop</i>
PPS	Puls pro Sekunde
PROMETHEUS	PROgraMme for a European Traffic of Highest Efficiency and Unprecedented Safety
PTP	Precision Time Protocol
PVS	Personal Vehicle System
PWM	Pulsweitenmodulation, engl. <i>Pulse Width Modulation</i>
Radar	Radio Detection And Ranging
RGB	Rot-Grün-Blau
RKL	Rundumkennleuchte
RLS	Rheinmetall Landsysteme GmbH
RMMV	Rheinmetall MAN Military Vehicles GmbH
RNDF	Road Network Definition File
ROS	Robot Operating System
RTCM	<i>Radio Technical Commission for Maritime Services</i>
RTK	Echtzeitkinematik, engl. <i>Real Time Kinematic</i>
S-MET	Small Multipurpose Equipment Transport
SAE	Verband der Automobilingenieure, engl. <i>Society of Automotive Engineers</i>
SAE	SAE International
SARTRE	Safe Road Trains for the Environment
SDK	Softwareentwicklungssystem, engl. <i>Software Development Kit</i>
SFB	Sonderforschungsbereich
SLAM	Simultaneous Localization and Mapping
SPKF	Sigma-Punkt Kalman-Filter
StÜbPI	Standortübungsplatz
StrAsRob	Straßentransport mit Assistenzfunktionen von Robotern
StS	StreetScooter GmbH
SUV	Geländelimousine, engl. <i>Sport Utility Vehicle</i>
TARDEC	Tank Automotive Research, Development and Engineering Center
TAS	Institut für Technik Autonomer Systeme
TCO	Total Cost of Ownership
TCP	Transmission Control Protocol
TULF	Technologieträger Unbemanntes Landfahrzeug
UDP	User Datagram Protocol
UKF	Unscented Kalman-Filter
UniBw M	Universität der Bundeswehr München
USB	Universal Serial Bus
UTM	Universal Transverse Mercator

UWB	Ultrabreitband, engl. <i>Ultra Wide Band</i>
V2V	Fahrzeug-zu-Fahrzeug, engl. <i>Vehicle-to-Vehicle</i>
VaMoRs	Versuchsfahrzeug für autonome Mobilität und Rechnersehen
VaMP	VaMoRs-PKW
VBK 25	Vektorbasiskarte 25
VFOV	Vertikales Sichtfeld, engl. <i>Vertical Field of View</i>
VIAC	Artificial Vision and Intelligent Systems Laboratory (VisLab) Intercontinental Autonomous Challenge
VisLab	Artificial Vision and Intelligent Systems Laboratory
VITA	Vision Information Technology Application
VLAN	Virtuelles lokales Netzwerk, engl. <i>Virtual Local Area Network</i>
WGS 84	World Geodetic System 1984
WLAN	Drahtloses lokales Netzwerk, engl. <i>Wireless Local Area Network</i>
XAI	Erklärbare Künstliche Intelligenz, engl. <i>Explainable artificial intelligence</i>
ZGeoBw	Zentrum für Geoinformationswesen der Bundeswehr
ZKfWBw	Zentrum Kraftfahrwesen der Bundeswehr

Konventionen

Basisnotation

$(\cdot)_{[i]}$	ite Komponente
$(\cdot)^T$	Transponiert
\mathbf{M}	Matrix
s	Skalar
\mathbf{v}	Vektor
$(\dot{\cdot}), (\ddot{\cdot})$	gleichbedeutend mit $\frac{d}{dt}(\cdot)$ und $\frac{d^2}{dt^2}(\cdot)$

Kalibrierung

f	Brennweite
-----	------------

Koordinaten, Transformationen

Θ	Pitchwinkel
Φ	Rollwinkel
β	Schwimmwinkel
α	Schräglaufwinkel
Ψ	Gierwinkel
\mathbf{F}	Kartesisches Koordinatensystem
\mathbf{Y}	Ellipsoidisches Koordinatensystem
\mathbf{H}	Homogene Transformationsmatrix
\mathbf{P}	6-DOF-Pose
\mathbf{t}	Vektor der 3 translatorischen Freiheitsgrade

Lokalisierung und Kartierung

h	Ellipsoidische Höhe / Geografische Höhe
φ	Geographische Breite

Konventionen

λ	Geographische Länge
H	Orthometrische Höhe
N	Geoidhöhe, Geoid-Undulation
\mathcal{C}	Menge an Kreuzungen (entlang einer Route aus einer Mission)
\mathcal{P}	Route (aus einer Mission)

Mathematische Definitionen

σ	Standardabweichung
$ \cdot $	Betrag
\mathbb{R}	Menge der Reellen Zahlen
\mathbf{p}	3D-Punkt $\mathbf{p} = [x, y, z]$

Physikalische Größen

$\dot{\omega}$	Rotationsbeschleunigung in (rad/s ²)
ω	Rotationsgeschwindigkeit in (rad/s)
f	Frequenz in (Hz)
M	Drehmoment in (N m)
a	Beschleunigung in (m/s ²)
d	Abstand in (m)
F	Kraft
i	Getriebeübersetzung
c	Seitenkraftbeiwert
l	Länge in (m)
m	Masse
J	Massenträgheitsmoment
MP	Momentanpol
r	Radius eines Kreises / Zylindrischer Abstand
v	Geschwindigkeit in (m/s)
w	Breite in (m)

Zustandsraumdarstellung

\mathbf{D}	System Durchgangsmatrix
\mathbf{B}	Eingangsmatrix, zeitdiskret
\mathbf{D}	Störgrößeneingangsmatrix
\mathbf{u}	Systemeingang
$g(\cdot)$	Messgleichung, nichtlinear
\mathbf{w}	Messrauschvektor
\mathbf{y}	Messvektor
\mathbf{C}	Ausgangsmatrix, zeitdiskret
\mathbf{x}	Zustandsvektor
$f(\cdot)$	Systemgleichungen, nichtlinear
\mathbf{v}	Systemrauschvektor, zeitdiskret
Φ	Transitionsmatrix oder Zustandsübergangsmatrix, zeitdiskret
\mathbf{F}	Systemmatrix, zeitkontinuierlich

Zeit

k	Diskreter Zeitpunkt
Δt	Zeitdifferenz zwischen zwei Zeitpunkten: $\Delta t = t_1 - t_0$
t	Zeitpunkt

Steuerung und Regelung

c_0	Krümmung
c_1	lineare Krümmungsänderung
d	Ablage
$\Delta\Psi$	Kurswinkelfehler
Fahrzeug und Fahrerassistenz	
λ	Lenkwinkel
$\dot{\lambda}$	Lenkrate
l_f	Achsabstand
l_h	Abstand Schwerpunkt zur Hinterachse
l_v	Abstand Schwerpunkt zur Vorderachse
Bayes'sche Filter	
S	Innovationskovarianzmatrix
K	Kalman-Verstärkungsmatrix
R	Kovarianzmatrix des Messrauschens
q	Parametervektor
Q	Kovarianzmatrix des Prozessrauschens
P	Kovarianzmatrix des Schätzfehlers
$\hat{(\cdot)}$	Geschätzte Werte
$(\cdot)^*$	Prädizierte Werte

Indizes

Koordinaten, Transformationen

0	ebenes Koordinatensystem
\lrcorner	Koordinatensystem in der linken oberen Ecke eines Bildes
CoGg	Koordinatensystem im Fahrzeugschwerpunkt, in die Fahrbahnebene projiziert
CoG	Koordinatensystem im Fahrzeugschwerpunkt
DR	Koppelnavigations-Koordinatensystem (engl. <i>Dead Reckoning coordinate system</i>), Lage abhängig vom verwendeten Prozessmodell
ECEF	Earth Centered, Earth Fixed
EME	Eigenbewegungskoordinatensystem, engl. <i>Egomotion Estimation coordinate system</i> , Lage abhängig vom verwendeten Prozessmodell
LLH	Breitengrad φ , Längengrad λ , Höhe h , engl. <i>Latitude, Longitude, Height</i>
Sensor	Sensorreferenzpunkt, nicht weiter spezifiziert
hg	Hinterachs-Koordinatensystem, projiziert in die Reifenaufstandsebene unterhalb des Hinterachsmittelpunktes
h	Hinterachs-Koordinatensystem
inertial	inertiales Koordinatensystem
nav	Navigationskoordinaten

objC	Koordinatensystem im Zentrum der Bounding-Box eines (getrackten) Objekts
veh _g	Koordinatensystem des Ego-Fahrzeugs, projiziert in die Reifenaufstandsebene unterhalb des Fahrzeugreferenzpunkts
veh	Koordinatensystem im Fahrzeugreferenzpunkt des Ego-Fahrzeugs
v _g	Vorderachs-Koordinatensystem, projiziert in die Reifenaufstandsebene unterhalb des Vorderachsmittelpunktes
(·) _g	Koordinatensystem in der Fahrbahnebene unterhalb von (·)
(·)	Beliebiges Koordinatensystem
DESM	Dynamisches Einspurmodell (DESM)
KESM	Kinematisches Einspurmodell (KESM)
Bayes'sche Filter	
†	Offset
‡	Skalierungsfaktor

Definitionen

Mathematische Definitionen

L2-Norm $\|\mathbf{v}\| \triangleq \sqrt{\sum_{i=1}^n \mathbf{v}_{[i]}^2}$, wobei $\mathbf{v} \in \mathbb{R}^n$

Glossar

Beidou

Chinesisches GNSS zur Positionsbestimmung. Beidou, im internationalen Gebrauch auch als Compass bezeichnet, ist seit 2011 in Betrieb und war anfangs nur in großen Teilen Asiens und des Pazifikraums nutzbar. Seit 2020 ist das System mit 35 aktiven Satelliten zur weltweiten Nutzung freigegeben.

Digitales Geländemodell

Höhenmodell, welches die Erdoberfläche (Grenzschicht Pedosphäre/Boden zu Atmosphäre) ohne Vegetation und Bauwerke repräsentiert.

Digitales Höhenmodell

Oberbegriff für Digitales Geländemodell und Digitales Oberflächenmodell.

Digitales Oberflächenmodell

Höhenmodell, welches die Erdoberfläche (Grenzschicht Pedosphäre/Boden zu Atmosphäre) inklusive der darauf befindlichen Objekte (Vegetation, Bauwerke, Straßen, Gewässer usw.) repräsentiert.

Dilution of Precision

DOP ist ein skalare Größe, durch die der Einfluß der Satellitenkonstellation auf die Genauigkeit der Positionsbestimmung mittels GNSS beschrieben wird. Von den verschiedenen DOP-Werten wird vor allem der PDOP (Positional DOP) verwendet, um die Genauigkeit einer dreidimensionalen Positionsbestimmung zu beschreiben. Durch Multiplikation der Standardabweichung einer Pseudoentfernungsmessung mit dem PDOP-Wert ergibt sich die Standardabweichung der aus GNSS-Messungen abgeleiteten 3D-Position. Je kleiner PDOP, umso genauer ist das Positionsergebnis. Der Wert von 1 beschreibt die bestmögliche geometrische Anordnung der Satelliten, ein Wert von ungefähr 6 ist noch brauchbar. DOP-Werte größer als 10 lassen keine Auswertung mehr zu.

Echtzeitkinematik, engl. Real Time Kinematic

RTK ist eine GBAS-Georeferenzierungstechnik, welche die Genauigkeit von GNSS-Daten durch Erkennung der Phase der Trägerwelle eines Signals erhöhen soll und dabei Echtzeit-Korrekturen aus einer meist ortsfesten Referenz-/Basisstation verwendet. Die Korrekturdaten werden über eine separate Datenverbindung von der Referenzstation an die mobilen Empfänger übertragen.

Galileo

Europäisches GNSS zur Positionsbestimmung. Galileo ist seit Ende 2016 nutzbar. Seit Ende 2021 sind 30 Satelliten im Orbit, jedoch sind aus unterschiedlichen Gründen nicht alle nutzbar. Ab 2024 soll eine neue Satellitengeneration das System verstärken.

Geoid

Das Geoid ist eine wichtige Bezugsfläche im Schwerfeld der Erde und dient zur Definition von Höhen sowie zur Vermessung und Beschreibung der Erdfigur. Als gute Approximation des Geoids kann der mittlere Meeresspiegel der Weltmeere betrachtet werden – außerhalb der Landmassen ist das Geoid direkt in seiner Form sichtbar. Die Höhenabweichung zwischen dem Geoid und dem einfacheren Erdmodell Ellipsoid wird als Undulation N bezeichnet.

Global Positioning System

US-amerikanisches Globales Navigations satellitensystem, engl. *Global Navigation Satellite System* (GNSS) zur Positionsbestimmung. Die offizielle Bezeichnung lautet *Navigational Satellite Timing and Ranging GPS* (NAVSTAR GPS).

Orthophoto

Georeferenziertes Luftbild, auf eine orthogonale Projektion umgerechnet.

euRathlon

Serie an Wettbewerben für autonome Fahrzeuge in den Domänen Land, Luft und Unterwasser. Diese wurden in den Jahren 2013 bis 2015 ausgetragen und von der Europäischen Union mitfinanziert.

Grand Challenge

Wettbewerb für autonome Fahrzeuge, wurde in den Jahren 2004 und 2005 durch die Defense Advanced Research Projects Agency (DARPA) veranstaltet.

Robotics Challenge

Ein von 2012 bis 2015 durch die DARPA veranstalteter internationaler Robotikwettbewerb zur Förderung und Entwicklung von Technologien für Rettungseinsätze.

Stanford Racing Team

Das Stanford Racing Team ist ein von der Stanford University, dem Volkswagen Electronics Research Laboratory (ERL) und anderen Partnern für die Teilnahme an der Grand Challenge gegründetes Team.

Stanley

Fahrzeug des Stanford Racing Team, das für die Teilnahme an der Grand Challenge 2005 aufgebaut wurde.

Subterranean Challenge

Ein von 2018 bis 2019 durch die DARPA ausgerichteteter Wettbewerb für Erkundungs- und Rettungsroboter in diversen Untergrundszenarien, wie Tunneln, Höhlen oder Bahnstationen.

Urban Challenge

Wettbewerb für autonome Fahrzeuge, wurde im Jahr 2007 durch die DARPA veranstaltet.

Ethernet

Eine Technologie, die Software (Protokolle usw.) und Hardware (Kabel, Verteiler, Netzwerkkarten usw.) für kabelgebundene Datennetze spezifiziert, die üblicherweise für lokale Datennetze verwendet wird. Häufig genutzte Protokolle sind TCP und UDP, weiterhin gibt echtzeitfähige Ethernet-Derivate.

FireWire

FireWire, i.LINK oder IEEE1394 ist ein Bus für serielle Datenübertragung. Es gibt zwei Versionen: IEEE1394a mit bis zu 400 Mbit/s und IEEE1394b mit bis zu 800 Mbit/s.

GigE

Gigabit Ethernet

GigE Vision

Kamera-Schnittstelle, basierend auf GigE

Matlab

Matlab ist eine kommerzielle Software des US-amerikanischen Unternehmens MathWorks zur Lösung mathematischer Probleme und zur grafischen Darstellung

der Ergebnisse. Matlab ist vor allem für numerische Berechnungen mithilfe von Matrizen ausgelegt, woher sich auch der Name ableitet: MATrix LABoratory.

Middleware

Anwendungsneutrale Software, die der Vermittlung und dem Datenaustausch anderer Anwendungen dient.

MuCAR-Central

Bei TAS entwickelte Grafische Benutzerschnittstelle, engl. *Graphical User Interface* (GUI), die zur Visualisierung des Fahrzeugs und seiner Umgebung sowie zur Missionsplanung verwendet wird.

Network Time Protocol

ein Standard zur Synchronisierung von Uhren in Computersystemen über paketbasierte Kommunikationsnetze

Precision Time Protocol

ein Standard zur Synchronisierung von Uhren in Computersystemen und Sensoren über paketbasierte Kommunikationsnetze mit hoher Präzision im zig-Nanosekunden-Bereich

tas::vehicle

Am Institut für Technik Autonomer Systeme (TAS) entwickelte Softwarebibliothek zur Abstraktion eines Fahrzeuges. Bietet Informationen über Fahrzeugparameter (Geometrie, Dynamik, Kommunikation), eine Übersicht über integrierte Sensoren, Schnittstellen für Mess- und Stellwerte, sowie Prozess- und Messmodelle für Bewegung und Sensorik.

HX58

Geländegängiger militärischer 6x6 LKW mit einer Nutzlast von 11 t.

Servotronic

Geschwindigkeitsabhängige Steuerung der Servolenkung: Bei geringer Geschwindigkeit, z. B. beim Rangieren, wird die maximale Lenkunterstützung angeboten. Bei hoher Geschwindigkeit, z. B. auf der Autobahn, wird die Lenkunterstützung minimiert.

dSPACE AutoBox

Echtzeitsystem für schnelles Funktions-Prototyping der Firma dSPACE

dSPACE MicroAutoBox

Echtzeitsystem für schnelles Funktions-Prototyping der Firma dSPACE

Hyper-Threading

Hyper-Threading ist eine spezielle Implementierung von hardwareseitigem Multithreading in Intel-Prozessoren, die auch von AMD übernommen wurde.

PARAVAN SpaceDrive

Ein nachrüstbares, straßenzugelassenes DbW-System (engl. *Drive-by-Wire*) vom Hersteller PARAVAN GmbH, welches ursprünglich als Fahr- und Lenkhilfe für behinderte Personen entwickelt wurde, heute aber auch zur Umrüstung autonomer Roboterfahrzeuge genutzt wird. Dieses System bietet eine analoge Schnittstelle für die Sollwertvorgabe.

