



**KAMERABASIERTER ASSISTENZ FÜR  
NUTZFAHRZEUGE**

von

**Christian Fuchs**

Genehmigte Dissertation zur Verleihung  
des akademischen Grades eines

**Doktors der Naturwissenschaften (Dr. rer. nat.)**

Fachbereich 4: Informatik  
UNIVERSITÄT KOBLENZ-LANDAU

Vorsitzender des Promotionsausschusses	<b>Prof. Dr. Jan Jürjens</b>
Vorsitzende der Promotionskommission	<b>Prof. Dr. Maria A. Wimmer</b>
Erster Berichterstatter und Betreuer	<b>Prof. Dr.-Ing. Dietrich Paulus</b>
Zweiter Berichterstatter	<b>Prof. Dr. Dieter Zöbel</b>
Datum der wissenschaftlichen Aussprache	<b>15. Mai 2020</b>



## ZUSAMMENFASSUNG

---

Lastkraftwagen werden weltweit in verschiedenen Konfigurationen für den Transport von Gütern und für Spezialanwendungen genutzt. Das sichere Navigieren solcher Großfahrzeuge stellt eine herausfordernde Aufgabe für den Fahrer und unter Umständen für automatische Steuerungssysteme, z. B. im semi-autonomen oder autonomen Betrieb, dar. Insbesondere in kinematisch schwierigen Situationen, auf unebenem Untergrund oder bei engen Platzverhältnissen steigt die Komplexität des sicheren Steuerns.

Umso verwunderlicher ist es, dass unterstützende Systeme, welche die speziellen Belange von LKW und insbesondere LKW-Gespansen adressieren, im Vergleich zum Angebot auf dem PKW-Markt nur in rudimentärem Maß angeboten werden. Das Gefahrenpotential, das von Großfahrzeugen ausgeht, wird zunehmend in der aktuellen Tagespolitik diskutiert. Unter anderem wird die verpflichtende Einführung von Fahrerassistenzsystemen gefordert.

In dieser Arbeit werden zwei Themen behandelt, die sich mit den speziellen Anforderungen von Nutzfahrzeugen befassen. Im ersten Themenblock wird die Erfassung der Artikulation in einem LKW-Anhänger-Gespann mit Drehschemelanhänger bearbeitet. Dabei wird ein neuartiges Verfahren vorgestellt, das auf an der Universität Koblenz-Landau geleisteten Vorarbeiten aufbaut. Es ermöglicht die Erfassung von Knickwinkeln in mehreren Dimensionen und ist daher auch für den Einsatz auf unebenem Grund, wie beispielsweise in Baustellen- oder Agrarszenarien geeignet. Das Verfahren nutzt dabei eine Kamera und ein künstliches Muster, um den Gespannzustand zu ermitteln.

Im zweiten Bereich wird die Umfeldvisualisierung durch eine Vogelperspektivenansicht thematisiert. Herkömmliche Systeme weisen systembedingte Verzerrungen auf, die unnatürlich wirkende Ansichten zur Folge haben. Diesem Problem wird durch die Entwicklung einer neuen Herangehensweise begegnet: Mit den Tiefendaten von Stereokameras wird eine Umgebungsmodellierung realisiert, die im nächsten Schritt perspektivisch korrekt mit Kamerabildern fusioniert wird. Mittels einer speziell für diesen Fall erweiterten Stereo-Hindererkennung wird zudem eine zusätzliche Umfeldüberwachung ermöglicht.

Die beiden in dieser Dissertation vorgestellten Verfahren beinhalten grundlegende Arbeiten, die für die Entwicklung maßgeschneiderter Assistenzsysteme für LKW angewandt werden können.



## ABSTRACT

---

Commercial vehicles are used worldwide in a variety of configurations for the transportation of goods and special applications. The safe navigation of such large vehicles is a challenging task for the driver and possibly for automatic control systems, e.g. in semi-autonomous or autonomous operation. Especially in kinematically complex scenarios, on uneven ground or in narrow spaces, the complexity of safe control rises.

Surprisingly supporting systems which specifically address the issues of trucks and in particular truck-trailer-combinations are offered rarely compared to the passenger car market. The risk potential caused by large vehicles is increasingly discussed in daily politics. A mandatory implementation of driver assistance systems is claimed.

Two main topics dealing with the special requirements of commercial vehicles are investigated in this thesis. The first topic deals with the detection of the articulation state within in a truck-trailer-combination vehicle with a fifth-wheel trailer. A novel approach is presented which is based upon prior work accomplished at the University of Koblenz-Landau. It enables the detection of articulation angles in multiple dimensions and is applicable in off-road terrain like on construction sites or in agricultural applications. The method uses a camera and an artificial pattern for the articulation detection.

The second topic addresses the visualization of the vehicle's surroundings using a bird's eye view. Conventional systems suffer from systematic heavy distortions which cause unnatural looking views. The problem is overcome by developing a new approach: Using depth information extracted from stereo images, an environment model is created which is fused with camera images in the next step. Utilizing a stereo obstacle detection specifically extended for this purpose, an additional environment monitoring is enabled.

Both methods presented in this PhD thesis contain fundamental work which is applicable for the development of tailored assistance systems for commercial vehicles.