PARAVAN SpaceDrive II

Ein nachrüstbares, straßenzugelassenes DbW-System vom Hersteller PARAVAN GmbH, welches ursprünglich als Fahr- und Lenkhilfe für behinderte Personen entwickelt wurde, heute aber auch zur Umrüstung autonomer Roboterfahrzeuge

genutzt wird. Dieses System bietet eine digitale CAN-Bus-Schnittstelle (Controller Area Network) für die Sollwertvorgabe.

Velodyne HDL-32E

High Definition LiDAR der Firma Velodyne

Velodyne HDL-64E S1

High Definition LiDAR der Firma Velodyne, 1. Generation

Velodyne HDL-64E S2

High Definition LiDAR der Firma Velodyne, 2. Generation

Velodyne VLP-16 Puck

LiDAR der Firma Velodyne

Velodyne VLS-128

High Definition LiDAR der Firma Velodyne

Xsens MTi

kleiner IMU-Sensor auf MEMS-Basis

Publikationen als Erst- oder Co-Autor

- A. Backhaus, T. Luettel, und H.-J. Wuensche, "Yolopoint: Joint Keypoint and Object Detection," in *Proceedings of Advanced Concepts for Intelligent Vision Systems (ACIVS)*, Ser. Lecture Notes in Computer Science, Bd. 14124. Kumamoto, Japan: Springer, Aug. 2023.
- S. F. X. Bayerl, T. Luettel, und H.-J. Wuensche, "Following Dirt Roads at Night Time: Sensors and Features for Lane Recognition and Tracking," in *Proceedings of 7th Workshop On Planning, Perception and Navigation for Intelligent Vehicles (PPNIV), IEEE/RSJ International Conference on Intelligent Robots and Systems*, Hamburg, Germany, Sep. 2015.
- L. Beer, T. Luettel, und H.-J. Wuensche, "GenPa-SLAM: Using a General Panoptic Segmentation for a Real-Time Semantic Landmark SLAM," in *Proceedings of IEEE Intelligent Transportation Systems Conference (ITSC)*, Macau, China, Okt. 2022, S. 873–879.
- , "Präzise GNSS-lose Lokalisierung Autonomer Landfahrzeuge im Urbanen und Suburbanen Raum," in *Tagungsband DWT-SGW Forum Unmanned Systems IV*, Bonn, Germany, Apr. 2023.
- P. Berthold, M. Michaelis, T. Luettel, D. Meissner, und H.-J. Wuensche, "Radar Reflection Characteristics of Vehicles for Contour and Feature Estimation," in *11th Symposium Sensor Data Fusion: Trends, Solutions and Applications*, Bonn, Germany, Okt. 2017.
- , "An Abstracted Radar Measurement Model for Extended Object Tracking," in *Proceedings of IEEE Intelligent Transportation Systems Conference (ITSC)*, Maui, Hawaii, USA, Nov. 2018.
- , "A Radar Measurement Model for Extended Object Tracking in Dynamic Scenarios," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Paris, France, Juni 2019.
- , "Deriving Spatial Occupancy Evidence from Radar Detection Data," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Las Vegas, NV, USA (Virtual Conference), Okt. 2020.
- , "A Continuous Probabilistic Origin Association Filter for Extended Object Tracking," in *Proceedings of IEEE International Conference on Multisensor Fusion and Integration (MFI)*, Karlsruhe, Germany (Virtual Conference), Sep. 2020.
- P. Berthold, B. Forkel, T. Luettel, M. Kusenbach, M. Michaelis, und H.-J. Wuensche, "Kalibrierung und Synchronisierung – Abschlussbericht zur BAAINBw-Studie E E810 IC045 IF010 „Wahrnehmung in komplexer Umgebung“ AP 8," Universität der Bundeswehr München, Neubiberg, Tech. Ber. UniBwM / LRT / TAS / TR 2021:5, Dez. 2021.

- P. Berthold, M. Michaelis, T. Luettel, D. Meissner, und H.-J. Wuensche, "Probabilistic Vehicle Tracking with Sparse Radar Detection Measurements," *ISIF Journal of Advances in Information Fusion*, Bd. 17, no. 2, S. 116–139, Dez. 2022. [Online]. Verfügbar: https://confcats_isif.s3.amazonaws.com/web-files/journals/entries/PagesfromJAIF-v17-i2-4_4.pdf
- P. Burger, T. Luettel, und H.-J. Wuensche, "Positionierung im Raum – Die landmarkenbasierte Lokalisierung zur Navigation von autonomen Landfahrzeugen," in *IT-Report 2020*. Bonn, Germany: Mittler Report Verlag, Apr. 2020, Bd. 1, S. 46–48. [Online]. Verfügbar: https://www.hardthoehenkurier.de/images/downloads/IT-Report+/IT_REPORT+_2020_WEB.pdf
- F. Ebert, P. Berthold, P. Burger, T. Engler, A. Frericks, B. C. Heinrich, J. Kallwies, M. Kusenbach, T. Luettel, K. A. Metzger, M. Michaelis, B. Naujoks, A. F. Sticht, und H.-J. Wuensche, "ELROB 2018 – Convoy and Mule of Team MuCAR," in *Proceedings of 21th International Symposium on Measurement and Control in Robotics (ISMCR)*, Mons, Belgium, Sep. 2018. [Online]. Verfügbar: <https://www.ici-belgium.be/wp-content/uploads/2017/11/PROCEEDINGS-PART-3.pdf>
- D. Fassbender, B. C. Heinrich, T. Luettel, und H.-J. Wuensche, "An Optimization Approach to Trajectory Generation for Autonomous Vehicle Following," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vancouver, BC, Canada, Sep. 2017.
- B. Forkel, T. Luettel, und H.-J. Wuensche, "Umgebungsmodell – Abschlussbericht zur BAAINBw-Studie E E810 IC045 IF010 „Wahrnehmung in komplexer Umgebung“ AP 5," Universität der Bundeswehr München, Neubiberg, Tech. Ber. UniBwM / LRT / TAS / TR 2021:3, Dez. 2021.
- B. Forkel, J. Kallwies, T. Luettel, und H.-J. Wuensche, "Straßen- und Wege-Tracking – Abschlussbericht zur BAAINBw-Studie E E810 IC045 IF010 „Wahrnehmung in komplexer Umgebung“ AP 6," Universität der Bundeswehr München, Neubiberg, Tech. Ber. UniBwM / LRT / TAS / TR 2023:1, Feb. 2023.
- C. Fries, T. Luettel, und H.-J. Wuensche, "Combining Model- and Template-based Vehicle Tracking for Autonomous Convoy Driving," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Gold Coast, Queensland, Australia, Juni 2013.
- C. Fries, P. Burger, J. Kallwies, B. Naujoks, T. Luettel, und H.-J. Wuensche, "How MuCAR Won the Convoy Scenario at ELROB 2016," in *Proceedings of IEEE Intelligent Transportation Systems Conference (ITSC)*, Yokohama, Japan, Okt. 2017.
- F. Hecker, T. Luettel, und H.-J. Wuensche, "Object Related Reactive Offset Maneuver," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, San Diego, CA, USA, Juni 2010, S. 1092–1097.

- N. F. Heide, P. Mortimer, J. Petereit, T. Luettel, und H.-J. Wuensche, "Das GOOSE-Dataset: Ein gemeinsamer Datensatz für KI-Anwendungen," in *Tagungsband DWT-SGW Forum Unmanned Systems VIII*, Bonn, Germany, Okt. 2021.
- B. C. Heinrich, T. Luettel, und H.-J. Wuensche, "A New Control Architecture for MuCAR," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Redondo Beach, CA, USA, Juni 2017.
- B. C. Heinrich, T. Luettel, D. Fassbender, P. Burger, F. Ebert, M. Himmelsbach, H. Jaspers, M. Kusenbach, G. R. Mueller, B. Naujoks, F. Orth, F. Schmitt, und H.-J. Wuensche, "Prototyping an Autonomous Delivery Vehicle," *at – Automatisierungstechnik*, Bd. 66, no. 2, S. 160–182, 2018.
- M. Himmelsbach, T. Luettel, und H.-J. Wuensche, "Real-time Object Classification in 3D Point Clouds Using Point Feature Histograms," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, St. Louis, MO, USA, Okt. 2009, S. 994–1000.
- M. Himmelsbach, F. von Hundelshausen, T. Luettel, M. Manz, A. Mueller, S. Schneider, und H.-J. Wuensche, "Team MuCAR-3 at C-ELROB 2009," in *Proceedings of 1st Workshop on Field Robotics, Civilian European Land Robot Trial 2009*, Ser. ISBN 978-951-42-9176-0. Oulu, Finland: University of Oulu, Juni 2009. [Online]. Verfügbar: <https://www.mucar3.de/bib/mihi/himmelsbach2009celrob.pdf>
- M. Himmelsbach, T. Luettel, F. Hecker, F. von Hundelshausen, und H.-J. Wuensche, "Autonomous Off-Road Navigation for MuCAR-3 – Improving the Tentacles Approach: Integral Structures for Sensing and Motion," *KI – Künstliche Intelligenz*, Bd. 25, no. 2, S. 145–149, 2011, special Issue on Off-Road-Robotics.
- M. Himmelsbach, A. Müller, T. Lüttel, und H.-J. Wünsche, "LIDAR-based 3D Object Perception," in *Proceedings of 1st International Workshop on Cognition for Technical Systems*, München, Okt. 2008.
- J. Kallwies, T. Luettel, und H.-J. Wuensche, "Sensordatenfusion bei Assistenzsystemen für militärische Konvois," in *Tagungsband Wehrtechnisches F&T Symposium „Assistenzsysteme“*, Mannheim, Germany, Juni 2015.
- M. Kusenbach und T. Lüttel, "Sensoren für autonome Fahrzeuge," in *Fortschrittsbericht MORE – Jahresbericht Dezember 2021*, C. Trapp, Hrsg. Neubiberg, Germany: Universität der Bundeswehr München, Munich Mobility Research Campus (MORE), Dez. 2021, Kap. 4.3.2.
- M. Kusenbach, T. Luettel, und H.-J. Wuensche, "Enhanced Temporal Data Organization for LiDAR Data in Autonomous Driving Environments," in *Proceedings of IEEE Intelligent Transportation Systems Conference (ITSC)*, Auckland, New Zealand, Okt. 2019.
- , "MuCAR-3 auf der M-ELROB 2018: Herausforderungen in Konvoi- und Mule-Szenarien," in *Tagungsband DWT-SGW Forum Unmanned Systems VII*, Bonn, Germany, Juni 2019.

- , “Fast Object Classification for Autonomous Driving Using Shape and Motion Information Applying the Dempster-Shafer Theory,” in *Proceedings of IEEE Intelligent Transportation Systems Conference (ITSC)*, Rhodes, Greece (Virtual Conference), Sep. 2020.
- T. Luettel und H.-J. Wuensche, “Automatisierung in militärischen Konvois: Sachstand und Ausblick,” in *Tagungsband DWT-SGW Forum Unmanned Vehicles IV*, Bonn, Germany, Mai 2013.
- T. Luettel, M. Himmelsbach, F. von Hundelshausen, M. Manz, A. Mueller, und H.-J. Wuensche., “Autonomous Offroad Navigation Under Poor GPS Conditions,” in *Proceedings of 3rd Workshop On Planning, Perception and Navigation for Intelligent Vehicles (PPNIV), IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, MO, USA, Okt. 2009.
- T. Luettel, M. Himmelsbach, M. Manz, A. Mueller, F. von Hundelshausen, und H.-J. Wuensche, “Combining Multiple Robot Behaviors for Complex Off-Road Missions,” in *Proceedings of IEEE Intelligent Transportation Systems Conference (ITSC)*, Washington, DC, USA, Okt. 2011.
- T. Luettel, M. Manz, und H.-J. Wuensche, “Erweiterung der EMS Vision-Software zur Steigerung der Autonomie: Meilensteinpraesentation 29.11.2011,” Nov. 2011.
- , “Unbemanntes Konvoi-Fahren auf und abseits von Wegen,” in *Wehrtechnischer Report – Unbemannte Systeme*. Bonn-Bad Godesberg, Germany: Mittler Report Verlag, Dez. 2011, Bd. 7, S. 79–81. [Online]. Verfügbar: <https://www.mucar3.de/bib/thlu/luettel2011wtr.pdf>
- T. Luettel, M. Himmelsbach, und H.-J. Wuensche, “Autonomous Ground Vehicles – Concepts and a Path to the Future,” *Proceedings of the IEEE*, Bd. 100, no. Special Centennial Issue, S. 1831–1839, Mai 2012.
- T. Luettel, F. Ebert, D. Fassbender, C. Fries, B. C. Heinrich, H. Jaspers, J. Kallwies, B. Naujoks, und H.-J. Wuensche, “Assistenz- und Autonomiefunktionen zur Entlastung des Fahrers in Offroad-Szenarien,” *at – Automatisierungstechnik*, Bd. 65, no. 7, 2017.
- T. Lüttel, D. Fassbender, C. Fries, B. Heinrich, und M. Himmelsbach, “Anhang Abschlussbericht UniBwM,” in *Projektabschlussbericht – Technologieträger Unbemanntes Landfahrzeug (TULF)*. Rheinmetall Landsysteme GmbH, März 2014, Kap. A, S. 93–142.
- T. Lüttel, J. Kallwies, und C. Fries, “Robustheitssteigerung Konvoi-System für das Vorhaben StrAsRob – Abschlussbericht – Studie E UR7P CA117 CF141,” Universität der Bundeswehr München, Neubiberg, Tech. Ber. UniBwM / LRT / TAS / TR 2015:19, Dez. 2015.
- M. Manz, M. Himmelsbach, T. Luettel, und H.-J. Wuensche, “Fusing LIDAR and Vision for Autonomous Dirt Road Following – Incorporating a Visual Feature into the Tentacles Approach,” in *Autonome Mobile Systeme (AMS)*, Ser. Informatik

- aktuell, R. Dillmann, J. Beyerer, C. Stiller, J. M. Zöllner, und T. Gindele, Hrsg. Berlin, Heidelberg: Springer, Dez. 2009, S. 17–24.
- , “Detection and Tracking of Road Networks in Rural Terrain By Fusing Vision and LIDAR,” in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, San Francisco, CA, USA, Sep. 2011.
- M. Manz, T. Luettel, F. von Hundelshausen, und H.-J. Wuensche, “Monocular Model-Based 3D Vehicle Tracking for Autonomous Vehicles in Unstructured Environment,” in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, Mai 2011.
- M. Michaelis, P. Berthold, T. Luettel, D. Meissner, und H.-J. Wuensche, “A Merging Strategy for Gaussian Process Extended Target Estimates in Multi-Sensor Applications,” in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Paris, France, Juni 2019.
- , “Extended Object Tracking with an Improved Measurement-to-Contour Association,” in *Proceedings of IEEE International Conference on Information Fusion (FUSION)*, Rustenburg, South Africa (Virtual Conference), Juni 2020.
- M. Michaelis, P. Berthold, T. Luettel, und H.-J. Wuensche, “Extended Target Tracking with a Particle Filter Using State Dependent Target Measurement Models,” in *Proceedings of International Conference on Information Fusion (FUSION)*, Linköping, Sweden, Juli 2022.
- A. Mueller, M. Himmelsbach, T. Luettel, F. von Hundelshausen, und H.-J. Wuensche, “GIS-Based Topological Robot Localization through LIDAR Crossroad Detection,” in *Proceedings of IEEE Intelligent Transportation Systems Conference (ITSC)*, Washington, DC, USA, Okt. 2011.
- B. Naujoks, T. Engler, M. Michaelis, T. Luettel, und H.-J. Wuensche, “Considering Measurement Uncertainty in Dynamic Object Tracking for Autonomous Driving Applications (Bedeutung der Messunsicherheit beim Tracking dynamischer Objekte im Bereich des autonomen Fahrens),” *tm – Technisches Messen*, Bd. 85, no. 12, S. 764–778, 2018.
- S. Schneider, M. Himmelsbach, T. Luettel, und H.-J. Wuensche, “Fusing Vision and LIDAR - Synchronization, Correction and Occlusion Reasoning,” in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, San Diego, CA, USA, Juni 2010, S. 388–393.
- S. Schneider, T. Luettel, und H.-J. Wuensche, “Odometry-based Online Extrinsic Sensor Calibration,” in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Tokyo, Japan, Nov. 2013, S. 1287–1292.
- F. von Hundelshausen, T. Luettel, und H.-J. Wuensche, “Cognitive Navigation – An Overview of three Navigation Paradigms Leading to the Concept of an Affordance Hierarchy,” *KI – Künstliche Intelligenz*, Bd. 25, no. 2, S. 125–132, 2011, special Issue on Off-Road-Robotics.