## EIGENE PUBLIKATIONEN

---

- [1] Simon Eggert, Christian Fuchs, Frank Bohdanowicz und Dieter Zöbel. Reaktive optische Einknickwinkelvermessung bei Gliederfahrzeugen. In Wolfgang A. Halang (Editor), *Echtzeit 2013: Funktionale Sicherheit*, Seiten 19–28. Springer, Berlin Heidelberg, 2013. DOI: 10.1007/978-3-642-41309-4\_3.
- [2] Christian Fuchs, Simon Eggert, Benjamin Knopp und Dieter Zöbel. Pose detection in truck and trailer combinations for advanced driver assistance systems. In *IEEE Intelligent Vehicles Symposium (IV)*, Seiten 1175–1180. IEEE, 2014. DOI: 10.1109/IVS.2014.6856547.
- [3] Christian Fuchs, Simon Eggert und Dieter Zöbel. Optical Measurement of the Angle between Truck and Trailer. In Karsten Berns, Syed Atif Mehdi und Abubakr Muhammad (Editoren), *Field and Assistive Robotics: Advances in Systems and Algorithms*, Seiten 44–53. Shaker Verlag, Herzogenrath, 2014. ISBN: 978-3-8440-2753-2.
- [4] Christian Fuchs, Benjamin Knopp, Dieter Zöbel und Dietrich Paulus. Model-based Evaluation of Practical Sensor Noise Impacts in Articulated Vehicle Driving Scenarios. In *IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2017. DOI: 10.1109/IVS.2017.7995780.
- [5] Christian Fuchs, Frank Neuhaus und Dietrich Paulus. Advanced 3-D Pose Estimation for Articulated Vehicles. In Dietrich Paulus, Christian Fuchs und Detlev Droege (Editoren), *9th Open German-Russian Workshop on Pattern Recognition and Image Understanding (OGRW 2014), Electronic on-site Proceedings*, Seiten 20–22, Koblenz, 2015. Universität Koblenz-Landau. URN: urn:nbn:de:hbz:kob7-2015051206.
- [6] Christian Fuchs, Frank Neuhaus und Dietrich Paulus. Advanced 3-D trailer pose estimation for articulated vehicles. In *IEEE Intelligent Vehicles Symposium (IV)*, Seiten 211–216. IEEE, 2015. DOI: 10.1109/IVS.2015.7225688.
- [7] Christian Fuchs, Frank Neuhaus und Dietrich Paulus. 3D Pose Estimation for Articulated Vehicles Using Kalman-Filter Based Tracking. *Pattern Recognition and Image Analysis*, 26(1):109–113, 2016. DOI: 10.1134/S1054661816010077. ISSN: 1555-6212.
- [8] Christian Fuchs und Dietrich Paulus. Perspectively Correct Bird’s Views Using Stereo Vision. In *Autonomous Vehicles and Machines Conference, IS&T Electronic Imaging 2017*, Seiten 22–28(7),

Springfield, USA, 2017. Society for Imaging Science and Technology (IS&T). DOI: 10.2352/ISSN.2470-1173.2017.19.AVM-014.

- [9] Christian Fuchs und Dietrich Paulus. Perspectively correct construction of virtual views. In *6th International Conference on Pattern Recognition Applications and Methods - Volume 1: ICPRAM*, Seiten 626–632. Scitepress, 2017. ISBN: 978-989-758-222-6.
- [10] Christian Fuchs und Dietrich Paulus. Dense Surround View Computation with Perspective Correctness. In *Autonomous Vehicles and Machines Conference, IS&T Electronic Imaging 2018*, Seiten 282–1–282–8, Springfield, USA, 2018. Society for Imaging Science and Technology (IS&T). DOI: 10.2352/ISSN.2470-1173.2018.17.AVM-282.
- [11] Christian Fuchs und Dietrich Paulus. Integration of Advanced Stereo Obstacle Detection with Perspectively Correct Surround Views. In *Autonomous Vehicles and Machines Conference, IS&T Electronic Imaging 2019*, Seiten 32–1–32–10, Springfield, USA, 2019. Society for Imaging Science and Technology (IS&T). DOI: 10.2352/ISSN.2470-1173.2019.15.AVM-032.
- [12] Christian Fuchs, Dieter Zöbel und Dietrich Paulus. 3-D Pose Detection for Articulated Vehicles. In Emanuele Menegatti, Nathan Michael, Karsten Berns und Hiroaki Yamaguchi (Editoren), *Intelligent Autonomous Systems 13*, Band 302, Seiten 459–472. Springer, Cham, 2016. DOI: 10.1007/978-3-319-08338-4\_34. ISBN: 978-3-319-08338-4.
- [13] Frank Neuhaus, Christian Fuchs und Dietrich Paulus. High-resolution hyperspectral ground mapping for robotic vision. In *International Conference on Machine Vision (ICMV 2017; SPIE 10696)*, Seiten 1–9, Bellingham, WA USA, 2018. DOI: 10.1117/12.2310066.
- [14] Christian Winkens, Christian Fuchs, Frank Neuhaus und Dietrich Paulus. Optical Truck Tracking for Autonomous Platooning. In George Azzopardi und Nicolai Petkov (Editoren), *Computer Analysis of Images and Patterns (CAIP)*. LNCS Vol. 9257, Seiten 38–48. Springer, Berlin, Heidelberg, 2015. DOI: 10.1007/978-3-319-23117-4\_4. ISBN: 978-3-319-23116-7.