Literaturverzeichnis

- 106th Congress of the United States of America, "National Defense Authorization Act for Fiscal Year 2001," Okt. 2000. [Online]. Verfügbar: <https://www.gpo.gov/fdsys/pkg/PLAW-106publ398/pdf/PLAW-106publ398.pdf>
- E. Ackerman. (2019, Jan.) Apex.AI Does the Invisible Work That Will Make Self-Driving Cars Possible – Seeing the road is a challenge for self-driving cars, but so is developing a secure and reliable software framework. IEEE Spectrum. [Online]. Verfügbar: <https://spectrum.ieee.org/apexai-does-the-invisible-critical-work-that-will-make-selfdriving-cars-possible>
- E. Ackerman und E. Guizzo. (2017, Nov.) Wizards of ROS: Willow Garage and the Making of the Robot Operating System – How a small band of Silicon Valley engineers started a global robotics revolution. IEEE Spectrum. [Online]. Verfügbar: <https://spectrum.ieee.org/wizards-of-ros-willow-garage-and-the-making-of-the-robot-operating-system>
- R. Ackermann, "Improvements on Axletrees applicable to Four-wheeled Carriages," GB patent 4212, 1818.
- AdV-Arbeitsgruppe ATKIS. (2003, Juli) ATKIS – Objektartenkatalog Basis-DLM. online. AdV-Arbeitsgruppe ATKIS, AdV-Arbeitskreis Geotopographie. München, Germany. [Online]. Verfügbar: <http://www.atkis.de>
- M. Aramrattana, J. Detournay, C. Englund, V. Frimodig, O. U. Jansson, T. Larsson, W. Mostowski, V. D. Rodríguez, T. Rosenstatter, und G. Shahanoor, "Team Halmstad Approach to Cooperative Driving in the Grand Cooperative Driving Challenge 2016," *IEEE Transactions on Intelligent Transportation Systems*, Bd. 19, no. 4, S. 1248–1261, 2018.
- Audi AG, *Self-study programme 633 - Audi Q7 (type 4M) Chassis*, Ingolstadt, Feb. 2015. [Online]. Verfügbar: <https://www.vdveer-engineering.nl/en/information/vag-ssp/ssp-vag/ssp-vag-en/412-ssp-633-audi-q7-type-4m-chassis/download>
- , *Self-study programme 672 - Audi Q8 (type 4M)*, Ingolstadt, Mai 2018. [Online]. Verfügbar: <https://www.vdveer-engineering.nl/en/information/vag-ssp/ssp-vag/ssp-vag-fr/517-ssp-672-audi-q8-type-4m/download>
- Australian Road Research Board (ARRB). (2022, Aug.) Autonomous Leader Follower Convoy been Developed for Defence. [Online]. Verfügbar: <https://www.arrb.com.au/news/autonomous-leader-follower-convoy-being-developed-for-defence>
- A. Bacha, C. Bauman, R. Faruque, M. Fleming, C. Terwelp, C. Reinholtz, D. Hong, A. Wicks, T. Alberi, D. Anderson, S. Cacciola, P. Currier, A. Dalton, J. Farmer, J. Hurdus, S. Kimmel, P. King, A. Taylor, D. V. Covern, und M. Webster, "Odin: Team VictorTango's entry in the DARPA Urban Challenge," *Journal of Field Robotics*, Bd. 25, no. 9, S. 569–597, 2008.

- A. Backhaus, T. Luettel, und H.-J. Wuensche, "Yolopoint: Joint Keypoint and Object Detection," in *Proceedings of Advanced Concepts for Intelligent Vision Systems (ACIVS)*, Ser. Lecture Notes in Computer Science, Bd. 14124. Kumamoto, Japan: Springer, Aug. 2023.
- M. Ball. (2018, Sep.) Autonomous Snowploughs to be Tested at Oslo Airport. Unmanned Systems Technology. [Online]. Verfügbar: <https://www.unmannedsystemstechnology.com/2018/09/autonomous-snowploughs-to-be-tested-at-oslo-airport/>
- . (2022, Nov.) Autonomous Convoy Capabilities for Military Trucks Demonstrated. Unmanned Systems Technology. [Online]. Verfügbar: <https://www.unmannedsystemstechnology.com/2022/11/autonomous-convoy-capabilities-for-military-trucks-demonstrated/>
- M. C. A. Baltzer, C. Rudolph, D. López, und F. Flemisch, *Cooperative Guidance and Control in Highly Automated Convoys—StrAsRob*. Cham: Springer International Publishing, 2017, S. 547–559. [Online]. Verfügbar: http://dx.doi.org/10.1007/978-3-319-41682-3_46
- Y. Bar-Shalom, X.-R. Li, und T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. New York, NY, USA: John Wiley & Sons, Inc., 2001.
- Y. Bar-Shalom, H. Chen, M. Mallick, und R. Washburn, "One-Step Solution for the General Out-of-Sequence-Measurement Problem in Tracking," in *Proceedings of IEEE Aerospace Conference*, Bd. 4, no. 1, 2002, S. 1551–1559.
- M. Barjenbruch, D. Kellner, J. Klappstein, J. Dickmann, und K. Dietmayer, "Joint Spatial- and Doppler-based Ego-Motion Estimation for Automotive Radars," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, 2015, S. 839–844.
- K. Barry. (2019, Mai) Tesla's Updated Navigate on Autopilot Requires Significant Driver Intervention. Consumer Reports. [Online]. Verfügbar: <https://www.consumerreports.org/autonomous-driving/tesla-navigate-on-autopilot-automatic-lane-change-requires-significant-driver-intervention/>
- BASF. (2017, Mai) BASF optimiert Logistik am Standort Ludwigshafen. BASF. Ludwigshafen, Germany. [Online]. Verfügbar: <https://www.basf.com/global/de/who-we-are/organization/locations/europe/german-sites/ludwigshafen/the-site/news-and-media/news-releases/2017/05/p-17-183.html>
- S. F. X. Bayerl, "Erkennung von Wegen und Kreuzungen – Abschlussbericht Phase I 2012-2015," Universität der Bundeswehr München, Neubiberg, Tech. Ber. UniBwM / LRT / TAS / TR 2015:14, Dez. 2015.
- S. F. X. Bayerl und H.-J. Wuensche, "Detection and Tracking of Rural Crossroads Combining Vision and LiDAR Measurements," in *Proceedings of IEEE Intelligent Transportation Systems Conference (ITSC)*, Qingdao, China, Okt. 2014.

- J. Becker. (2021, Okt.) 'Wir können die Städte nicht mit privaten Pkw vollpumpen'. *Süddeutsche Zeitung*. München, Germany. [Online]. Verfügbar: <https://www.sueddeutsche.de/auto/autonomes-fahren-verkehrswendestadtverkehr-1.5436406>
- M. Belfiore. (2007, Nov.) 3 Bots Neck-and-Neck in Robot Car Race. *wired.com*. [Online]. Verfügbar: <https://web.archive.org/web/20071105085239/http://blog.wired.com/defense/2007/11/3-bots-neck-and.html>
- . (2007, Nov.) 3 Teams Out of the Running at Auto-Bot Race. *wired.com*. [Online]. Verfügbar: <https://web.archive.org/web/20071106093639/http://blog.wired.com/defense/2007/11/3-teams-out-of-.html>
- R. Bellan. (2021, Nov.) Cruise launches driverless robotaxi service in San Francisco. *TechCrunch*. [Online]. Verfügbar: <https://techcrunch.com/2021/11/03/cruise-launches-driverless-robotaxi-service-for-employees-in-san-francisco/>
- R. E. Bellman, "On a routing problem," *Quarterly of Applied Mathematics*, Bd. 16, S. 87–90, 1958. [Online]. Verfügbar: <https://doi.org/10.1090/qam/102435>
- P. Bellmer. (2023, Jan.) Snapdragon Ride Flex: Qualcomm nennt erste Details zum Zentralrechner fürs Auto. *Heise Online*. Hannover, Germany. [Online]. Verfügbar: <https://www.heise.de/news/Snapdragon-Ride-Flex-Qualcomm-nennt-erste-Details-zum-Zentralrechner-fuers-Auto-7447658.html>
- C. Berghem, S. Shladover, E. Coelingh, C. Englund, und S. Tsugawa, "Overview of Platooning Systems," in *Proceedings of the 19th ITS World Congress*, Vienna, Austria, Okt. 2012.
- C. Berger und B. Rumpe, "Autonomes Fahren – Erkenntnisse aus der DARPA Urban Challenge," *it – Information Technology*, Bd. 50, no. 4, S. 258–264, 2008.
- M. Bertozzi, L. Bombini, A. Broggi, M. Buzzoni, E. Cardarelli, S. Cattani, P. Cerri, A. Coati, S. Debattisti, A. Falzoni, R. Fedriga, M. Felisa, L. Gatti, A. Giacomazzo, P. Grisleri, M. Laghi, L. Mazzei, P. Medici, M. Panciroli, P. Porta, P. Zani, und P. Versari, "VIAC: an Out of Ordinary Experiment," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Baden-Baden, Germany, Juni 2011, S. 175–180.
- M. Bertozzi und A. Broggi, "GOLD: a Parallel Real-Time Stereo Vision System for Generic Obstacle and Lane Detection," *IEEE Transactions on Image Processing*, Bd. 7, no. 1, S. 62–81, Jan. 1998.
- M. Bertozzi, A. Broggi, E. Cardarelli, R. I. Fedriga, L. Mazzei, und P. P. Porta, "VIAC Expedition Toward Autonomous Mobility," *IEEE Robotics and Automation Magazine*, Bd. 18, no. 3, S. 120–124, Sep. 2011.
- BEVER TMC B.V., *Einbauanleitung - Elektrische Gas-/Bremsanlage Universal 2996000000*, De Andelst, Niederlande, 2008.

- D. M. Bevy und B. Parkinson, "Cascaded Kalman Filters for Accurate Estimation of Multiple Biases, Dead-Reckoning Navigation, and Full State Feedback Control of Ground Vehicles," *IEEE Transactions on Control Systems Technology*, Bd. 15, no. 2, S. 199–208, 2007.
- D. Bird, C. Beal, A. Thomson, und C. Vinson, "New Technology & Innovation: Report 2 – Autonomous Mining Equipment," RFC Cambrian, Report, Mai 2019. [Online]. Verfügbar: <https://www.rfcambrian.com/wp-content/uploads/2019/04/RFCAN-TI-Report-2-Autonomous-Mining-Equipment-May-2019.pdf>
- J.-L. Blanco, J.-A. Fernández-Madriral, und J. Gonzalez-Jimenez, "Towards a Unified Bayesian Approach to Hybrid Metric-Topological SLAM," *IEEE Transactions on Robotics*, Bd. 24, no. 2, S. 259–270, 2008.
- BMW Group. (2019, Feb.) BMW Group und Daimler wollen zusammen die nächste Technologiegeneration für automatisiertes Fahren entwickeln. Pressemitteilung. BMW Group. [Online]. Verfügbar: <https://www.press.bmwgroup.com/deutschland/article/detail/T0292550DE/bmw-group-und-daimler-wollen-zusammen-die-naechste-technologiegeneration-fuer-automatisiertes-fahren-entwickeln>
- J. Bolliger. (2023, Mai) Der Autopilot von Tesla soll für zahlreiche Crashes verantwortlich sein. Tagesanzeiger. [Online]. Verfügbar: <https://www.tagesanzeiger.ch/der-autopilot-von-tesla-soll-fuer-zahlreiche-crashes-verantwortlich-sein-166089029882>
- R. Bongiovanni und J. Lowenberg-Deboer, "Precision Agriculture and Sustainability," *Precision Agriculture*, Bd. 5, S. 359–387, 2004.
- Bosch und CARIAD. (2022, Jan.) Automatisiertes Fahren: Bosch und Volkswagen Konzerntochter Cariad vereinbaren umfassende Zusammenarbeit. Pressemitteilung. Robert Bosch GmbH. [Online]. Verfügbar: <https://www.bosch-presse.de/pressportal/de/de/automatisiertes-fahren-bosch-und-volkswagen-konzerntochter-cariad-vereinbaren-umfassende-zusammenarbeit-237313.html>
- Bosch Telecom und Fraunhofer IITB. (1993, Okt.) Autonome Führung von Straßenfahrzeugen. Hildesheim, Germany. [Online]. Verfügbar: <https://www.youtube.com/watch?v=JTnBiTlvGqY>
- W. Boston und M. Colias. (2019, Feb.) VW to Invest Nearly \$2 Billion in Ford Self-Driving Car Venture. *The Wall Street Journal*. [Online]. Verfügbar: <https://www.wsj.com/articles/vw-to-invest-nearly-2-billion-in-ford-self-driving-car-venture-11551185359>
- D. Braid, A. Broggi, und G. Schmiedel, "The TerraMax Autonomous Vehicle," *Journal of Field Robotics*, Bd. 23, no. 9, S. 693–708, Sep. 2006.
- A. Broggi. (2001) The ARGO Project. Universität Parma. Parma, Italien. Archived version, last change regarding October 2001. [Online]. Verfügbar: <https://vislab.it/argo/ARGO/english/index.html>

- A. Broggi, M. Bertozzi, A. Fascioli, und G. Conte, *Automatic Vehicle Guidance: The Experience of the ARGO Autonomous Vehicle*. Singapur: World Scientific Publishing Co. Pte. Ltd., Apr. 1999.
- A. Broggi, M. Bertozzi, und A. Fascioli, "ARGO and the MilleMiglia in Automatico Tour," *IEEE Intelligent Systems and their Applications*, Bd. 14, no. 1, S. 55–64, Jan. 1999.
- A. Broggi, P. Cerri, M. Felisa, M. C. Laghi, L. Mazzei, und P. P. Porta, "The VisLab Intercontinental Autonomous Challenge: an Extensive Test for a Platoon of Intelligent Vehicles," *International Journal of Vehicle Autonomous Systems*, Bd. 10, no. 3, S. 147–164, 2011.
- C. Brüninghaus. (2016, März) Highway Pilot Connect vernetzt Lkw zum Platoon. Springer Professional. [Online]. Verfügbar: <https://www.springerprofessional.de/nutzfahrzeuge/automatisiertes-fahren/highway-pilot-connect-vernetzt-lkw-zum-platoon/7839058>
- T. Bruns, "Modellierung der Längsdynamik und Entwurf eines Beschleunigungsreglers für ein autonomes Kraftfahrzeug," Diplomarbeit UniBwM / LRT / TAS / DA 2008:2, Universität der Bundeswehr München, Neubiberg, Feb. 2008.
- C. Büchler. (2020, Apr.) Präzision und Frequenzstabilität: Die Stärken von 32,768 kHz-Oszillatoren. Jauch Blog. Jauch Quartz GmbH. Villingen-Schwenningen, Germany. [Online]. Verfügbar: <https://www.jauch.com/blog/praezision-und-frequenzstabilitaet-die-staerken-von-32768-khz-oszillatoren/>
- M. Buehler, K. Iagnemma, und S. Singh, Hrsg., *The 2005 DARPA Grand Challenge: The Great Robot Race*, Ser. Springer Tracts in Advanced Robotics (STAR), Bd. 36. Springer, 2007.
- , *The DARPA Urban Challenge: Autonomous Vehicles in City Traffic*, Ser. Springer Tracts in Advanced Robotics (STAR), Bd. 56. Springer, 2009.
- Bundesamt für Wehrtechnik und Beschaffung, "Leistungsbeschreibung – E UR7P CA117 CF141 – „Anteil Robustheitssteigerung Konvoi-System“ für das Vorhaben „Straßentransport mit Assistenzfunktionen von Robotern (StrAsRob)“,“ Juli 2012.
- Bundesanstalt für Straßenwesen. (2021, März) Selbstfahrende Autos – assistiert, automatisiert oder autonom? Pressemitteilung. Bundesanstalt für Straßenwesen. Bergisch Gladbach, Germany. [Online]. Verfügbar: <https://www.bast.de/DE/Presse/Mitteilungen/2021/06-2021.html>
- Bundesministerium für Digitales und Verkehr. (2021, Juli) Gesetz zum autonomen Fahren tritt in Kraft. Bundesministerium für Digitales und Verkehr. [Online]. Verfügbar: <https://bmdv.bund.de/SharedDocs/DE/Artikel/DG/gesetz-zum-autonomen-fahren.html>

- P. Burger, "Kooperative Lokalisierung und Kartierung in unstrukturierter Umgebung," Dissertation, Universität der Bundeswehr München, Fakultät für Luft- und Raumfahrttechnik, Neubiberg, Mai 2022. [Online]. Verfügbar: <https://athene-forschung.unibw.de/143546>
- P. Burger und H.-J. Wuensche, "Fast Multi-pass 3D Point Segmentation Based on a Structured Mesh Graph for Ground Vehicles," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Changshu, Suzhou, China, Juni 2018.
- A. Burkert. (2018, Juli) *Autonom und fehlerfrei über den Betriebshof*. Springer Professional. [Online]. Verfügbar: <https://www.springerprofessional.de/nutzfahrzeuge/automatisiertes-fahren/autonom-und-fehlerfrei-ueber-den-betriebshof/15944492>
- J. Bursztynsky und H. McCracken. (2023, März) *Aurora's self-driving truck tech is one step closer to hitting the road*. Fast Company. [Online]. Verfügbar: <https://www.fastcompany.com/90870517/auroras-self-driving-truck-tech-is-one-step-closer-to-hitting-the-road>
- M. Böhm, "Entwicklung geeigneter Verfahren zur Sicherstellung der Reaktionsfähigkeit eines Sicherheitsfahrers in einem autonomen Kraftfahrzeug," Studienarbeit UniBwM / LRT / TAS / SA 2007:1, Universität der Bundeswehr München, Neubiberg, 2007.
- G. Campion und W. Chung, "Wheeled robots," in *Springer Handbook of Robotics*, B. Siciliano und O. Khatib, Hrsg. Springer, 2008, S. 391–410. [Online]. Verfügbar: <http://www.springerlink.com/content/r6m219/>
- Y. Chen, V. Sundareswaran, C. Anderson, A. Broggi, P. Grisleri, P. P. Porta, P. Zani, und J. Beck, "TerraMax™: Team Oshkosh Urban Robot," *Journal of Field Robotics*, Bd. 25, no. 10, S. 841–860, 2008.
- Y. Clerico, S. Scheugenpflug, und D. Schleder, *Geodaten in der Praxis - Hauptdokument*, 1. Ausg., Bayerische Vermessungsverwaltung, Sep. 2009. [Online]. Verfügbar: https://www.ldbv.bayern.de/file/pdf/2067/Hauptdokument_Geodaten_in_der_Praxis.pdf
- B. Conrad und L. Leicht. (2022, Okt.) *Aus für Robo-Auto-Entwickler*. Auto Motor Sport – MOOVE Magazin. [Online]. Verfügbar: <https://www.auto-motor-und-sport.de/verkehr/ford-volkswagen-argo-ai-ende/>
- Daimler AG. (2018) *Definition „autonomes fahren“*. Daimler AG. Stuttgart, Germany. [Online]. Verfügbar: <https://www.daimler.com/innovation/autonomes-fahren/special/definition.html>
- Daimler Communications. (2007, Okt.) *1986: Startschuss zum PROMETHEUS-Forschungsprogramm*. Pressemitteilung. Stuttgart, Germany. [Online]. Verfügbar: <https://media.mercedes-benz.com/article/e7142515-d8a3-44d9-be3d-5b1cb09f05f4>