# INHALTSVERZEICHNIS

---

1	MOTIVATION UND EINFÜHRUNG	1
1.1	Eigener Beitrag . . . . .	5
2	DEFINITIONEN UND NOTATIONEN	9
2.1	Koordinatensysteme und Vektoren . . . . .	9
2.2	Position und Orientierung . . . . .	11
2.3	Definition mathematischer Funktionen . . . . .	12
2.4	Fehlermaße . . . . .	13
2.4.1	Statistische Fehlermaße . . . . .	13
2.4.2	Earth Mover's Distance . . . . .	14
3	GRUNDLAGEN UND STAND DER TECHNIK	17
3.1	Kameras als optische Sensoren . . . . .	17
3.2	Kamerakalibrierung . . . . .	18
3.2.1	Verzerrungskorrektur . . . . .	20
3.3	Perspektivische 2-D-Verzerrung . . . . .	22
3.3.1	Berechnung einer Homographie . . . . .	22
3.4	Künstliche Marker . . . . .	24
4	ZUSTAND ARTIKULIERTER FAHRZEUGE	27
4.1	Einleitung . . . . .	27
4.2	Stand der Technik . . . . .	27
4.2.1	2-D-Verfahren zur Statusschätzung . . . . .	29
4.2.2	Kinematikmodelle artikulierter Fahrzeuge . . . . .	33
4.2.3	Trajektorienbeurteilung . . . . .	34
4.3	Modellbildung . . . . .	34
4.3.1	Zugfahrzeug . . . . .	35
4.3.2	Anhänger . . . . .	36
4.3.3	Fahrzeuggespann . . . . .	37
4.4	2-D-Ansatz . . . . .	38
4.4.1	Virtuelle Testumgebung . . . . .	40
4.4.2	Evaluation . . . . .	42
4.5	3-D-Ansatz . . . . .	49
4.5.1	Musterpose und Winkeldekomposition . . . . .	50
4.5.2	Evaluation . . . . .	55
4.6	Temporale Kohärenz . . . . .	60
4.6.1	Modellierung des Systemzustands . . . . .	62
4.6.2	Evaluation . . . . .	64
4.7	Auswirkungen auf die Trajektorienqualität . . . . .	68
4.7.1	Gespannzustand und Zustandsschätzung . . . . .	68
4.7.2	Fahrermodell-basierte Simulation . . . . .	69
4.7.3	Evaluation . . . . .	72

5	PERSPEKTIVISCH KORREKTE VOGELANSICHT	79
5.1	Einleitung . . . . .	79
5.2	Stand der Technik . . . . .	79
5.2.1	Stereokameras und Stereoverarbeitung . . . . .	79
5.2.2	Perspektivische 2-D Transformationen mit Homographien . . . . .	89
5.2.3	Berechnung der Vogelperspektive . . . . .	89
5.2.4	Bild-basiertes Rendering . . . . .	90
5.2.5	Stereo-Hindernisdetektion . . . . .	91
5.2.6	Umgebungsrepräsentation mit Rasterkarten . . . . .	92
5.3	Modellbildung . . . . .	92
5.4	Homographie-Verschattung . . . . .	95
5.5	Transformation mit Tiefendaten . . . . .	97
5.5.1	Geometrie virtueller Kameras . . . . .	98
5.5.2	Punktwolken und virtuelle Kameraansichten . . . . .	101
5.5.3	Ergebnisse mit generischen Datensätzen . . . . .	104
5.5.4	Ergebnisse mit Fahrzeug-Stereo-Datensätzen . . . . .	112
5.6	Integration und geschlossenes Oberflächenmodell . . . . .	116
5.6.1	Gleitende lokale Rasterkarte . . . . .	116
5.6.2	Geschlossenes Oberflächenmodell . . . . .	122
5.6.3	Partielle Homographien . . . . .	124
5.6.4	Ergebnisse . . . . .	128
5.7	Verbesserte Bodenapproximation . . . . .	135
5.8	Erweiterte Stereo-Hinderniserkennung . . . . .	140
5.8.1	Verallgemeinerter Ansatz . . . . .	144
5.8.2	Testergebnisse . . . . .	152
5.9	Anwendungsszenario und Testdatensatz . . . . .	158
6	FAZIT	171
	ABBILDUNGSVERZEICHNIS	175
	TABELLENVERZEICHNIS	179
	ALGORITHMENVERZEICHNIS	179
	VERZEICHNIS MATHEMATISCHER SYMBOLE	181
	LITERATURVERZEICHNIS	187
	INTERNETQUELLEN	201

## MOTIVATION UND EINFÜHRUNG

---

Lastkraftwagen (LKW) werden weltweit für eine Vielzahl von logistischen Aufgaben eingesetzt. In Deutschland wurden im Jahr 2018 ca. 3 200 Millionen Tonnen Güter durch inländische LKW transportiert [12]. Dabei lässt sich über die vergangenen zehn Jahre ein kontinuierlich steigendes Transportaufkommen feststellen und auch für die kommenden Jahre prognostizieren. Dies wird auch durch die in den vergangenen fünf Jahren kontinuierlich steigenden Zulassungszahlen belegt [4]. Der Gesamtanteil an Transportleistung von LKW im Güterverkehr liegt nach Angaben des Bundesamts für Güterverkehr bei 71.8 % im Jahr 2017 bei kontinuierlich steigender Prognose bis mindestens in das Jahr 2021 [5].

Das sichere Navigieren großer und schwerer Fahrzeuge stellt hohe Anforderungen an die Fahrer, Fahrerassistenzsysteme und an die ggf. automatisierten Fahrzeuge. Insbesondere in kinematisch schwierigen Situationen, auf unebenem Untergrund wie in Baustellen oder in Agrar- und Forstszenarien, sowie bei enger Bebauung steigt die Komplexität des sicheren Steuerns.