- . (2017, Okt.) Schneeräumen auf dem Flugfeld: Automatisierte Mercedes-Benz Arocs machen den Weg frei. Pressemitteilung. Stuttgart / Bad Sobernheim, Germany. [Online]. Verfügbar: <https://media.daimler.com/marsMediaSite/de/instance/ko/Schneeraeumen-auf-dem-Flugfeld-Automatisierte-Mercedes-Benz-Arocs-machen-den-Weg-frei.xhtml?oid=29890415>
- Daimler Trucks. (2019, März) Daimler Trucks erwirbt Mehrheitsanteil von Torc Robotics. Daimler Trucks. [Online]. Verfügbar: <https://www.daimler.com/investoren/berichte-news/finanznachrichten/20190329-daimler-trucks-torc-robotics.html>
- DARPA. (2007, März) Urban Challenge: Route Network Definition File (RNDF) and Mission Data File (MDF) Formats. Defense Advanced Research Projects Agency. Arlington, VA, USA. [Online]. Verfügbar: https://www.grandchallenge.org/grandchallenge/docs/RNDF_MDF_Formats_031407.pdf
- . (2007, Okt.) Urban Challenge: Rules. Defense Advanced Research Projects Agency. Arlington, VA, USA. [Online]. Verfügbar: https://www.grandchallenge.org/grandchallenge/docs/Urban_Challenge_Rules_102707.pdf
- . (2007, Nov.) DARPA Urban Challenge website. Archivierte Version. [Online]. Verfügbar: <https://www.grandchallenge.org/>
- . (2015) DARPA Robotics Challenge website. [Online]. Verfügbar: <https://archive.darpa.mil/roboticschallenge/>
- . (2019) DARPA Subterranean Challenge website. [Online]. Verfügbar: <https://www.subtchallenge.com/>
- S. E. Dean. (2023, Jan.) Main Ground Combat System (MGCS): A Status Report. European Security & Defence. [Online]. Verfügbar: <https://euro-sd.com/2023/01/articles/29122/main-ground-combat-system-mgcs-a-status-report/>
- S. Dent. (2023, Feb.) Amazon's Zoox is now operating its purpose-built autonomous taxi on public roads. *engadget*. [Online]. Verfügbar: <https://www.engadget.com/amazons-zoox-successfully-tests-its-purpose-built-autonomous-taxi-on-public-roads-134628930.html>
- Der Spiegel, "Gepeinigte Regenwürmer am Straßenrand – In Italien testeten Bastler der Universität Parma einen Lancia mit dem billigsten Autopiloten der Welt," *Der Spiegel*, no. 24, S. 178, Juni 1998. [Online]. Verfügbar: <https://magazin.spiegel.de/EpubDelivery/spiegel/pdf/7907551>
- Deutscher Bundestag, Ausschuss für Verkehr und digitale Infrastruktur, "Beschlussempfehlung und Bericht: Entwurf eines ... Gesetzes zur Änderung des Straßenverkehrsgesetzes," Deutscher Bundestag 18. Wahlperiode, Drucksache 18/11776, 2017. [Online]. Verfügbar: <http://dip21.bundestag.de/dip21/btd/18/117/1811776.pdf>

- DICKEY-john, *RADAR II AND III Ground Speed Sensors*, Auburn, IL, USA, Jan. 2020, 5002001 Rev. C. [Online]. Verfügbar: https://dickey-john.com/getmedia/62d81cce-8ec6-43d8-8bec-b4ec3a418624/Radar-II-and-III_US_5002001_RevC_Web_1.pdf
- D. Dickmanns, "Kronos," Universität der Bundeswehr München, LRT, Neubiberg, Germany, Benutzerhandbuch, 1992.
- E. D. Dickmanns. (2020) *Dynamic Machine Vision*. Brunthal, Germany. [Online]. Verfügbar: <https://dyna-vision.de/>
- , "Untersuchungen und Arbeitsschritte zum Thema künstliche Intelligenz: Rechnersehen und -steuerung dynamischer Systeme," Hochschule der Bundeswehr München, Interner Bericht HSBw M/LRT/WE 13a/IB/80-1, 1980.
- , "Steuerung dynamischer Systeme durch Rechnersehen. Beschreibung eines längerfristigen Forschungsvorhabens des Instituts für Systemdynamik und Flugmechanik und des Instituts für Meßtechnik am Fachbereich LRT der HSBw München," Juni 1981.
- , "4-D-Dynamic Scene Analysis with Integral Spatio-Temporal Models," in *Proceedings of the 4th International Symposium on Robotics Research*, R. C. Bolles und B. Roth, Hrsg. Cambridge, MA, USA: MIT Press, Mai 1988, S. 311–318.
- , *Dynamic Vision for Perception and Control of Motion*. London: Springer Verlag, 2007.
- E. D. Dickmanns und B. D. Mysliwetz, "Recursive 3-D Road and Relative Ego-State Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Bd. 14, no. 2, S. 199–213, 1992.
- E. D. Dickmanns und A. Zapp, "Autonomous High Speed Road Vehicle Guidance by Computer Vision," in *Proc. 10th IFAC World Congress*, Bd. 4 (preprint), 1987, S. 232–237.
- E. D. Dickmanns, V. Graefe, und W. Niegel, "Abschlussbericht "Definitionsphase PROMETHEUS, Pro-Art"1.10.86 bis 14.11.87," Nov. 1987. [Online]. Verfügbar: https://dyna-vision.de/wp-content/uploads/2021/05/1987-AbschlBerDefPhase-Pro-Art_UBM.pdf
- N. Dicu, G.-D. Andreescu, und E. HoratiuGurban, "Automotive Dead-Reckoning Navigation System Based on Vehicle Speed and Yaw Rate," in *IEEE International Symposium on Applied Computational Intelligence and Informatics (SACI)*, Mai 2018, S. 225–228.
- E. W. Dijkstra, "A Note on Two Problems in Connexion with Graphs," *Numerische Mathematik*, Bd. 1, no. 1, S. 269–271, Dez. 1959.

- DIN Deutsches Institut für Normung e.V., "DIN 70 000 Straßenfahrzeuge; Fahrzeugdynamik und Fahrverhalten; Begriffe (ISO 8855:1991, modifiziert)," Beuth Verlag, Berlin, Jan. 1994. [Online]. Verfügbar: <http://www.techstreet.com/products/1055555>
- , "DIN ISO 8855 Straßenfahrzeuge - Fahrzeugdynamik und Fahrverhalten - Begriffe (ISO 8855:2011)," Beuth Verlag, Berlin, Nov. 2013, diese Norm ersetzt DIN 70000:1994-01.
- D. Dolgov, S. Thrun, M. Montemerlo, und J. Diebel, "Path Planning for Autonomous Vehicles in Unknown Semi-structured Environments," *The International Journal of Robotics Research*, Bd. 29, no. 5, S. 485–501, 2010.
- S. Doll. (2021, Juli) Toyota's Woven Planet acquires Lyft's Level 5 self-driving division. *electrek*. [Online]. Verfügbar: <https://electrek.co/2021/07/20/toyotas-woven-planet-acquires-lyfts-level-5-self-driving-division/>
- R. Dorobantu, "Simulation des Verhaltens einer low-cost Strapdown-IMU unter Laborbedingungen," Technische Universität München, Institut für Astronomische und Physikalische Geodäsie, Forschungseinrichtung Satellitengeodäsie, München, Germany, Forschungsbericht IAPG/FESG No. 6, 1999. [Online]. Verfügbar: <https://mediatum.ub.tum.de/doc/1365913/784160.pdf>
- F. Dudenhöffer und W. Schneider, "Fehlender rechtlicher Rahmen verschafft den USA Zeitvorsprung beim Zukunftsmarkt »individuelle Mobilität«,," *ifo Schnelldienst*, S. 31–33, Apr. 2015. [Online]. Verfügbar: https://www.ifo.de/DocDL/ifosd_2015_07_2.pdf
- J. Duník, S. K. Biswas, A. G. Dempster, T. Pany, und P. Closas, "State Estimation Methods in Navigation: Overview and Application," *IEEE Aerospace and Electronics Systems Magazine*, Bd. 35, no. 12, S. 16–31, 2020.
- P. Dzikiy. (2019, Juni) Fiat in, VW out, and now Aurora gets fresh investment from Hyundai to build self-driving platform. *electrek*. [Online]. Verfügbar: <https://electrek.co/2019/06/13/hyundai-kia-aurora-self-driving/>
- L. D. Earnest. (2012, Dez.) Stanford cart. Stanford University. [Online]. Verfügbar: <https://web.stanford.edu/~learnest/sail/oldcart.html>
- F. Ebert, D. Fassbender, B. Naujoks, und H.-J. Wuensche, "Robust Long-Range Teach-and-Repeat in Non-Urban Environments," in *Proceedings of IEEE Intelligent Transportation Systems Conference (ITSC)*, Yokohama, Japan, Okt. 2017.
- E. Eckermann, *Die Achsschenkelenkung und andere Fahrzeug-Lenksysteme*, Ser. Technikgeschichte. München: Deutsches Museum, 1998. [Online]. Verfügbar: <http://www.yumpu.com/de/document/view/3500484/die-achsschenkelenkung-und-andere-fahrzeug-lenksysteme>
- W. Eckl-Dorna. (2018, März) Jaguar will mit Google zum Roboterauto-Riesen werden. *Manager Magazin*. [Online]. Verfügbar:

- <https://www.manager-magazin.de/unternehmen/autoindustrie/selbstfahrendes-kfz-kooperation-jaguar-liefert-waymo-20-000-e-roboterautos-a-1200108.html>
- J. Egleder. (2023, Mai) Personal, wo bist du? Verband der Reservisten der Deutschen Bundeswehr e. V. Bonn, Germany. [Online]. Verfügbar: <https://www.reservistenverband.de/magazin-loyal/personalmangel/>
- A. Eidehall und F. Gustafsson, "Combined road prediction and target tracking in collision avoidance," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, 2004, S. 619–624.
- A. Elfes, "Using Occupancy Grids for Mobile Robot Perception and Navigation," *Computer*, Bd. 22, no. 6, S. 46–57, Juni 1989.
- F. Englberger, M. Gerdt, T. Lüttel, und A. Neve, *Fortschrittsbericht MORE – Jahresbericht Dezember 2022 – Teilbereich Vernetzung & Autonomie*, C. Trapp, Hrsg. Neubiberg, Germany: Universität der Bundeswehr München, Munich Mobility Research Campus (MORE), Dez. 2022.
- D. Fassbender, A. Mueller, und H.-J. Wuensche, "Trajectory Planning for Car-Like Robots in Unknown, Unstructured Environments," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Chicago, IL, USA, Sep. 2014.
- D. Fassbender, B. C. Heinrich, und H.-J. Wuensche, "Motion Planning for Autonomous Vehicles in Highly Constrained Urban Environments," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Daejeon, South Korea, Okt. 2016.
- P. J. Fell, "World Geodetic System 1984 - A summary of enhancements through the development of the new geopotential model EGM96," in *Proceedings of MTS/IEEE OCEANS Conference*, Bd. 2, Halifax, NS, Canada, Okt. 1997, S. 1504.
- M. Ferres und J.-P. Klein. (2019, Juli) Decoding the Autonomous Driving Landscape: Software will indeed eat the (automotive) world. Medium. Firstmile VC. [Online]. Verfügbar: <https://medium.com/@firstmilevc/avlandscape-8a21491f1f54>
- B. Feuerborn. (2022, Mai) Lemken und Krone: fahrerloser Traktor mäht und grubbert. agrarheute, Deutscher Landwirtschaftsverlag GmbH. München, Germany. [Online]. Verfügbar: <https://www.agrarheute.com/technik/lemkenkrone-fahrerloser-traktor-maeht-grubbert-593982>
- H. Flämig, "Autonome Fahrzeuge und autonomes Fahren im Bereich des Gütertransportes," in *Autonomes Fahren: Technische, rechtliche und gesellschaftliche Aspekte*, M. Maurer, J. C. Gerdes, B. Lenz, und H. Winner, Hrsg. Berlin, Heidelberg, Germany: Springer Vieweg, 2015, S. 377–398.
- A. Floemer. (2022, März) Aus für Mobileye bei BMW: Level 3 ab 2025 mit Qualcomm und Arriver. t3n. [Online]. Verfügbar: <https://t3n.de/news/fuer-mobileye-bmw-level-3-ab-1458912/>

- . (2023, März) Autonomes Fahren nach Level 3: VW und Bosch schicken erste Fahrzeuge auf die Straße. t3n. [Online]. Verfügbar: <https://t3n.de/news/autonomes-fahren-level-3-vw-bosch-1544827/>
- L. R. Ford, *Network Flow Theory*. Santa Monica, CA, USA: RAND Corporation, Aug. 1956, paper P-923. [Online]. Verfügbar: <https://apps.dtic.mil/dtic/tr/fulltext/u2/422842.pdf>
- Ford Motor Company. (2017, Feb.) Ford invests in Argo AI, a new artificial intelligence company, in drive for autonomous vehicle leadership. San Francisco, CA, USA. [Online]. Verfügbar: <https://media.ford.com/content/fordmedia/fna/us/en/news/2017/02/10/ford-invests-in-argo-ai-new-artificial-intelligence-company.html>
- B. Forkel und H.-J. Wuensche, "Combined Road Tracking for Paved Roads and Dirt Roads: LiDAR Measurements and Image Color Modes," in *Proceedings of International Conference on Information Fusion (FUSION)*, Linköping, Sweden, Juli 2022.
- , "Dynamic Resolution Terrain Estimation for Autonomous (Dirt) Road Driving Fusing LiDAR and Vision," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Aachen, Germany, Juni 2022.
- B. Forkel, J. Kallwies, und H.-J. Wuensche, "Probabilistic Terrain Estimation for Autonomous Off-Road Driving," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, Xi'an, China (Hybrid Conference), Juni 2021.
- , "Combined Road Tracking for Paved Roads and Dirt Roads: Framework and Image Measurements," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Nagoya, Japan (Virtual Conference), Juli 2021.
- Fraunhofer IOSB. (2021, Juli) Fortschritte bei Dekontaminationsaufgaben: ROBDEKON präsentiert Entwicklungsstand. Fraunhofer IOSB. Karlsruhe und Ilmenau, Germany. [Online]. Verfügbar: <https://www.iosb.fraunhofer.de/en/press/press-releases/2021/robdekon-presents-stage-of-development.html>
- C. Fries und H.-J. Wuensche, "Monocular Template-based Vehicle Tracking for Autonomous Convoy Driving," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Chicago, IL, USA, Sep. 2014.
- , "Autonomous Convoy Driving by Night: The Vehicle Tracking System," in *Proceedings of IEEE International Conference on Technologies for Practical Robot Applications (TePRA)*, Boston, MA, USA, Mai 2015.
- S. Fürst, "Cronos," Universität der Bundeswehr München, Neubiberg, Germany, Benutzerhandbuch UniBwM / LRT / WE13 / PB95-1, Jan. 2001, v4.4.2.145.
- FZI Forschungszentrum Informatik. (2019, Mai) Drei autonome Minibusse für Karlsruhe. Pressemitteilung. Karlsruhe, Germany. [Online]. Verfügbar: <https://taf-bw.de/aktuelles/news/detail/drei-autonome-minibusse-fr-karlsruhe0>

- S. Gelowicz. (2021, Okt.) Vogel / Automobil Industrie. [Online]. Verfügbar: <https://www.automobil-industrie.vogel.de/bietergefecht-um-veoneer-qualcomm-gewinnt-gegen-magna-a-1061635/>
- F. Gensler und M. Breitenberger, "Mit einer Messanordnung versehenes Lenkrad für Kraftfahrzeuge," Deutschland Patent DE 10 048 956C1, Mai 2, 2002.
- B. Gerkey. (2015, Dez.) ROS, the Robot Operating System, Is Growing Faster Than Ever, Celebrates 8 Years – From humanoids to industrial arms to self-driving cars, robots powered by ROS are everywhere. IEEE Spectrum. [Online]. Verfügbar: <https://spectrum.ieee.org/ros-robot-operating-system-celebrates-8-years>
- D. Gerlach, J. Pellenz, A. Retterath, und A. Volk, "Landmobilität der Bundeswehr (Teil 5): Automatisierung und unbemanntes Fahren," *Hardthöhenkurier*, Bd. 4/2020, S. 64–69, 2020. [Online]. Verfügbar: <https://hardthoehenkurier.de/hhkemags/hhkfreemags/2020-04/#page=64>
- GIS Geography. (2022) 5 Free Global DEM Data Sources – Digital Elevation Models. [Online]. Verfügbar: <https://gisgeography.com/free-global-dem-data-sources/>
- M. Goebel und G. Färber, "A Real-Time-capable Hard- and Software Architecture for Joint Image and Knowledge Processing in Cognitive Automobiles," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Istanbul, Turkey, Juni 2007.
- P. Groves, *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems, Second Edition*. Artech House, 2013.
- S. Hanley. (2023, Mai) Volkswagen's Plan To Fix Cariad Includes Collaboration With Mobileye. CleanTechnica. [Online]. Verfügbar: <https://cleantechnica.com/2023/05/10/volkswagens-plan-to-fix-cariad-includes-collaboration-with-mobileye/>
- B. J. Harker, "PROMOTE-CHAUFFEUR II & 5.8 GHz vehicle to vehicle communications system," in *International Conference on Advanced Driver Assistance Systems (ADAS)*, Sep. 2001, S. 81–85.
- P. E. Hart, N. J. Nilsson, und B. Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," *IEEE Transactions on Systems Science and Cybernetics*, Bd. 4, no. 2, S. 100–107, 1968. [Online]. Verfügbar: <https://www.cs.auckland.ac.nz/courses/compsci709s2c/resources/Mike.d/astarNilsson.pdf>
- W. Hauger, W. Schnell, und D. Gross, *Technische Mechanik: Band 3: Kinetik*, 6te Ausg., Ser. Springer-Lehrbuch. Springer, Aug. 1999.
- A. J. Hawkins. (2023, Mai) GM's Ultra Cruise will use radar, camera, and lidar to enable hands-free driving. [Online]. Verfügbar: <https://www.theverge.com/2023/3/7/23627656/gm-ultra-cruise-sensor-radar-lidar-hands-free>
- F. Hecker, "Querführung," Universität der Bundeswehr München, Neubiberg, Interner Bericht UniBwM / LRT / TAS / IB 2007:5, Okt. 2007.

- M. Hecking. (2019, Jan.) Autonomes-Fahren-Start-up Aurora Innovations: In zwei Jahren auf zwei Milliarden Dollar. *Manager Magazin*. [Online]. Verfügbar: <https://www.manager-magazin.de/unternehmen/autoindustrie/volkswagen-partner-aurora-innovations-2-milliarden-wert-a-1247180-2.html>
- G. Heiming. (2021, Juli) iMUGS – Verbundeinsatz unbemannter Systeme und bemannter Einheiten. *Soldat & Technik*. [Online]. Verfügbar: <https://soldat-und-technik.de/2021/07/mobilitaet/27757/imugs-verbundeinsatz-unbemannter-systeme-und-bemannter-einheiten/>
- B. C. Heinrich, "Towards Smoother and More Precise Autonomous Driving," Dissertation, Universität der Bundeswehr München, Fakultät für Luft- und Raumfahrt-technik, 2023, to appear.
- B. C. Heinrich, D. Fassbender, und H.-J. Wuensche, "Precise Object-Relative Positioning for Car-Like Robots," in *Proceedings of IEEE Intelligent Transportation Systems Conference (ITSC)*, Rio de Janeiro, Brazil, Nov. 2016, S. 1720–1726.
- B. Heißing, M. Ersoy, und S. Gies, *Fahrwerkhandbuch: Grundlagen, Fahrdynamik, Komponenten, Systeme, Mechatronik, Perspektiven*, Ser. ATZ/MTZ-Fachbuch. Vieweg+Teubner Verlag, 2011. [Online]. Verfügbar: <http://books.google.de/books?id=kSKq5QTu2FgC>
- Hengstler GmbH, "RA 58-P with CAN-Bus (Layer 2): Technical Manual," Juli 1999. [Online]. Verfügbar: http://www.hengstler.com/gfx/file/download/archive/Datenblatt_RA58_CAN
- K. Henning, L. Petry, R. Ramakers, und J. Meinhold, "Sustainable transport – Knowledge and innovations at RWTH Aachen university for Europe's systems of tomorrow," in *Proceedings of IEEE International Conference on Systems, Man and Cybernetics (SMC)*, San Antonio, TX, USA, Okt. 2009, S. 2403–2408.
- M. Himmelsbach, "Multimodales Umgebungsmodell – Abschlussbericht Phase I 2012-2015," Universität der Bundeswehr München, Neubiberg, Tech. Ber. UniB-wM / LRT / TAS / TR 2015:10, Dez. 2015.
- M. Himmelsbach und H.-J. Wuensche, "Tracking and Classification of Arbitrary Objects with Bottom-Up/Top-Down Detection," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Alcalá de Henares, Spain, Juni 2012, S. 577–582.
- M. Himmelsbach, F. von Hundelshausen, und H.-J. Wuensche, "Fast Segmentation of 3D Point Clouds for Ground Vehicles," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, San Diego, CA, USA, Juni 2010.
- M. Himmelsbach, S. Schneider, und H.-J. Wuensche, "A Comparison of Error Metrics for Extrinsic Calibration and Fusion of Camera and Multi-Layer LIDAR," in *Proceedings of IASTED International Conference on Signal Processing, Pattern Recognition, and Applications (SPPRA)*, Innsbruck, Austria, Feb. 2011.

- O. Hoffmann. (2022, Nov.) Forschungsvorhaben InterRoC – Rheinmetall HX-LKW demonstrieren automatisierte Fahrfähigkeiten. Rheinmetall AG. Düsseldorf, Germany. [Online]. Verfügbar: https://www.rheinmetall.com/de/media/news-watch/news/2022/2022-11-08_forschungsvorhaben-interroc
- G. Honsel, "Der blinde Fleck," *MIT Technology Review*, Feb. 2023. [Online]. Verfügbar: https://www.mi.fu-berlin.de/inf/groups/ag-ki/news/TR_DerBlindeFleck.pdf
- J. Hooper. (2004, Juni) From Darpa Grand Challenge 2004: DARPA's Debacle in the Desert – Behind the scenes at the DARPA Grand Challenge, the 142-mile robot race that died at mile 7. Popular Science. [Online]. Verfügbar: <https://www.popsci.com/scitech/article/2004-06/darpa-grand-challenge-2004darpas-debacle-desert/>
- M. Horn, T. Wodtko, M. Buchholz, und K. Dietmayer, "Online Extrinsic Calibration Based on Per-Sensor Ego-Motion Using Dual Quaternions," *IEEE Robotics and Automation Letters*, Bd. 6, no. 2, S. 982–989, 2021.
- F. Hubik. (2019, Jan.) Vernetzte Trucks – Daimler beerdigt die Hoffnungstechnik Platooning. Handelsblatt.com. [Online]. Verfügbar: <https://www.handelsblatt.com/unternehmen/industrie/technik-ineffizient-vernetztes-trucks-daimler-beerdigt-die-hoffnungstechnik-platooning/23836178.html>
- . (2021, Sep.) Daimler und BMW sollten sich beim Rennen ums autonome Fahren auf deutsche Tugenden besinnen. Handelsblatt. [Online]. Verfügbar: <https://www.handelsblatt.com/meinung/kommentare/kommentar-daimler-und-bmw-sollten-sich-beim-rennen-ums-autonome-fahren-auf-deutsche-tugenden-besinnen/27607950.html>
- F. Hubik und J. Hofer. (2022, Feb.) Machtprobe in der Autoindustrie: Techkonzerne saugen die Gewinne von Mercedes, BMW und VW auf. Handelsblatt. [Online]. Verfügbar: <https://www.handelsblatt.com/unternehmen/industrie/nvidia-google-amazon-machtprobe-in-der-autoindustrie-techkonzerne-saugen-die-gewinne-von-mercedes-bmw-und-vw-auf/28072096.html>
- D. Hull. (2017, Okt.) The PayPal Mafia of Self-Driving Cars Has Been at It a Decade. Bloomberg. [Online]. Verfügbar: <https://www.bloomberg.com/news/features/2017-10-30/it-s-been-10-years-since-robots-proved-they-could-drive>
- IAV GmbH. (2020, Juni) HEAT is back: Autonomer Bus fährt wieder. Pressemitteilung. Berlin, Germany. [Online]. Verfügbar: <https://www.iav.com/news/heat-is-back-autonomer-bus-faehrt-wieder/>
- Ibeo AS. (2019, Jan.) Start des Projektes „AirPortMover“. Pressemitteilung. Ibeo Automotive Systems GmbH. Enge-Sande, Germany. [Online]. Verfügbar: <https://cdn.www.ibeo-as.com/6cee636caa79aa37148e391d81d7ff399756d0ba/Pressemitteilung-AirPortMover.pdf>
- Ibeo Automotive Systems GmbH, *Features of ibeo LUX 8L and ibeo LUX (model 2010)*, Hamburg, Juli 2012.

- G. Ismar. (2023, Mai) Die Bundeswehr hat ein akutes Personalproblem. *Süddeutsche Zeitung*. Berlin, Germany. [Online]. Verfügbar: <https://www.sueddeutsche.de/politik/bundeswehr-personalmangel-ideen-probleme-1.5834753>
- T. Jacob und R. Triebel, "Automatisierung im Bergbau der K+S Gruppe," in *Kali und Steinsalz*, no. 2. Kassel, Germany: Kaliverein e.V., 2002, S. 18–27.
- Jaguar Land Rover. (2016, Juli) Jaguar Land Rover Demonstrates All-Terrain Self-Driving Technology. Whitley, UK. [Online]. Verfügbar: <http://media.jaguarlandrover.com/news/2016/07/jaguar-land-rover-demonstrates-all-terrain-self-driving-technology>
- S. Jansen. (2016, Sep.) Volvo Construction Equipment erwartet auf der elektrischen Baustelle eine Verringerung der CO₂-Emissionen um bis zu 95 Prozent. Volvo Construction Equipment Germany GmbH. [Online]. Verfügbar: <https://www.volvoce.com/deutschland/de-de/about-us/news/2016/elektrische-baustellenloesung/>
- H. Jaspers, G. R. Mueller, und H.-J. Wuensche, "High Accuracy Model-Based Object Pose Estimation for Autonomous Recharging Applications," in *Proceedings of IEEE Winter Conference on Applications of Computer Vision (WACV)*, Lake Placid, NY, USA, März 2016.
- H. Jaspers, M. Himmelsbach, und H.-J. Wuensche, "Multi-modal Local Terrain Maps from Vision and LiDAR," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Redondo Beach, CA, USA, Juni 2017.
- T. Jochem, D. Pomerleau, B. Kumar, und J. Armstrong, "PANS: A Portable Navigation Platform," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Detroit, MI, USA, Sep. 1995, S. 107–112.
- S. J. Julier und J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, Bd. 92, no. 3, S. 401–422, März 2004.
- KADOMO GmbH, "Neues Steuersystem von KADOMO - Mit der CAN-Control-Steuerbox Fahrzeugfunktionen steuern," Pressemitteilung, Okt. 2010. [Online]. Verfügbar: <https://www.openpr.de/pdf/473904/Neues-Steuersystem-von-KADOMO-Mit-der-CAN-Control-Steuerbox-Fahrzeugfunktionen-steuern.pdf>
- J. Kallwies und H.-J. Wuensche, "Effective Combination of Vertical and Horizontal Stereo Vision," in *Proceedings of IEEE Winter Conference on Applications of Computer Vision (WACV)*, Lake Tahoe, NV/CA, USA, März 2018.
- J. Kallwies, B. Forkel, und H.-J. Wuensche, "Determining and Improving the Localization Accuracy of AprilTag Detection," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France (Virtual Conference), Juni 2020.
- R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Transactions of the ASME – Journal of Basic Engineering*,

- Bd. 82, no. Series D, S. 35–45, 1960. [Online]. Verfügbar: <https://www.cs.unc.edu/~welch/kalman/media/pdf/Kalman1960.pdf>
- K. Kaluski. (2016, Mai) Driverless Citroën DS from the 1960's. Hooniverse. [Online]. Verfügbar: <https://hooniverse.com/driverless-citroen-ds-from-the-1960s/>
- S. Kammel, J. Ziegler, B. Pitzer, M. Werling, T. Gindele, D. Jagzent, J. Schröder, M. Thuy, M. Goebel, F. von Hundelshausen, O. Pink, C. Frese, und C. Stiller, "Team AnnieWAY's autonomous system for the DARPA Urban Challenge 2007," *Journal of Field Robotics*, Bd. 25, no. 9, S. 615–639, 2008.
- N. Karamanov, D. Andreev, M. Pfeifle, H. Bock, M. Otto, und M. Schulze, "Map Line Interface for Autonomous Driving," in *Proceedings of IEEE Intelligent Transportation Systems Conference (ITSC)*, Maui, Hawaii, USA, Nov. 2018, S. 26–33. [Online]. Verfügbar: <https://www.visteon.com/wp-content/uploads/2019/01/map-line-interface-for-autonomous-driving.pdf>
- S. Kehl, "Querregelung eines Versuchsfahrzeugs entlang vorgegebener Bahnen," Dissertation, Universität Stuttgart, Fakultät Maschinenbau, 2007.
- D. Keilhoff, D. Niedballa, H.-C. Reuss, M. Buchholz, F. Gies, K. Dietmayer, M. Lauer, C. Stiller, S. Ackermann, H. Winner, A. Kampmann, B. Alrifaaee, S. Kowalewski, F. Klein, M. Struth, T. Woopen, und L. Eckstein, "UNICARagil – New architectures for disruptive vehicle concepts," in *19. Internationales Stuttgarter Symposium*, M. Bargende, H.-C. Reuss, A. Wagner, und J. Wiedemann, Hrsg. Wiesbaden: Springer Fachmedien Wiesbaden, 2019, S. 830–842.
- J. Keller. (2022, Nov.) Soldiers are finally getting a 'robotic mule' to haul all their heavy sh*t for them. Task & Purpose. [Online]. Verfügbar: <https://taskandpurpose.com/tech-tactics/army-s-met-robotic-mule-fielding/>
- K. Kilbon, "Tomorrow's Thruway is Here Today - A report on the RCA electronic highway system," *electronic age*, Bd. 19, no. 3, S. 26–29, 1960. [Online]. Verfügbar: <https://worldradiohistory.com/Archive-Radio-Age/ElectronicAge-1960-Autumn.pdf>
- Kistler Gruppe, "WPT Sensoren – Radimpulsgeber, Typ CWPTA," Datenblatt, 2020.
- K. Korosec. (2020, Dez.) Uber sells self-driving unit Uber ATG in deal that will push Aurora's valuation to \$10B. TechCrunch. [Online]. Verfügbar: <https://techcrunch.com/2020/12/07/uber-sells-self-driving-unit-uber-atg-in-deal-that-will-push-auroras-valuation-to-10b>
- Kraftfahrt-Bundesamt. (2023) Erprobungsgenehmigung. [Online]. Verfügbar: https://www.kba.de/DE/Themen/Typgenehmigung/Autonomes_automatisiertes_Fahren/Erprobungsgenehmigung/erprobungsgenehmigung_node.html
- S. Krempl. (2023, Juni) Autonomes Fahren: Mercedes "Drive Pilot" erhält Zulassung für Kalifornien. Heise Online. Hannover, Germany. [Online]. Verfügbar: <https://www.heise.de/news/Autonomes-Fahren-Mercedes-Drive-Pilot-erhaelt-Zulassung-fuer-Kalifornien-9182920.html>

- F. Kröger, "Fahrerlos und unfallfrei. Eine frühe automobiler Technikutopie und ihre populärkulturelle Bildgeschichte," in *Technology Fiction – Technische Visionen und Utopien in der Hochmoderne*, U. Fraunholz und A. Woschek, Hrsg. Bielefeld, Germany: Transcript Verlag, 2012. [Online]. Verfügbar: <https://fabiankroeger.com/2012/04/25/fahrerlos-und-unfallfrei-eine-fruhe-automobile-technikutopie-und-ihre-popularkulturelle-bildgeschichte/>
- F. Kröger, "Das automatisierte Fahren im gesellschaftsgeschichtlichen und kulturwissenschaftlichen Kontext," in *Autonomes Fahren: Technische, rechtliche und gesellschaftliche Aspekte*, M. Maurer, J. C. Gerdes, B. Lenz, und H. Winner, Hrsg. Berlin, Heidelberg, Germany: Springer Vieweg, 2015, S. 41–67.
- W. Kubinger und RelCon-Team, "RelCon: Reliable Control of Semi-Autonomous Platforms," in *6. KIRAS Fachtagung*, 2016. [Online]. Verfügbar: http://www.kiras.at/fileadmin/dateien/fachtagung16/02_RELCON.pdf
- K.-D. Kuhnert, "Zur Echtzeit-Bildfolgenanalyse mit Vorwissen," Dissertation, Universität der Bundeswehr München, Fakultät für Luft- und Raumfahrttechnik, Institut für Meßtechnik, Neubiberg, Germany, 1988.
- C. Kunkel. (2021, Aug.) Daimler und Bosch bauen doch keine Robotaxis. Süddeutsche Zeitung. [Online]. Verfügbar: <https://www.sueddeutsche.de/wirtschaft/bosch-daimler-athena-robotaxi-ende-1.5378806>
- H. Kunkel, "Unbemannter Materialtransport Land – Forderungslage der SKB," in *Symposium „Aspekte zur Technologie militärischer Fahrzeuge mit autonomen Fahrsystemen“*. Mannheim, Germany: Bildungszentrum der Bundeswehr, Juni 2019, Vortrag.
- R. Kunze, R. Ramakers, K. Henning, und S. Jeschke, "Organization and Operation of Electronically Coupled Truck Platoons on German Motorways," in *Intelligent Robotics and Applications*, M. Xie, Y. Xiong, C. Xiong, H. Liu, und Z. Hu, Hrsg. Berlin, Heidelberg: Springer, 2009, S. 135–146.
- M. Kusenbach, "Sensoren für autonome Fahrzeuge," in *Fortschrittsbericht MORE – Jahresbericht Dezember 2022 – Teilbereich Vernetzung & Autonomie*, C. Trapp, Hrsg. Neubiberg, Germany: Universität der Bundeswehr München, Munich Mobility Research Campus (MORE), Dez. 2022, Kap. 2.2.
- T. Kuther. (2017, Mai) Elektrisch angetrieben, autonom und ohne Fahrerhaus. Vogel / ElektronikPraxis. [Online]. Verfügbar: <https://www.elektronikpraxis.vogel.de/elektrisch-angetrieben-autonom-und-ohne-fahrerhaus-a-610383/>
- Landesamt für Digitalisierung, Breitband und Vermessung. (2022) Geländemodell: Ganz Bayern in 3D – vom Main bis zur Zugspitze. München, Germany. [Online]. Verfügbar: <https://www.ldbv.bayern.de/produkte/3dprodukte/gelaende.html>
- . (2022) Digitales Oberflächenmodell: Das Luftbild wird dreidimensional. München, Germany. [Online]. Verfügbar: <https://www.ldbv.bayern.de/produkte/3dprodukte/dom.html>