Im Fokus der aktuell verfügbaren und innovativen Unterstützungssysteme stehen primär Personenkraftwagen (PKW): Für moderne PKW werden eine Vielzahl von Fahrerassistenzsystemen angeboten, die es zum Ziel haben, den Fahrer zu entlasten, zur Steigerung der Verkehrssicherheit beizutragen oder das Steuern des Fahrzeugs komfortabler zu gestalten. Die Fahrerassistenzsysteme nutzen dabei Sensoren, um Messdaten aus ihrer Umwelt aufzunehmen und zu interpretieren. Je nach Verarbeitungsergebnis wird eine Aktion ausgelöst, die dem Fahrer ein Signal gibt oder ein direktes Eingreifen in das Fahrzeug bewirkt.

Auf dem Weg zu immer komplexeren Fahrerassistenzsystemen und damit zu höheren Autonomiegraden für Fahrzeuge werden immer mehr neue Sensoren integriert und genutzt. Seit einigen Jahren zählen zu diesen Sensoren auch Kameras, die zur Beobachtung inner- und außerhalb des Fahrzeugs eingesetzt werden.

Eine beispielhafte, sehr einfache Anwendung stellen Rückfahrkameras dar. Dabei werden Bilder einer an der Rückseite des Fahrzeugs angebrachten Kamera genutzt, um sie dem Fahrer nach einfacher Aufbereitung direkt anzuzeigen und den Blick hinter das Fahrzeug zu ermöglichen und damit den Totraum, der bei der Verwendung von Fahrzeugspiegeln in Kombination mit der gewöhnlichen Fahrzeuggeometrie vorhanden ist, zu minimieren. Bereits eine so einfache, passive Assistenz dient der Steigerung der operativen Sicher-

heit des Fahrzeugs. Komplexere Assistenzsysteme unterstützen inzwischen teilweise eigenständig ganze Fahraufgaben in einfacheren Fahr Szenarien, wie z. B. auf Autobahnen (u. a. Spurhalte-, Distanz- und Totraum-Überwachungsassistenten).

Für LKW sind vergleichsweise nur wenige und grundlegende Fahrerassistenzsysteme verfügbar, die den Fahrer beim Betrieb unterstützen, obwohl aktuelle Technologien ein großes Potential für erweiterte Fahrerassistenzsysteme bieten. Eine Weiterentwicklung in diesem Bereich stellt folglich eine wichtige Aufgabe dar. Die verfügbaren Assistenzsysteme beschränken sich zudem meist auf das Zugfahrzeug, obwohl insbesondere in Transportszenarien andere Fahrzeugkonfigurationen eingesetzt werden: Oftmals werden LKW in Kombination mit unterschiedlichen Anhängern verwendet, um die Effizienz des Transportes zu erhöhen. Angekuppelte Anhänger werden vom Zugfahrzeug bewegt und können relativ zu diesem ihre Lage während der Fahrt verändern. Da dadurch in Fahrzeuggespannen gewisse Freiheitsgrade vorliegen, werden diese als *artikulierte* Fahrzeuge bezeichnet.

Die Kinematik artikulierter Fahrzeuge ist naturgemäß komplexer als die nicht-artikulierter Fahrzeuge. Das sichere Navigieren artikulierter Fahrzeuge stellt folglich eine noch komplexere Aufgabe dar, da die Bewegung der Anhänger stets in den Steuerungsprozess und dessen Planung miteinbezogen werden muss.

Die Aktualität der Fragestellung bzgl. Assistenzsystemen für Nutzfahrzeuge bzw. Großfahrzeuge – sowohl politisch als auch gesellschaftlich – wird u. a. durch die folgenden Thematisierungen bzw. Berichterstattungen verdeutlicht:

- Zur Landtagswahl in Rheinland-Pfalz (2016) stellte die Bundeszentrale für politische Bildung ein Online-System zur Verfügung („Wahl-O-Mat“), anhand dessen die Wähler einen Vergleich ihrer Ansichten mit parteilichen Positionen anhand von 38 ausgewählten Thesen anstellen können. Eine dieser Thesen beschäftigt sich mit der Zulassung von Großgespannen auf öffentlichen Straßen: *Lang-LKW („Gigaliner“) sollen in Rheinland-Pfalz versuchsweise fahren dürfen.* [@6]
- Die Abgeordneten des Deutschen Bundestags haben die Bundesregierung im Juni 2018 in einem fraktionsübergreifenden Antrag dazu aufgefordert, Abbiegeassistenten in Nutzfahrzeugen gesetzlich vorzuschreiben. [@8]
- Im Juli 2018 hat Bundesverkehrsminister Andreas Scheuer eine Initiative zur Ausstattung von LKW mit elektronischen Abbiegeassistenten gestartet. Vorreiter sollen demnach Fahrzeuge der Bundesbehörde werden, die im Jahr 2019 vollständig ausgerüstet sein sollen. [@10]

- Im Januar 2019 haben Experten des Deutschen Verkehrsgerichtstages an die Bundesregierung appelliert, auf den verpflichtenden Einbau von Assistenzsystemen in LKW hinzuwirken, um die Zahl der Unfalltoten zu verringern [3].
- Die Tagesschau berichtet im Januar 2019 über ein Rechtsgutachten, das eine Verpflichtung zu Abbiege-Assistenzsystemen für LKW grundsätzlich möglich sei. Im Artikel wird Bundesverkehrsminister Andreas Scheuer zitiert, der darüber berichtet, dass die EU-Kommission vorschlägt, dass solche Systeme erst ab 2024 zur Pflicht werden sollen [11].
- Im August 2019 diskutiert ein Artikel des „Spiegel Online“ die Kontroverse zwischen dem Nutzen von Assistenzsystemen für Lastkraftwagen und den damit verbundenen Kosten. Dabei wird insbesondere der Einsatz von Notbremsassistenten in LKW diskutiert [7].

Die Entwicklung von Fahrerassistenzsystemen und Systemen zur Umfelderkennung für Großfahrzeuge, die auf ihre speziellen Belange und ihre speziellen Konfigurationen zugeschnitten sind, stellt eine herausfordernde, interessante und gesellschaftsrelevante Aufgabe dar. Die vorliegende Arbeit beschäftigt sich dabei mit zwei Feldern, die in diesem Bereich angesiedelt sind.

In der Forschungsgruppe Echtzeitsysteme der Universität Koblenz-Landau werden u. a. Verfahren und Fahrerassistenzsysteme entwickelt, die artikulierte Fahrzeuge adressieren. Dabei ist ein unterstütztes Rückwärtsfahren von Gespannen mit einer kamerabasierten Assistenz und entsprechenden Steuerunterstützungen ein zentrales Thema. Unter anderem wurde ein patentiertes System für die Bestimmung der beiden Artikulationswinkel für ein Gespann mit Drehschemelhänger („Zweiachsanhänger“) entwickelt [BZW06]. Das System nutzt eine Kamera, um ein Pyramiden-förmiges Muster zu beobachten und extrahiert die Gierwinkel aus den Bildern.

An dieser Stelle setzen die hier vorgestellten Arbeiten im Bereich der Gespannstatusschätzung an. Ziel ist es, das Verfahren auf eine 3-D-Umgebung, d. h. auf unebenen Fahruntergrund, zu erweitern und dazu notwendige Modelle zu entwickeln. Die Gespannstatusschätzung stellt eine wichtige und kritische Grundlage für kinematische Berechnungen und damit verbundene Fahrerassistenzsysteme dar.

In einem Forschungsprojekt der Arbeitsgruppe Aktives Sehen der Universität Koblenz-Landau in Kooperation mit einem Industrieunternehmen – in dessen Rahmen Teile der in der Dissertation vorgestellten Arbeiten angefertigt wurden – wird der Einsatz und die Entwicklung von perspektivisch korrekten Ansichten aus der Vogelperspektive auf einem Großfahrzeug thematisiert.

Konkret betrifft der Aufbau ein spezialisiertes Baustellenfahrzeug, das durch einen sich hinter dem Fahrzeug befindlichen Bediener ferngesteuert wird. Der Bediener hat demzufolge keine direkte Sicht auf den Frontbereich des Fahrzeugs. Dabei wird keine weitere Person eingesetzt, die beispielsweise den Fahrerplatz des Fahrzeugs einnimmt. Hauptaufgabe des Bedieners ist die Steuerung einer sich am Heck des Fahrzeugs befindlichen Saugeinrichtung („Saugbagger“), die dem Abtragen von Erdmaterial dient. Im ferngesteuerten Modus bewegt sich das Fahrzeug in langsamer Geschwindigkeit (maximal 6 km/h). Fernziel ist es, dem Bediener eine visuelle Überwachung des vorderen Fahrzeugbereichs zu ermöglichen, um eine operative Sicherheit zu gewährleisten.

Die dazu notwendige Visualisierung soll möglichst einfach verständlich sein, um den Bediener nicht abzulenken. Eine Ansicht aus der Vogelperspektive auf das Fahrzeug bietet dabei eine intuitive Modalität, um eine leicht verständliche Umgebungsvisualisierung zu ermöglichen. Konventionelle Systeme für die Berechnung von Ansichten aus der Vogelperspektive bieten jedoch nur eine eingeschränkte Sichtweite um das Fahrzeug und zeigen stark verzerrte Ansichten, sobald unebenes Terrain – wie in Baustellenszenarien zu erwarten – oder Hindernisse angezeigt werden sollen.

Neben der bloßen Visualisierung ist die Identifikation von potentiellen Gefahrenpunkten (Hindernissen) ein Aspekt, der einen Mehrwert für das System darstellt und als Grundlage für eine mögliche Weiterentwicklung im Hinblick auf Warnsysteme bzw. eingreifende Systeme (z. B. automatisierter Not-Stopp) dienen kann.

Ein erweitertes System zur Adressierung dieser Probleme ist eine interessante und sinnvolle Aufgabenstellung. Die dazu entwickelten Ideen und Algorithmen werden in der vorliegenden Arbeit im zweiten Themenbereich vorgestellt.

Die im Folgenden vorgestellten, neu entwickelten Verfahren verzichten bewusst auf Maschinelles Lernen bzw. Neuronale Netze, die in den letzten Jahren an Bedeutung im Bereich der Sensordatenverarbeitung gewinnen. Dies ist insbesondere dadurch zu begründen, dass das hinter den Arbeiten stehende Projektgefüge den Einsatz dieser Techniken nicht vorsieht. Die Einsetzbarkeit dieser Techniken in einem Produkt bzw. auf einem täglich operierenden Fahrzeug ist zum Zeitpunkt der Projektdurchführung rechtlich noch nicht hinreichend geklärt bzw. bewiesen. Im Rahmen der Betrachtung des jeweiligen Stands der Technik spielen Verfahren auf Basis von Maschinellern Lernen nichtsdestotrotz eine wichtige Rolle. An den entsprechenden Stellen werden im Folgenden entsprechende Ansätze aufgeführt.