- . (2021) Orthophotos: Entzerrte und maßstabsgetreue Luftbilder. München, Germany. [Online]. Verfügbar: <https://www.ldbv.bayern.de/produkte/luftbild/orthophotos.html>
- C. Lee. (2019, Feb.) Autonomous Convoy Tech Moves Toward Official Program. National Defense. [Online]. Verfügbar: <https://www.nationaldefensemagazine.org/articles/2019/2/22/autonomous-convoy-tech-moves-toward-official-program>
- T. B. Lee. (2023, Juni) The "death of self-driving cars" has been greatly exaggerated. Ars Technica. [Online]. Verfügbar: <https://arstechnica.com/cars/2023/06/the-death-of-self-driving-cars-is-greatly-exaggerated/>
- Z. Li, Q. Zhu, und C. Gold, *Digital Terrain Modeling: Principles and Methodology*. Boca Raton, FL, USA: CRC Press, 2005.
- M. Linden. (2023, Mai) Porsche nimmt Kurs auf autonomes Fahren. Golem.de. [Online]. Verfügbar: <https://www.golem.de/news/kooperation-mit-mobileye-porsche-nimmt-kurs-auf-autonomes-fahren-2305-174039.html>
- C. Lindner. (2019, Aug.) A Cave Is No Place for Humans, So DARPA Is Sending In the Robots. Popular Mechanics. [Online]. Verfügbar: <https://www.popularmechanics.com/military/research/a28771417/darpa-subterranean-challenge/>
- M. Link. (2017) Bundesrat stimmt Gesetz für selbstfahrende Autos zu. Heise Online. Hannover, Germany. [Online]. Verfügbar: <https://heise.de/-3713452>
- R. Lohmann und J. Schneider. (2008) ParkShuttle. [Online]. Verfügbar: <http://faculty.washington.edu/jbs/itrans/Parkshuttle-update.doc>
- M. Lützeler, M. Maurer, und E. D. Dickmanns, "Ein kompaktes System zur visuellen Straßenerkennung," in *Mustererkennung 1996 - 18. DAGM-Symposium*, Ser. Informatik aktuell, B. Jähne, P. Geißler, H. Haußecker, und F. Hering, Hrsg., Heidelberg, Germany, 1996, S. 61–626.
- M. Manz, "Visuelle Erkennung und Verfolgung der Fahrspur unter Verwendung von Kalmanfilteralgorithmen und einer aktiven Kameraplattform," Universität der Bundeswehr München, Neubiberg, Interner Bericht UniBwM / LRT / TAS / IB 2008:2, Feb. 2008.
- , "Modellbasierte visuelle Wahrnehmung zur autonomen Fahrzeugführung," Dissertation, Universität der Bundeswehr München, Fakultät für Luft- und Raumfahrttechnik, Neubiberg, 2013. [Online]. Verfügbar: <https://athene-forschung.unibw.de/node?id=90737>
- M. Manz, F. von Hundelshausen, und H.-J. Wuensche, "A Hybrid Estimation Approach for Autonomous Dirt Road Following using Multiple Clothoid Segments," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, Anchorage, AK, USA, Mai 2010.
- J. Marek, "MEMS-Sensoren im Überblick," *AUTOMOBIL-ELEKTRONIK*, S. 16–18, Apr. 2007. [Online]. Verfügbar: https://web.archive.org/web/20130523061558/http://imperiam-verlag.de/imperia/md/upload/article/ael07_02_016.pdf

- J. Markoff, "Military Lags in Push for Robotic Ground Vehicles," *The New York Times*, Sep. 2013. [Online]. Verfügbar: <http://www.nytimes.com/2013/09/24/science/military-lags-in-push-for-robotic-ground-vehicles.html>
- J. Markoff und J. M. Broder. (2004, März) No Winner in a Desert Race for Robots. *The New York Times*. [Online]. Verfügbar: <https://www.nytimes.com/2004/03/14/us/no-winner-in-a-desert-race-for-robots.html>
- MathWorks. (2023) Visualizing Geoid Height for Earth Geopotential Model 1996. [Online]. Verfügbar: https://de.mathworks.com/help/aerotbx/ug/visualizing-geoid-height-for-earth-geopotential-model-1996.html?searchHighlight=ast2DGeoidPlot&s_tid=srchtitle_ast2DGeoidPlot_1
- R. Matthaei, A. Reschka, J. Rieken, F. Dierkes, S. Ulbrich, T. Winkle, und M. Maurer, "Autonomes Fahren," in *Handbuch Fahrerassistenzsysteme: Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort*, 3. Ausg., Ser. ATZ/MTZ-Fachbuch, H. Winner, S. Hakuli, F. Lotz, und C. Singer, Hrsg. Springer Vieweg, 2015, S. 1139–1166.
- M. Maurer, J. C. Gerdes, B. Lenz, und H. Winner, Hrsg., *Autonomes Fahren: Technische, rechtliche und gesellschaftliche Aspekte*. Berlin, Heidelberg, Germany: Springer Vieweg, 2015.
- maxon motor GmbH, "Programm 2023/2024: Hochpräzise Antriebe und Systeme," Onlinekatalog, 2023. [Online]. Verfügbar: <https://online.flippingbook.com/view/623537/>
- P. S. Maybeck, *Stochastic Models, Estimation, and Control*, Ser. Mathematics in Science and Engineering. London, UK: Academic Press, 1979, Bd. 141.
- P. McGee. (2019, Juni) Volkswagen breaks with Silicon Valley self-driving start-up Aurora. *Financial Times*. [Online]. Verfügbar: <https://www.ft.com/content/2885b3c6-8c60-11e9-a1c1-51bf8f989972>
- S. McKay, M. E. Boyer, N. M. Beyene, M. Lerario, M. W. Lewis, K. D. Stanley, R. Steeb, B. Wilson, und K. Giglio, *Automating Army Convoys: Technical and Tactical Risks and Opportunities*. Santa Monica, CA, USA: RAND Corporation, 2020. [Online]. Verfügbar: <https://apps.dtic.mil/sti/pdfs/AD1096337.pdf>
- Mercedes-Benz Museum GmbH. (2016, Sep.) Das Projekt PROMETHEUS ab 1986: Vorreiter des autonomen Fahrens. Pressemitteilung. Stuttgart, Germany. [Online]. Verfügbar: <https://media.mercedes-benz.com/article/b7bf109a-260a-4031-91b4-9d32830dfd49>
- Mercedes-Benz Research & Development North America, Inc., "Introducing DRIVE PILOT: An Automated Driving System for the Highway," Sunnyvale, CA, USA, Feb. 2019. [Online]. Verfügbar: <https://group.mercedes-benz.com/documents/innovation/other/2019-02-20-vssa-mercedes-benz-drive-pilot-a.pdf>

- F. Mertens. (2018, Juli) Mit dem People Mover in die Zukunft. autogazette. [Online]. Verfügbar: <https://www.autogazette.de/zf/mover/sicherheit/mit-dem-people-mover-in-die-zukunft-der-mobilitaet-989387441.html>
- R. van der Merwe, "Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models," Ph.D. dissertation, OGI School of Science & Engineering, Oregon Health & Science University, Portland, OR, USA, Apr. 2004. [Online]. Verfügbar: <https://scholararchive.ohsu.edu/concern/etds/v405s9361?locale=de>
- C. Meyer. (2000, März) VW zeigt autonomen autofahrenden Roboter. Heise Online. Hannover, Germany. [Online]. Verfügbar: <https://www.heise.de/news/VW-zeigt-autonomen-autofahrenden-Roboter-23011.html>
- M. Mitschke und H. Wallentowitz, *Dynamik der Kraftfahrzeuge*, 5te Ausg., Ser. VDI-Buch. Springer Vieweg, 2014. [Online]. Verfügbar: <https://link.springer.com/book/10.1007/978-3-658-05068-9>
- M. Montemerlo, J. Becker, S. Bhat, H. Dahlkamp, D. Dolgov, S. Ettinger, D. Haehnel, T. Hilden, G. Hoffmann, B. Huhnke, D. Johnston, S. Klumpp, D. Langer, A. Levandowski, J. Levinson, J. Marcil, D. Orenstein, J. Paefgen, I. Penny, A. Petrovskaya, M. Pflueger, G. Stanek, D. Stavens, A. Vogt, und S. Thrun, "Junior: The Stanford Entry in the Urban Challenge," *Journal of Field Robotics*, Bd. 25, no. 8, S. 467–492, 2008.
- M. Monticello. (2020, Okt.) Cadillac's Super Cruise Outperforms Other Driving Assistance Systems. Consumer Reports. [Online]. Verfügbar: <https://www.consumerreports.org/cars/car-safety/cadillac-super-cruise-outperforms-other-active-driving-assistance-systems-a1113486809/>
- D. C. Moore, A. S. Huang, M. Walter, E. Olson, L. Fletcher, J. Leonard, und S. Teller, "Simultaneous Local and Global State Estimation for Robotic Navigation," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, Kobe, Japan, 2009, S. 3794–3799.
- M. Mörbe, "Fahrndynamik-Sensoren für FAS," in *Handbuch Fahrerassistenzsysteme: Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort*, 3. Ausg., Ser. ATZ/MTZ-Fachbuch, H. Winner, S. Hakuli, F. Lotz, und C. Singer, Hrsg. Springer Vieweg, 2015, S. 223–242.
- MSO Meßtechnik und Ortung GmbH, "Bedienungsanleitung Speed Wedge Software 044 Serie 1.11 Gehäuse Neu," online <http://www.mso-technik.de/sensor/speedwedge.html>, 2011.
- N. Müller, "Autonomes Manövrieren und Navigieren mit einem sehenden Fahrzeug," Dissertation, Universität der Bundeswehr München, Fakultät für Luft- und Raumfahrttechnik, Institut für Systemdynamik und Flugmechanik, Neubiberg, 1996, also as Fortschritt-Berichte VDI Reihe 12 Verkehrstechnik/Fahrzeugtechnik, Nr. 281, VDI-Verlag, Düsseldorf, 1996.

- Münchner Verkehrs- und Tarifverbund GmbH, "Netzplan S-Bahn, U-Bahn, Regionalzug und Tram im MVV," Dez. 2018. [Online]. Verfügbar: https://www.mvv-muenchen.de/fileadmin/mediapool/03-Plaene_Bahnhoefe/Netzplaene/MVV_Netzplan_S_U_R_T.pdf
- M. Münster, M. Brost, T. Siefkes, G. Kopp, E. Beeh, F. Rinderknecht, S. Schmid, M. Osebek, S. Scheibe, R. Hahn, D. Heyner, P. Klein, G. Piazza, C. Ulrich, W. Kraft, F. Philipps, L. Köhler, M. Buchholz, T. Wodtke, K. Dietmayer, M. Frey, F. Weitz, F. Gauterin, H. Stoll, M. Schindewolf, H. Guissouma, F. Krauter, E. Sax, J. Neubeck, S. Müller, S. Eberts, M. Göldner, S. Teichmann, J. Kiebler, M. Saljanin, M. Bargende, und A. Wagner, "U-shift ii vision and project goals," in *22. Internationales Stuttgarter Symposium*, M. Bargende, H.-C. Reuss, und A. Wagner, Hrsg. Wiesbaden: Springer Fachmedien Wiesbaden, 2022, S. 18–31.
- M. M. Muntzinger, "Zustandsschätzung mit chronologisch ungeordneten Sensordaten für die Fahrzeugumfelderfassung," Dissertation, Universität Ulm, Nov. 2011.
- B. Mysliwetz, "Parallelrechner-basierte Bildfolgen-Interpretation zur autonomen Fahrzeugführung," Dissertation, Universität der Bundeswehr München, Fakultät für Luft- und Raumfahrttechnik, Institut für Systemdynamik und Flugmechanik, Neubiberg, 1990.
- B. Naujoks und H.-J. Wuensche, "An Orientation Corrected Bounding Box Fit Based on the Convex Hull under Real Time Constraints," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Changshu, Suzhou, China, Juni 2018.
- B. Naujoks, P. Burger, und H.-J. Wuensche, "The Greedy Dirichlet Process Filter – An Online Clustering Multi-Target Tracker," in *Proceedings of IEEE Global Conference on Signal and Information Processing (GLOBALSIP)*, Anaheim, CA, USA, Nov. 2018.
- , "Fast 3D Extended Target Tracking using NURBS Surfaces," in *Proceedings of IEEE Intelligent Transportation Systems Conference (ITSC)*, Auckland, New Zealand, Okt. 2019.
- Nerian Vision Technologies, *SP1 Stereo Vision System – User Manual*, Leinfelden-Echterdingen, Germany, Feb. 2017, v1.12. [Online]. Verfügbar: https://nerian.de/nerian-content/downloads/manuals/sp1/sp1_manual_v1_12.pdf
- K. Nordenbrock. (2022, Nov.) Waymo darf in San Francisco komplett ohne Fahrer fahren. t3n. [Online]. Verfügbar: <https://t3n.de/news/waymo-darf-in-san-francisco-komplett-ohne-fahrer-fahren-1514582/>
- P. Olsen. (2018, Okt.) Cadillac Tops Tesla in Consumer Reports' First Ranking of Automated Driving Systems. Consumer Reports. [Online]. Verfügbar: <https://www.consumerreports.org/autonomous-driving/cadillac-tops-tesla-in-automated-systems-ranking/>
- OpenStreetMap. (o.J.) Openstreetmap project homepage. [Online]. Verfügbar: www.openstreetmap.org

- Ouster, Inc., *Datasheet – OS0 Ultra-Wide View High-Resolution Imaging Lidar*, San Francisco, CA, USA, Juli 2020, rev 04/05/23 for Hardware Rev C. [Online]. Verfügbar: <https://data.ouster.io/downloads/datasheets/datasheet-revc-v2p5-os0.pdf>
- Oxford Technical Solutions Ltd., *RT GNSS-aided inertial measurement systems - User Manual - Covers RT3000 v3 and RT500 v1 models*, Middleton Stoney, Oxfordshire, Großbritannien, Feb. 2020, document Revision 200220. [Online]. Verfügbar: <https://www.oxts.com/wp-content/uploads/2020/03/rتمان-200302.pdf>
- H. B. Pacejka und I. J. M. Besselink, “Magic Formula Tyre Model with Transient Properties,” *Vehicle System Dynamics*, Bd. 27, no. sup001, S. 234–249, Aug. 1997.
- J. Pan und P. Sotiriadis, “Near Field Monitoring System MORE (NeFiMMore),” in *Fortschrittsbericht MORE – Jahresbericht Dezember 2022 – Teilbereich Vernetzung & Autonomie*, C. Trapp, Hrsg. Neubiberg, Germany: Universität der Bundeswehr München, Munich Mobility Research Campus (MORE), Dez. 2022, Kap. 2.11.
- J. Pan, P. Sotiriadis, und F. Englberger, “Collision Avoidance of Autonomous Driving at Low Speed in the Near Field of Vehicle,” *International Journal of Emerging Engineering and Technology (IJEET)*, Bd. 2, no. 1, S. 57–62, 2023.
- J. Pander. (2012, Juni) Projekt 'Sartre': Hände weg vom Steuer! Spiegel Online. [Online]. Verfügbar: <https://www.spiegel.de/auto/aktuell/sartre-projekt-autonome-autokolonne-auf-oeffentlicher-autobahn-a-837307.html>
- M. Parent und J.-M. Blosseville, “Automated Vehicles in Cities: A First Step Towards the Automated Highway,” in *Future Transportation Technology Conference and Exposition*. Costa Mesa, CA, USA: SAE Technical Paper Series, Aug. 1998.
- J. Pellenz, A. Retterath, und A. Volk, “Robotik und automatisierte Systeme in der Bundeswehr,” *Mittler Report Verlag*, Bd. 2/2021, S. 36–39, Juni 2021. [Online]. Verfügbar: <https://www.blauer-bund.de/robotik-und-automatisierte-systeme-in-der-bundeswehr/>
- M. Pellkofer, U. Hofmann, und E. D. Dickmanns, “Autonomous cross country driving using Active Vision,” in *Intelligent Robots and Computer Vision XXI: Algorithms, Techniques, and Active Vision*, D. P. Casasent, E. L. Hall, und J. Roning, Hrsg., Bd. 5267, International Society for Optics and Photonics. SPIE, 2003, S. 78–89.
- F. Pertschy. (2023, Mai) Apex.AI hilft Volkswagen bei autonomen Shuttles. *automotiveIT*. [Online]. Verfügbar: <https://www.automotiveit.eu/technology/apexai-hilft-volkswagen-bei-autonomen-shuttles-212.html>
- D. Petereit. (2023, Jan.) Probleme mit autonomen Taxis: San Francisco will Waymo und Cruise bremsen. *t3n*. [Online]. Verfügbar: <https://t3n.de/news/robotaxis-san-francisco-waymo-cruise-bremsen-1530949/>