In dieser Dissertation werden zwei Aspekte hinsichtlich Lastkraftwagen untersucht mit der Intention, Grundlagen für auf große Fahrzeuge zugeschnittene Assistenzsysteme zu entwickeln – unabhängig davon, ob das Fahrzeug von einem menschlichen Fahrer, semiauto-

nom oder autonom gesteuert wird.

Um die operative Sicherheit von *artikulierten* Fahrzeugen erhöhen zu können, ist die Kenntnis des aktuellen Artikulationszustands eine zwingende Voraussetzung. Kinematische Extrapolation sowie Steuerungshilfen sind nur dann sinnvoll realisierbar, wenn der Spannungszustand kontinuierlich ermittelt werden kann, um eventuelle Abweichungen in geplanten Trajektorien zu detektieren.

Das sichere Navigieren eines Großfahrzeugs setzt eine Umgebungserfassung und eine geeignete Visualisierung für Fahrer bzw. bei semi-autonom oder autonom betriebenen Fahrzeugen überwachende Personen voraus. Da die Fahrzeuge über größere Dimensionen als PKW verfügen ist es erforderlich, entsprechend große Bereiche damit abzudecken. Insbesondere bei engen Straßenverhältnissen, in Abbiegesituationen oder bei unebenem Terrain, wie z. B. auf Baustellen oder im Agrarbereich, sind unterstützende Systeme besonders wichtig. Ein konkretes Beispiel ist die Visualisierung aus der Vogelperspektive, die die direkte Umgebung sowie die dazu relative Position des Fahrzeugs anzeigt.

## 1.1 EIGENER BEITRAG

Die in der vorliegenden Dissertation vorgestellten Publikationen sind bereits auf internationalen Konferenzen vorgestellt und veröffentlicht. Die Literaturliste der eigenen Veröffentlichungen befindet sich am Anfang der Ausarbeitung. Die dort verwendeten Referenzschlüssel sind numerisch (Formatierungsbeispiel: [1]), die Referenzschlüssel zu fremden referenzierten Arbeiten alphanumerisch (Formatierungsbeispiel: [ABC19]). Literaturschlüssel für Verweise auf Internetquellen beginnen mit „@“-Zeichen gefolgt von einer Nummerierung (Formatierungsbeispiel: [@1]). Anhand dieser Unterscheidung kann eine im folgenden Text angegebene Literaturreferenz den eigenen Publikationen oder den referenzierten Arbeiten im *allgemeinen* Literaturverzeichnis bzw. den Internetquellen zugeordnet werden.

In der vorliegenden Arbeit werden eigene Beiträge in zwei Großbereichen vorgestellt. In beiden behandelten Teilbereichen werden entwickelte Verfahren erläutert, die sich mit der Bewegung auf nicht-planarem Untergrund befassen. Die Annahme des ebenen Bodens ist jedoch in einer Vielzahl von Veröffentlichungen implizite oder explizite Grundannahme. Diese Annahme ist jedoch in den Szenarien bzw. Fahrmanövern, in denen der Beitrag der Verfahren zur operativen Betriebssicherheit einen besonderen Stellenwert einnimmt, verletzt. Dies begründet, dass dieser Aspekt einer besonderen Aufmerksamkeit würdig ist.

Im ersten Großbereich, der sich mit der Statuserfassung eines Fahrzeugespanns beschäftigt, wird ein an der Universität Koblenz-Landau entwickeltes Verfahren, das den Knickwinkelzustand eines Ge-

spanns mit Drehschemelanhänger („Zweiachsanhänger“) bestimmen kann, Software-technisch aufbereitet [1] und reimplementiert [3] und anschließend mit einer Simulationsumgebung evaluiert [2].

Das Verfahren stützt sich dabei auf eine 2-*D*-Ebenenannahme, die lediglich Artikulation in Gier-Richtung betrachtet. Da diese Annahme nicht in allen Situationen zutrifft (z. B. unebenes Terrain), wird das Verfahren auf eine 3-*D*-Statusschätzung erweitert, für die zunächst eine Modellierung des vorliegenden Problems vorgenommen wird, um die Anzahl der enthaltenen Freiheitsgrade zu verringern [12]. Die initiale Umsetzung, die auf der Auswertung von einzelnen Aufnahmen beruht, wird erweitert, um die Stabilität der Gespannstatuserkennung zu steigern und die Präzision in der Erkennung zu erhöhen [5, 6]. Abschließend werden die drei Verfahren mit einem simulierten Szenario gegeneinander getestet, um die Auswirkung der Erkennungspräzision auf ein Fahrmanöver abschätzen zu können [4].

Im Bereich der Gespannzustandsschätzung gibt es nur wenige Vorarbeiten und Patente. Diese beschränken sich jedoch auf Einachsanhänger (d. h. einen Artikulationspunkt) (siehe Kapitel 4.2.1). Der eigene Beitrag in diesem Bereich ist in der Erweiterung der an der Universität Koblenz-Landau entwickelten Gespannzustandsschätzung in eine 3-*D*-Umgebung angesiedelt. Damit einhergehend ist das Loslösen von der Annahme des ebenen Untergrunds. Die Entwicklung einer flexiblen Kamera-Simulationsumgebung für Fahrzeugbewegungen stellt die Grundlage für die vorgenommenen Evaluationen dar. Die Adaptierbarkeit der Simulationsumgebung wurde mit dem Einsatz in der Evaluation einer Folgefunktion für LKW dokumentiert [14].