- F. Pillau. (2023, Apr.) Continental plant mit Aurora "Hardware-as-a-Service" für autonome Lkw in den USA. Heise Autos. [Online]. Verfügbar: <https://www.heise.de/news/Continental-plant-mit-Aurora-Hardware-as-a-Service-fuer-autonome-Lkw-in-den-USA-8980610.html>
- . (2023, Jan.) Autonomes Auto: Mercedes "Drive Pilot" erstes Level 3-System mit US-Zertifikat. Heise Autos. [Online]. Verfügbar: <https://www.heise.de/news/Autonomes-Auto-Mercedes-Drive-Pilot-erstes-Level-3-System-mit-US-Zertifikat-7472566.html>
- R. Polzer und F. Huber, "Forschungsprojekt KIRAS," *MILIZ info - Informationen für Angehörige der Einsatzorganisation des Bundesheeres*, no. 1, S. 16, März 2013. [Online]. Verfügbar: https://www.bundesheer.at/pdf_pool/milizinfo/ausgabe0113.pdf
- D. Pomerleau, "RALPH: Rapidly Adapting Lateral Position Handler," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Detroit, MI, USA, Sep. 1995, S. 506–511.
- D. Pomerleau und T. Jochem. (1995) No Hands Across America. Carnegie Mellon University. Pittsburgh, PA, USA. Project Website. [Online]. Verfügbar: https://www.cs.cmu.edu/~tjochem/nhaa/nhaa_home_page.html
- S. Prawitz. (2022, Feb.) Jaguar Land Rover kooperiert mit Nvidia. Vogel / Automobil Industrie. [Online]. Verfügbar: <https://www.automobil-industrie.vogel.de/jaguar-land-rover-kooperiert-mit-nvidia-a-1096255/>
- A. Pronobis und R. P. N. Rao, "Learning Deep Generative Spatial Models for Mobile Robots," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vancouver, BC, Canada, Sep. 2017, S. 755–762.
- F. W. Rauskolb, K. Berger, C. Lipski, M. Magnor, K. Cornelsen, J. Effertz, T. Form, F. Graefe, S. Ohl, W. Schumacher, J. Wille, P. Hecker, T. Nothdurft, M. Doering, K. Homeier, J. Morgenroth, L. Wolf, C. Basarke, C. Berger, T. Gülke, F. Klose, und B. Rumpe, "Caroline: An Autonomously Driving Vehicle for Urban Environments," *Journal of Field Robotics*, Bd. 25, no. 9, S. 674–724, 2008.
- Red Team Robot Racing. (2006) Red Team – Carnegie Mellon. Robotics Institute, Carnegie Mellon University. Pittsburg, PA, USA. Archivierte Version. [Online]. Verfügbar: <https://web.archive.org/web/20070825065456/http://www.redteamracing.org/index.cfm>
- A. Reich und H.-J. Wuensche, "Fast Detection of Moving Traffic Participants in LiDAR Point Clouds by using Particles augmented with Free Space Information," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Kyoto, Japan, Okt. 2022.
- J. Reichel. (2020, Okt.) EasyMile setzt Meilenstein: Erstmals autonom auf Level 4 unterwegs. VISION mobility. [Online]. Verfügbar: <https://vision-mobility.de/news/easymile-setzt-meilenstein-erstmals-autonom-auf-level-4-unterwegs-71098.html>

- . (2020, Juni) Autonomes Fahren: Daimler dockt jetzt bei Nvidia an – bis Level 4. VISION mobility. [Online]. Verfügbar: <https://vision-mobility.de/news/autonomes-fahren-daimler-dockt-jetzt-bei-nvidia-an-bis-level-4-51930.html>
- I. Reichert, "Aufbau und Inbetriebnahme einer Testumgebung für Manöver eines Autonomen Kraftfahrzeugs," Diplomarbeit UniBwM / LRT / TAS / DA 2010:2, Universität der Bundeswehr München, Neubiberg, März 2010.
- , "Entwurf eines Positionsreglers für ein Autonomes Kraftfahrzeug," Studienarbeit UniBwM / LRT / TAS / SA 2009:2, Universität der Bundeswehr München, Neubiberg, Juli 2009.
- S. Reimann, "Steuern wie von Geisterhand: Automatisierung von Muldenkippern macht Rohstoffabbau sicherer und effizienter," *Deutsches Baublatt*, no. 394, S. 11, Okt. 2017. [Online]. Verfügbar: <http://www.baublatt.de/wp-content/uploads/2017/08/BB-394.pdf>
- F. Rheaume und A. R. Benaskeur, "Forward Prediction-based Approach to Target-Tracking with Out-of-Sequence Measurements," in *Proceedings of IEEE Conference on Decision and Control (CDC)*, Cancun, Mexico, Dez. 2008.
- P. Riekert und T. E. Schunck, "Zur Fahrmechanik des gummibereiften Kraftfahrzeugs," *Ingenieur-Archiv*, Bd. 11, no. 3, S. 210–224, Juni 1940.
- P. E. Ross, "The Audi A8: the World's First Production Car to Achieve Level 3 Autonomy," *IEEE Spectrum*, Juli 2017. [Online]. Verfügbar: <https://spectrum.ieee.org/cars-that-think/transportation/self-driving/the-audi-a8-the-worlds-first-production-car-to-achieve-level-3-autonomy>
- F. Rowsome Jr., "Educated Gas Pedal Keeps the Cops Away," in *Popular Science*, V. Torrey, Hrsg. New York, NY, USA: Godfrey Hammond, Jan. 1954, Bd. 164, no. 1, S. 164–169,264.
- SAE, "J3016_202104: Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles," Apr. 2021. [Online]. Verfügbar: https://www.sae.org/standards/content/j3016_202104/
- SAE International, "Surface Vehicle Recommended Practice: Vehicle Application Layer," März 2011, j1939/71_201103. [Online]. Verfügbar: https://forge.codesys.com/forge/talk/CODESYS-V2/thread/e3add933b3/dbf2/attachment/SAEJ1939_71.pdf
- Sandvik AB. (2018, Sep.) Twenty years of self-driving without a single accident involving people. Sandvik AB. Stockholm, Sweden. [Online]. Verfügbar: <https://www.home.sandvik/en/news-and-media/newslist/news/2018/09/twenty-years-of-self-driving-without-a-single-accident-involving-people/>
- R. Schesswendter. (2023, Feb.) Cariad: Bei VW soll Software jetzt die erste Geige spielen. t3n. [Online]. Verfügbar: <https://t3n.de/news/cariad-software-vw-1532353/>

- H. Schmidt. (2019, Dez.) Modulares Fahrzeugsystem nähert sich der Serienfertigung. *Neue Züricher Zeitung*. Zürich, Switzerland. [Online]. Verfügbar: <https://www.nzz.ch/mobilitaet/auto-mobil/rinspeed-metro-snap-2020-weiterentwickeltes-baukastensystem-ld.1528723>
- . (2020, Apr.) Audi steckt beim autonomen Fahren zurück. *Neue Züricher Zeitung*. [Online]. Verfügbar: <https://www.nzz.ch/mobilitaet/auto-mobil/autonomes-fahren-stufe-3-audi-verzichtet-im-a8-auf-staupilot-ld.1553933>
- F. E. Schneider, D. Wildermuth, und H.-L. Wolf, “ELROB and EURATHLON: Improving Search & Rescue Robotics through Real-World Robot Competitions,” in *10th International Workshop on Robot Motion and Control (RoMoCo)*, Juli 2015, S. 118–123.
- S. Schneider, “Fusion von Kamera und LiDAR zur modellbasierten autonomen Navigation,” Dissertation, Universität der Bundeswehr München, Fakultät für Luft- und Raumfahrttechnik, Neubiberg, Apr. 2023. [Online]. Verfügbar: <https://athene-forschung.unibw.de/144973>
- S. Schneider und H.-J. Wünsche, “Evaluierung von ROS als Middleware: Abschlussbericht E E810 BF127 Robotik Middleware ROS,” Universität der Bundeswehr München, Neubiberg, Tech. Ber. UniBwM / LRT / TAS / TR 2013:1, Jan. 2013.
- A. Schütz, D. E. Sánchez-Morales, und T. Pany, “Precise positioning through a loosely-coupled sensor fusion of GNSS-RTK, INS and LiDAR for autonomous driving,” in *IEEE/ION Position, Location and Navigation Symposium (PLANS)*, 2020, S. 219–225.
- A. Schwarzenegger, D. E. Bonner, und G. Valverde, *California Driver Handbook*. State of California, Department of Motor Vehicles, 2007. [Online]. Verfügbar: <https://www.csac-eia.org/services/risk-control/toolbox/best-practices-library/motor-vehicle-driving/2006-ca-driver-handbook-california-dmv/>
- D. Shapiro. (2022, Nov.) New Volvo EX90 SUV Heralds AI Era for Swedish Automaker, Built on NVIDIA DRIVE. Nvidia. [Online]. Verfügbar: <https://blogs.nvidia.com/blog/2022/11/09/volvo-ex90-suv-ai-nvidia-drive/>
- S. E. Shladover, “PATH at 20 – History and Major Milestones,” *IEEE Transactions on Intelligent Transportation Systems*, S. 584–592, Dez. 2007.
- K.-H. Siedersberger, “Komponenten zur automatischen Fahrzeugführung in sehenden (semi-)automomen Fahrzeugen,” Dissertation, Universität der Bundeswehr München, Fakultät für Luft- und Raumfahrttechnik, Institut für Systemdynamik und Flugmechanik, Neubiberg, 2003. [Online]. Verfügbar: <http://d-nb.info/972053271/34>
- G. G. Slabaugh, “Computing Euler angles from a rotation matrix,” 1999. [Online]. Verfügbar: <http://www.soi.city.ac.uk/~sbbh653/publications/euler.pdf>

- Smart Microwave Sensors GmbH, *Automotive Sensor UMRR-11 Type 132 – Data Sheet*, Braunschweig, Germany, Jan. 2021. [Online]. Verfügbar: https://www.smartmicro.com/fileadmin/media/Downloads/Automotive_Radar/Sensor_Data_Sheets_76-81GHz/UMRR-11_Type_132_Automotive_Data_Sheet.pdf
- , *Automotive Sensor UMRR-96 Type 153 – Data Sheet*, Braunschweig, Germany, Nov. 2020. [Online]. Verfügbar: https://www.smartmicro.com/fileadmin/media/Downloads/Automotive_Radar/Sensor_Data_Sheets_76-81GHz/UMRR_Automotive_Type_153_Data_Sheet.pdf
- SMG Technologie GmbH, “LUXACT 1D Compact - Technical Data Sheet,” Apr. 2018, v1.4. [Online]. Verfügbar: https://www.imc-tm.de/fileadmin/Public/Downloads/Datasheets/Luxact/TD_LUXACT_1D_Compact.pdf
- M. Smolarek, *Digital Infantry Battlefield Solution. Research and Innovation: DIBS project. Part III*. Milrem Robotics, März 2019, Kap. Germany, S. 61–68. [Online]. Verfügbar: <https://depot.ceon.pl/bitstream/handle/123456789/17122/digital-infantry-battlefield-solution-research-and-innovation-part-three-Germany.pdf?sequence=1&isAllowed=y>
- D. A. Sokolov. (2020, Aug.) Autonome Güterbeförderung kommt schneller als Robotaxis. Heise Online. Hannover, Germany. [Online]. Verfügbar: <https://www.heise.de/newsticker/meldung/Autonome-Gueterbefoerderung-kommt-schneller-als-Robotaxis-4882441.html>
- Stanford Racing. (2005) Stanford Racing. Stanford University. [Online]. Verfügbar: <https://cs.stanford.edu/group/roadrunner//old/index.html>
- Stanford Racing Team. (2007) Stanford Racing Team. Stanford University. [Online]. Verfügbar: <https://cs.stanford.edu/group/roadrunner/>
- W. Steingrube, “Quantitative Erfassung, Analyse und Darstellung des Ist-Zustandes,” in *Methoden und Instrumente räumlicher Planung*, E.-H. Ritter und K. Wolf, Hrsg. Akademie für Raumforschung und Landesplanung, 1998, S. 67–94.
- N. Steinhardt und S. Leinen, “Datenfusion für die präzise Lokalisierung,” in *Handbuch Fahrerassistenzsysteme: Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort*, 3. Ausg., Ser. ATZ/MTZ-Fachbuch, H. Winner, S. Hakuli, F. Lotz, und C. Singer, Hrsg. Springer Vieweg, 2015, S. 481–512.
- StreetScooter, Department of Autonomous Logistics. (2020, Mai) A Story of Autonomous Logistics: From rapid robot prototyping to pre-series robot production. ROS-Industrial Blog. Guest Article. [Online]. Verfügbar: <https://rosindustrial.org/news/2020/5/11/guest-article-on-the-story-of-the-autonomous-logistics>
- G. Struck, J. Geisler, F. Laubenstein, H.-H. Nagel, und G. Siegle, “Interaction Between Digital Road Map Systems And Trinocular Autonomous Driving,” in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Juli 1993, S. 461–466.

- S. Strygulec, D. Müller, M. Meuter, C. Nunn, S. Ghosh, und C. Wöhler, "Road Boundary Detection and Tracking Using Monochrome Camera Images," in *Proceedings of International Conference on Information Fusion (FUSION)*, 2013, S. 864–870.
- K. Stumpe. (2016, Nov.) Für deutsche Unternehmen keine Option: Japanische Firma entwickelt Muldenkipper ohne Führerhaus. Neue Osnabrücker Zeitung. Osnabrück, Germany. [Online]. Verfügbar: <https://www.noz.de/deutschland-welt/digitale-welt/artikel/811017/japanische-firma-entwickelt-muldenkipper-ohne-fuehrerhaus>
- T. Sugimachi, T. Fukao, Y. Suzuki, und H. Kawashima, "Development of Autonomous Platooning System for Heavy-duty Trucks," in *7th IFAC Symposium on Advances in Automotive Control*, 2013.
- Y. Tadjeh. (2022, Jan.) Army Sees Progress with Leader-Follower Vehicle Technology. National Defense - NDIA's Business and Technology Magazine. Arlington, VA, USA. [Online]. Verfügbar: <https://www.nationaldefensemagazine.org/articles/2022/1/21/army-sees-progress-with-leader-follower-vehicle-technology>
- Talend Germany GmbH. (2022) Was ist Middleware? Der Vermittler zwischen Ihren Systemen. Talend Germany GmbH. Bonn, Germany. [Online]. Verfügbar: <https://www.talend.com/de/resources/what-is-middleware>
- Tartan Racing. (2007) Tartan Racing – Carnegie Mellon. Field Robotics Center, Carnegie Mellon University. Pittsburg, PA, USA. [Online]. Verfügbar: <http://www.tartanracing.org/>
- Ö. S. Tas, N. O. Salscheider, F. Poggenhans, S. Wirges, C. Bandera, M. R. Zofka, T. Strauss, J. M. Zöllner, und C. Stiller, "Making Bertha Cooperate – Team AnnieWAY's Entry to the 2016 Grand Cooperative Driving Challenge," *IEEE Transactions on Intelligent Transportation Systems*, Bd. 19, no. 4, S. 1262–1276, 2018.
- Team Jefferson. (2007) Team Jefferson 2007 DARPA Urban Challenge Debrief. [Online]. Verfügbar: <https://web.archive.org/web/20150211202449/http://teamjefferson.com/wp-content/uploads/2014/06/Debrief-Complete-TeamJefferson-DRAFT.pdf>
- Team Victor Tango. (2007, Nov.) Victor Tango – Urban Challenge. Archivierte Version. [Online]. Verfügbar: <https://web.archive.org/web/20110304053755/http://www.me.vt.edu/urbanchallenge/>
- R. R. Teetor, "Speed control device for resisting operation of the accelerator," US Patent US 2519859, 1950.
- The Institute of Navigation. (2023) The ION Autonomous Snowplow Competition. Project Website. The Institute of Navigation, Inc. [Online]. Verfügbar: <http://www.autosnowplow.com/welcome.html>

- Thelker Geoinformation. (2005, Juli) GeolnFOReg für Microsoft Access. Access-Datei. Amt für Geoinformationswesen der Bundeswehr, Dezernat Geoinformatik. V. 1.3.40.
- C. Thompson, "Uber Would Like to Buy Your Robotics Department," *The New York Times Magazine*, Sep. 2015. [Online]. Verfügbar: <https://www.nytimes.com/2015/09/13/magazine/uber-would-like-to-buy-your-robotics-department.html>
- C. E. C. Thorpe, M. Hebert, T. Kanade, und S. Shafer, "Toward Autonomous Driving: The CMU Navlab. Part I: Perception," *IEEE Expert*, Bd. 6, no. 1, S. 31–42, Aug. 1991.
- , "Toward Autonomous Driving: The CMU Navlab. Part II: System and Architecture," *IEEE Expert*, Bd. 6, no. 1, S. 44–52, Aug. 1991.
- S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. van Niekerk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian, und P. Mahoney, "Stanley: The robot that won the DARPA Grand Challenge: Research Articles," *Journal of Robotic Systems*, Bd. 23, no. 9, S. 661–692, 2006.
- S. Thrun, "Learning metric-topological maps for indoor mobile robot navigation," *Artificial Intelligence*, Bd. 99, no. 1, S. 21–71, Feb. 1998.
- , "Robotic Mapping: A Survey," in *Exploring Artificial Intelligence in the New Millenium*, G. Lakemeyer und B. Nebel, Hrsg. Morgan Kaufmann, 2002.
- S. Thrun, W. Burgard, und D. Fox, *Probabilistic Robotics*. Cambridge, MA: MIT Press, 2005.
- TLD Group. (2018, März) Die Einführung automatisierter Fahrzeuge. Pressemitteilung. [Online]. Verfügbar: <https://www.tld-group.com/de/nachrichten/autonome-fahrzeuge/>
- P. G. Trepagnier, J. Nagel, P. M. Kinney, C. Koutsougeras, und M. Dooner, "KAT-5: Robust Systems for Autonomous Vehicle Navigation in Challenging and Unknown Terrain," *Journal of Field Robotics*, Bd. 23, no. 8, S. 509–526, Aug. 2006.
- S. Tsugawa, "Vision-Based Vehicles in Japan: Machine Vision Svstems and Driving Control Svstems," *IEEE Transactions on Industrial Electronics*, Bd. 41, no. 4, S. 398–405, Aug. 1994.
- S. Tsugawa, S. Jeschke, und S. E. Shladovers, "A Review of Truck Platooning Projects for Energy Savings," *IEEE Transactions on Intelligent Vehicles*, Bd. 1, no. 1, S. 68–77, März 2016.
- R. Tyborski und F. Hubik. (2021, Sep.) Warum VW, Daimler und BMW beim automatisierten Fahren erfolgreicher als die Tech-Konzerne sind. Handelsblatt. [Online]. Verfügbar: <https://www.handelsblatt.com/unternehmen/industrie/mobilitaet->