Der zweite Großbereich befasst sich mit der Umgebungsvisualisierung in einer virtuellen Vogelperspektivenansicht. In diesem Bereich bilden einfache, schnell zu berechnende Verfahren den Stand der Technik, die jedoch nur eine geringe Reichweitenabdeckung um das Fahrzeug sowie eine Ebenenannahme für den Untergrund beinhalten.

Verbreitete Systeme für die Bereitstellung von Ansichten aus der Vogelperspektive nutzen am Fahrzeug verteilte Kameras, um eine virtuelle Ansicht von oben zu generieren. Dabei kommen Homographien zum Einsatz, um die Kameraansichten zu berechnen. Je Kamera wird eine Homographie genutzt, um die Ansicht zu transformieren. Bei diesen Techniken wird eine Ebenenannahme über die Position der Objekte in der Szene getroffen, die beim Auftreten von Objekten, die sich vom Boden erheben, zu starken und unnatürlichen Effekten führt. Diese Objekte sind jedoch gewöhnlich die für die Navigationsplanung relevantesten Objekte in der Szene. Dieses Problem wurde untersucht und mit dem Homographie-Verschattungseffekt beschrieben [8, 9].

Darauf aufbauend wird eine Technik vorgestellt, die Tiefenfor-



mationen für die Berechnung einer perspektivisch korrekten Ansicht nutzt. Dabei werden die aus Stereo-Aufnahmen extrahierten Tiefeninformationen genutzt, um einzelne Punkte korrekt für die Berechnung der virtuellen Ansicht zu transformieren. Das Verfahren zeigt auf hochauflösenden Datensätzen gute Ergebnisse [9]. Auf Fahrzeugdatensätzen funktioniert das Verfahren analog. Es kann jedoch keine vergleichbare Bereichsabdeckung und Bildqualität erreicht werden, da die dort üblich eingesetzten Kameras über eine geringere Auflösung verfügen und die Stereo-Matching-Verfahren weniger dichte Tiefendaten generieren [8].

Um dichte und perspektivisch korrekte Ansichten aus der Vogelperspektive zu ermöglichen, wird ein Rasterkarten-basiertes Höhenmodell entwickelt, das 3-D-Punktewolken, die u. a. aus Stereo-Daten gewonnen werden, als Eingabe nutzt. Das Höhenmodell wird genutzt, um eine Umgebungsrepräsentation des Fahrzeugumfelds zu realisieren. Auf Basis dieses Modells wird ein geschlossenes Oberflächenmodell erzeugt, dessen einzelne geometrische Elemente lokal planar sind. Das Attribut beschreibt dabei die Eigenschaft des Modells, eine möglichst lückenlose Beschreibung des Bodenbereichs abzudecken. Zur Texturierung dieses Oberflächenmodells werden partielle Homographien auf Kamerabilder angewandt, die die Bilder dann lokal korrekt transformieren.

Im Ergebnis wird eine qualitativ hochwertige, perspektivisch korrekte Vogelperspektivenansicht erzeugt, die als Zwischenschritt zudem eine 3-D-Ansicht der Umgebung ermöglicht [10].

Das so entwickelte Verfahren wird durch zwei Erweiterungen ausgebaut. Zum einen wird ein Zellunterteilungsverfahren für die Rasterkarte entwickelt, das eine Ausweitung des Abdeckbereichs des texturierten Oberflächenmodells zur besseren Beschreibung von Hindernisgrenzen ermöglicht. Zum anderen wird ein etabliertes Verfahren zur Stereo-Hinderniserkennung bzw. zum Bestimmen des freien Raumes vor dem Sensor erweitert, indem die darin enthaltene Grundannahmen verallgemeinert werden. Dadurch wird das Verfahren für die vorliegende Problemstellung einsatzfähig. Die beiden Erweiterungen werden dann in das Verfahren zur Bestimmung der perspektivisch korrekten Vogelperspektivenansicht integriert [11].

Für die Ausführungen relevante Definitionen und Hinweise zu Notationen werden in Kapitel 2 eingeführt. Wichtige Vorarbeiten und der Stand der Technik, der für beide Großbereiche relevant ist, werden in Kapitel 3 vorgestellt. Die Arbeiten zur Schätzung des Spannungszustands von artikulierten Fahrzeugen werden in Kapitel 4 vorgestellt. Kapitel 5 beinhaltet die Ergebnisse zum Themenblock „Vogelperspektivenansicht“. Eine Zusammenfassung und das Fazit sind in Kapitel 6 zu finden.



## DEFINITIONEN UND NOTATIONEN

In den folgenden Abschnitten werden Schreibweisen und Definitionen eingeführt und erläutert, die für die vorgestellten Ergebnisse verwendet werden.

In dieser Arbeit wird der international übliche Punkt als Dezimaltrennzeichen verwendet. Das im Deutschen sonst übliche Komma impliziert eine uneindeutige Notation in mathematischen Ausdrücken und wird daher nicht verwendet.

## 2.1 KOORDINATENSYSTEME UND VEKTOREN

Für die Beschreibung von Orten und Richtungen in mehrdimensionalen Räumen werden Vektoren genutzt. Dabei wird innerhalb der vorliegenden Arbeit eine einheitliche Notation und Darstellung verwendet. Sofern nicht anders angegeben, liegt den Ausführungen ein kartesisches Koordinatensystem mit Orthonormalbasen entlang der Raumdimensionen zugrunde. Eine Position im 3-D-Raum wird beispielsweise als dreidimensionaler Spaltenvektor  $\mathbf{p} \in \mathbb{R}^3$  dargestellt.