- warum-vw-daimler-und-bmw-beim-automatisierten-fahren-erfolgreicher-als-die-tech-konzerne-sind/27605860.html
- Umweltbundesamt, "Batterien und Akkus: Ihre Fragen – unsere Antworten zu Batterien, Akkus und Umwelt," Okt. 2012. [Online]. Verfügbar: <https://www.umweltbundesamt.de/sites/default/files/medien/publikation/long/4414.pdf>
- A. Unterholzner, "Sensor Orientation Selection and Adaptive Control of an Actuated Sensor Platform for Autonomous Vehicles," Dissertation, Universität der Bundeswehr München, Fakultät für Luft- und Raumfahrttechnik, Neubiberg, Apr. 2016. [Online]. Verfügbar: <http://athene-forschung.unibw.de/node?id=111301>
- A. Unterholzner und H.-J. Wuensche, "Adaptive State Space Control of a Camera Platform for an Autonomous Ground Vehicle," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Xi'an, China, Juni 2009, S. 591–596.
- , "Hybrid Adaptive Control of a Multi-Focal Vision System," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, San Diego, CA, USA, Juni 2010, S. 534–539.
- , "Adaptive Control of an Active Multi-Focal Vision System," in *Encyclopedia of Sustainability Science and Technology*, R. A. Meyers, Hrsg. Berlin, Germany: Springer, Juli 2012.
- A. Unterholzner, M. Rohland, M. Schweitzer, und H.-J. Wuensche, "Vision Based Online-Calibration of Inertial Gaze Stabilization," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, San Diego, CA, USA, Juni 2010, S. 646–651.
- C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M. N. Clark, J. Dolan, D. Duggins, T. Galatali, C. Geyer, M. Gittleman, S. Harbaugh, M. Hebert, T. M. Howard, S. Kolski, A. Kelly, M. Likhachev, M. McNaughton, N. Miller, K. Peterson, B. Pilnick, R. Rajkumar, P. Rybski, B. Salesky, Y.-W. Seo, S. Singh, J. Snider, A. Stentz, W. R. Whittaker, Z. Wolkowicki, J. Zigar, H. Bae, T. Brown, D. Demitrish, B. Litkouhi, J. Nickolaou, V. Sadekar, W. Zhang, J. Struble, M. Taylor, M. Darms, und D. Ferguson, "Autonomous Driving In Urban Environments: Boss and the Urban Challenge," *Journal of Field Robotics*, Bd. 25, no. 1, S. 425–466, Juni 2008.
- C. Urmson, J. Anhalt, D. Bartz, M. Clark, T. Galatali, A. Gutierrez, S. Harbaugh, J. Johnston, H. Kato, P. L. Koon, W. Messner, N. Miller, A. Mosher, K. Peterson, C. Ragusa, D. Ray, B. K. Smith, J. M. Snider, S. Spiker, J. C. S. A. Zigar, und W. R. L. Whittaker, "A Robust Approach to High-Speed Navigation for Unrehearsed Desert Terrain," *Journal of Field Robotics*, Bd. 23, no. 8, S. 467–508, Aug. 2006.
- Valtra Inc. (2019) The runway snowbot: Autonomous airport snow clearing – from vision to a proof of concept. Project Website. Valtra Inc. Suolahti, Finland. [Online]. Verfügbar: <https://www.runwaysnowbot.com/>
- C. C. van Helsdingen, J. O. Ten Cate, P. H. F. Peteri, und F. van der Heyden, "System for Navigation a Free Ranging Vehicle," US Patent 4 847 773, 1989.

- Velodyne Acoustics, Inc., *HDL-64E – User’s Manual*, Morgan Hill, CA, USA, Mai 2011, 63-HDL-64E Rev D. [Online]. Verfügbar: <https://www.manualslib.com/manual/893833/Velodyne-Hdl-64e.html>
- Velodyne LiDAR, Inc., *Velodyne HDL-32E User Manual*, Morgan Hill, CA, USA, Apr. 2017, 63-9113 Rev. M. [Online]. Verfügbar: <https://velodynelidar.com/wp-content/uploads/2019/12/63-9113-REV-M-MANUALUSERSHDL32E.pdf>
- , *HDL-64E S2 – User’s Manual and Programming Guide*, Morgan Hill, CA, USA, Mai 2011, 63HDL64E S2 Rev. D. [Online]. Verfügbar: <http://www.manualsdir.com/manuals/219159/velodyne-acoustics-hdl-64e-s2-hdl-64e-s21.html?download>
- , *Velodyne Lidar Puck VLP-16 - Datasheet*, Morgan Hill, CA, USA, 2018, 63-9229 Rev-H VLP-16.
- , *Velodyne Lidar Alpha Prime - Datasheet*, Apr. 2021, 63-9679 Rev-A VLS-128.
- Victron Energy B.V., *MultiPlus Wechselrichter/Ladegerät 800 VA – 5 kVA*, Almere, The Netherlands, 2023, Datenblatt. [Online]. Verfügbar: <https://www.victronenergy.de/upload/documents/Datasheet-MultiPlus-inverter-charger--800VA-5kVA-DE.pdf>
- Visteon, “Autonomous Driving – A Bird’s Eye View,” Jan. 2019. [Online]. Verfügbar: <https://www.visteon.com/wp-content/uploads/2019/01/autonomous-driving-a-birds-eye-view.pdf>
- A.-V. Vo, L. Truong-Hong, D. F. Laefer, und M. Bertolotto, “Octree-based region growing for point cloud segmentation,” *ISPRS Journal of Photogrammetry and Remote Sensing*, Bd. 104, S. 88–100, 2015.
- B.-N. Vo und W.-K. Ma, “The Gaussian Mixture Probability Hypothesis Density Filter,” *IEEE Transactions on Signal Processing*, Bd. 54, no. 11, S. 4091–4104, 2006.
- Volkswagen AG, “Lenkübersetzung des VW Touareg,” Feb. 2008, Mailverkehr mit dem Volkswagen Touareg Team am 29.02.2008.
- , “Lenkübersetzung des VW Tiguan,” Mai 2012, Mailverkehr mit Volkswagen am 09.05.2012.
- , *Selbststudienprogramm 275 - Der Phaeton: Luftfederung mit geregelter Dämpfung - Konstruktion und Funktion*, Wolfsburg, März 2002. [Online]. Verfügbar: <https://www.motor-talk.de/forum/aktion/Attachment.html?attachmentId=760105>
- , *Selbststudienprogramm 277 - Der Phaeton: Fahrwerk - Konstruktion und Funktion*, Wolfsburg, März 2002. [Online]. Verfügbar: <https://www.motor-talk.de/forum/aktion/Attachment.html?attachmentId=753521>

- , *Selbststudienprogramm 302 - Der Touareg: Fahrwerk und Allradkonzept - Konstruktion und Funktion*, Wolfsburg, Okt. 2002. [Online]. Verfügbar: <https://amaus.org/static/auto/vw/VW%20AUDI%20SSP/SSP%20302%20-%20Der%20Touareg%20-%20Fahrwerk%20und%20Allradkonzept.pdf>
- , *Selbststudienprogramm 374 - Schlupfregel- und Assistenzsysteme - Konstruktion und Funktion*, Wolfsburg, Sep. 2007.
- , *Selbststudienprogramm 389 - Der Parklenkassistent - Konstruktion und Funktion*, Wolfsburg, Jan. 2007. [Online]. Verfügbar: <https://www.motor-talk.de/forum/aktion/Attachment.html?attachmentId=677825>
- , *Selbststudienprogramm 399 - Die elektro-mechanische Lenkung mit Achs-Parallelem Antrieb (APA) - Konstruktion und Funktion*, Wolfsburg, Nov. 2007. [Online]. Verfügbar: http://www.vaglinks.com/vaglinks_com/docs/ssp/VWUSA.COM_SSP_399_Electromechanical_steering.pdf
- , *Selbststudienprogramm 404 - Der Tiguan 2008*, Wolfsburg, Okt. 2007. [Online]. Verfügbar: http://karo-nbg.selfhost.me/V.A.G-ssp/ssp404_d1.pdf
http://karo-nbg.selfhost.me/V.A.G-ssp/ssp404_d2.pdf
- Volvo. (2016, Sep.) Volvo first in the world with self-driving truck in underground mine. Press Release. Volvo Trucks. [Online]. Verfügbar: <https://www.volvogroup.com/en-en/news/2016/sep/volvo-first-in-the-world-with-self-driving-truck.html>
- E. von Hinüber, "Inertiale Messsysteme mit faseroptischen Kreiseln für Fahrdynamik und Topologiedaten-Erfassung," *ATZ - Automobiltechnische Zeitschrift*, Bd. 104, no. 6, S. 584–591, Juni 2002.
- F. von Hundelshausen, M. Himmelsbach, F. Hecker, A. Mueller, und H.-J. Wuensche, "Driving with Tentacles: Integral Structures of Sensing and Motion," *Journal of Field Robotics*, Bd. 25, no. 9, S. 640–673, 2008.
- H. Wallentowitz und K. Reif, Hrsg., *Handbuch Kraftfahrzeugelektronik*, Ser. ATZ/MTZ-Fachbuch. Friedr. Vieweg & Sohn Verlag, 2006. [Online]. Verfügbar: <http://books.google.de/books?id=Oe86b50DCDIC>
- J. Wendel, *Integrierte Navigationssysteme: Sensordatenfusion, GPS und Inertiale Navigation*, 2. Ausg. München: Oldenbourg Wissenschaftsverlag GmbH, 2011.
- M. Werling, L. Gröll, und G. Bretthauer, "Ein Multiregler zur Erprobung vollautonomen Fahrens – A Multi Controller for Testing Full-autonomous Driving," *at – Automatisierungstechnik, Schwerpunkt Themenheft: Kognitive Automobile*, Bd. 56, no. 11, S. 585–591, Nov. 2008.
- A. Westenberger, "Simultane Zustands- und Existenzschätzung mit chronologisch ungeordneten Sensordaten für die Fahrzeugumfelderfassung," Dissertation, Universität Ulm, Dez. 2014.
- K. Wildemann. (2019, Dez.) Die People-Mover-Pioniere. ZF Vision Magazin. [Online]. Verfügbar: https://www.zf.com/site/magazine/de/articles_25728.html

- H. Winner und M. Graupner, "PROMETHEUS – Welche Visionen wurden zur Realität?" in *Tagungsband 17. VDA Technischer Kongress*. Filderstadt, Germany: Verband der Automobilindustrie e.V. (VDA), März 2015, S. 25–48.
- H. Winner, S. Hakuli, F. Lotz, und C. Singer, Hrsg., *Handbuch Fahrerassistenzsysteme: Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort*, 3. Ausg., Ser. ATZ/MTZ-Fachbuch. Springer Vieweg, 2015.
- Wissenschaftlicher Beirat beim Bundesministerium Verkehr und digitale Infrastruktur, "Fahrermangel im deutschen Straßengüterverkehr – Strukturelle Treiber und verkehrspolitischer Handlungsbedarf," Feb. 2020. [Online]. Verfügbar: https://bmdv.bund.de/SharedDocs/DE/Anlage/G/fahrermangel-deutscher-strassengueterverkehr.pdf?__blob=publicationFile
- H. Wittich, G. Stegmaier, A. Of-Allinger, und U. Baumann. (2023, Mai) VOLVO EX90: Elektrischer Siebensitzer verzögert sich. *Auto Motor Sport*. [Online]. Verfügbar: <https://www.auto-motor-und-sport.de/neuheiten/neuer-volvo-ex90-als-xc90-nachfolger/>
- W. Woeber, M. Kefer, W. Kubinger, und D. Szuegyi, "Evaluation of Daylight and Thermal Infra-Red based Detection for Platooning Vehicles," in *Annals of DAAAM for 2012 and Proceedings of the 23rd International DAAAM Symposium*. DAAAM International, Vienna, Austria, 2012, S. 719–722. [Online]. Verfügbar: https://www.daaam.info/Downloads/Pdfs/proceedings/proceedings_2012/168.pdf
- P. Wolf, T. Ropertz, K. Berns, M. Thul, P. Wetzel, und A. Vogt, "Behavior-Based Control for Safe and Robust Navigation of an Unimog in Off-Road Environments," in *Proceedings of the 5th Commercial Vehicle Technology Symposium – CVT 2018*, K. Berns, K. Dressler, P. Fleischmann, D. Görges, R. Kalmar, B. Sauer, N. Stephan, R. Teutsch, und M. Thul, Hrsg. Wiesbaden, Germany: Springer Fachmedien, 2018, S. 63–76.
- M. Wood, P. Robbel, M. Maass, R. D. Tebbens, M. Meijs, M. Harb, J. Reach, K. Robinson, D. Wittmann, T. Srivastava, M. E. Bouzouraa, S. Liu, Y. Wang, C. Knobel, D. Boymanns, M. Löhning, B. Dehlink, D. Kaule, R. Krüger, J. Frtunikj, F. Raisch, M. Gruber, J. Steck, J. Mejia-Hernandez, S. Syguda, P. Blüher, K. Klonecki, P. Schnarz, T. Wiltschko, S. Pukallus, K. Sedlaczek, N. Garbacik, D. Smerza, D. Li, A. Timmons, M. Bellotti, M. O'Brien, M. Schöllhorn, U. Dannebaum, J. Weast, A. Tatourian, B. Dornieden, P. Schnetter, P. Themann, T. Weidner, und P. Schlicht, "Safety First for Autonomous Driving," White Paper, 2019, Aptiv Services US, LLC; AUDI AG; Bayrische Motoren Werke AG; Beijing Baidu Netcom Science Technology Co., Ltd; ContinentalTeves AG & Co oHG; Daimler AG; FCA US LLC; HERE Global B.V.; Infineon Technologies AG; Intel; Volkswagen AG. [Online]. Verfügbar: <https://group.mercedes-benz.com/documents/innovation/other/safety-first-for-automated-driving.pdf>
- WTD 81 - 320: Sensortechnologie und Robotik, "Leistungsbeschreibung – E E810 9C006 9F148 „Robotsensor Umwelterkennung – Erweiterung der EMS Vision-Software zur Steigerung der Autonomie“,“ Feb. 2009.

- H.-J. Wünsche, "Bewegungssteuerung durch Rechnersehen," Dissertation, Universität der Bundeswehr München, Fakultät für Luft- und Raumfahrttechnik, Institut für Systemdynamik und Flugmechanik, Neubiberg, 1987, also as Fachberichte Messen, Steuern, Regeln Bd. 20, Springer-Verlag, Berlin, 1988.
- Xsens Technologies B.V., *MTi MINIATURE ATTITUDE AND HEADING REFERENCE SYSTEM - Produktbroschüre*, Enschede, Niederlande, Aug. 2010. [Online]. Verfügbar: http://www.sungwhatech.com/product/pdf_01/MTi.pdf
- , *MTi and MTx User Manual and Technical Documentation*, Enschede, Niederlande, Okt. 2010, revision O.
- , *MTi and MTx User Manual and Technical Documentation*, Enschede, Niederlande, Jan. 2007, revision I.
- , *MTi-G User Manual and Technical Documentation*, Enschede, Niederlande, Okt. 2010, revision H.
- D. Yadron und D. Tynan, "Tesla driver dies in first fatal crash while using autopilot mode," *The Guardian*, Juni 2016. [Online]. Verfügbar: <https://www.theguardian.com/technology/2016/jun/30/tesla-autopilot-death-self-driving-car-elon-musk>
- Zeit Online, dpa, und pp. (2023, Feb.) Verband erwartet rund 87.000 fehlende Busfahrer bis 2030. Zeit Online. [Online]. Verfügbar: <https://www.zeit.de/mobilitaet/2023-02/busbranche-busfahrer-mangel-ausbildung-verkehrswende>
- ZF Lenksysteme GmbH, "Übersetzungverlauf der Lenkung im VW Touareg," Feb. 2008, Mailverkehr mit ZF am 27.02.2008.
- . (2007) ZF-Servotronic® 2 für Pkw und leichte Nutzfahrzeuge. [Online]. Verfügbar: <https://docplayer.org/49022831-Zf-servotronic-2-fuer-pkw-und-leichte-nutzfahrzeuge.html>
- M. Zhang, K. Liu, und C. Li, "Unmanned Ground Vehicle Positioning System by GPS/Dead-Reckoning/IMU Sensor Fusion," in *Proceedings of the Annual International Conference on Electronics, Electrical Engineering and Information Science (EEEIS)*, Ser. Advances in Engineering Research. Atlantis Press, Dez. 2016, S. 737–747.
- J. Ziegler, P. Bender, M. Schreiber, H. Lategahn, T. Strauss, C. Stiller, T. Dang, U. Franke, N. Appenrodt, C. G. Keller, E. Kaus, R. G. Herrtwich, C. Rabe, D. Pfeiffer, F. Lindner, F. Stein, F. Erbs, M. Enzweiler, C. Knoepfel, J. Hipp, M. Haeis, M. Trepte, C. Brenkand, A. Tamke, M. Ghanaat, M. Braun, A. Joos, H. Fritz, H. Mock, M. Hein, und E. Zeeb, "Making Bertha Drive – An Autonomous Journey on a Historic Route," *IEEE Intelligent Transportation Systems Magazine*, Bd. 6, no. 2, S. 8–20, 2014.
- J.-M. Zogg, *GPS und GNSS: Grundlagen der Ortung und Navigation mit Satelliten*. Thalwil, Schweiz: u-blox AG, Okt. 2011. [Online]. Verfügbar: http://www.zogg-jm.ch/Dateien/Update_Zogg_Deutsche_Version_Jan_09_Version_Z4x.pdf