Die Visualisierung der Koordinatensysteme hinsichtlich der jeweiligen Ursprünge und Orthonormalbasen wird durchgängig wie in Abbildung 2.1 verwendet. Es handelt sich um Rechtssysteme. In den folgenden Darstellungen sind die Färbungen der Achsen jeweils wie in der Abbildung gezeigt.

Bei der Verwendung mehrerer benannter Referenzkoordinatensysteme werden diese durch entsprechende Angabe definiert: Gegeben zwei Koordinatensysteme  $a$  und  $b$ , so beschreibt  $\mathbf{p}^a$  die Lage von  $\mathbf{p}$  relativ zu  $a$  und  $\mathbf{p}^b$  die Lage von  $\mathbf{p}$  relativ zu  $b$ .

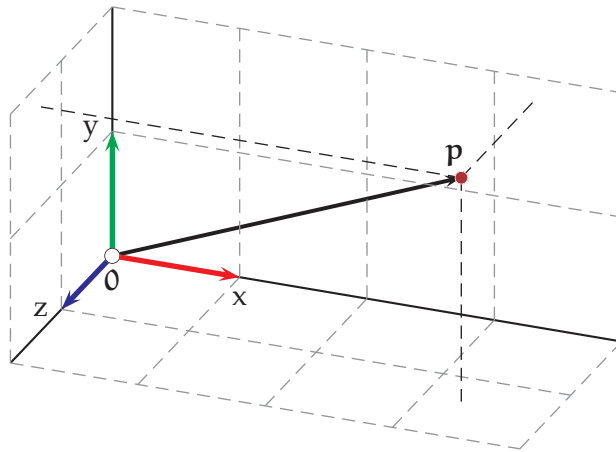
Der Nullvektor, d. h. ein Vektor mit der Länge 0, wird als  $\mathbf{0}$  notiert ( $\|\mathbf{0}\| = 0$ ):

$$\mathbf{0} = (a_1, \dots, a_n)^T \quad \text{mit} \quad a_i = 0, i \in \mathbb{N}, i \in [1; n], \mathbf{0} \in \mathbb{R}^n \quad .$$

Im Speziellen wird der Nullvektor ebenfalls für den Ursprung eines kartesischen Koordinatensystems genutzt. Die Schreibweise  $\mathbf{0}^a$  bezeichnet den Ursprung des Koordinatensystems  $a$ .

*Kreuzprodukt im 3-D-Raum*

Das Kreuzprodukt der beiden Vektoren  $\mathbf{p}, \mathbf{p}' \in \mathbb{R}^3$  im 3-D-Raum ist definiert als:



→ x-Achse (rot)    → y-Achse (grün)    → z-Achse (blau)

Abbildung 2.1: Beispiel für die Koordinatensystem-Darstellung eines kartesischen Koordinatensystems mit Orthonormalbasis

$$\mathbf{p} \times \mathbf{p}' = \begin{pmatrix} p_x \\ p_y \\ p_z \end{pmatrix} \times \begin{pmatrix} p'_x \\ p'_y \\ p'_z \end{pmatrix} = \begin{pmatrix} p_y p'_z - p_z p'_y \\ p_z p'_x - p_x p'_z \\ p_x p'_y - p_y p'_x \end{pmatrix} .$$

In manchen Situationen ist es sinnvoll, das Kreuzprodukt als Matrixmultiplikation zu notieren. Es wird folgende Schreibweise definiert:

$$\mathbf{p} \times \mathbf{p}' = [\mathbf{p}]_{\times} \cdot \mathbf{p}' \quad [\mathbf{p}]_{\times} = \begin{pmatrix} 0 & -p_z & p_y \\ p_z & 0 & -p_x \\ -p_y & p_x & 0 \end{pmatrix} .$$

### Skalarprodukt

Das Skalarprodukt zweier Vektoren  $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$  mit  $n \in \mathbb{N}$  wird mit dem Operator  $\circ$  notiert als:

$$\mathbf{a} \circ \mathbf{b} = a_1 \cdot b_1 + a_2 \cdot b_2 + \cdots + a_n \cdot b_n$$

### Homogene Koordinaten

Homogene Koordinaten finden insbesondere in der perspektivischen Geometrie ihre Anwendung. Sie vereinfachen die Anwendung von affinen Transformationen, indem eine zusätzliche Dimension an die entsprechenden Matrizen und Vektoren angefügt wird. Gegeben sei ein Vektor  $\mathbf{a} \in \mathbb{R}^n$  mit  $n \geq 1$ . Der Vektor  $\tilde{\mathbf{a}} \in \mathbb{P}^n$  beschreibt den korrespondierenden homogenen Vektor. Es gilt:

$$\begin{aligned}\mathbf{a} &= (a_1, a_2, \dots, a_n)^T, \quad \mathbf{a} \in \mathbb{R}^n, \quad n \in \mathbb{N} \\ \Rightarrow \tilde{\mathbf{a}} &= (s \cdot a_1, s \cdot a_2, \dots, s \cdot a_n, s)^T = s \cdot (a_1, a_2, \dots, a_n, 1)^T \\ \tilde{\mathbf{a}} &\in \mathbb{P}^n, \quad s \in \mathbb{R} \quad .\end{aligned}$$

Diese Schreibweise wird im Folgenden angewendet. Zusätzlich werden stellenweise *implizite* Konvertierungen zwischen homogenen und nicht-homogenen Vektoren vorgenommen, um die Lesbarkeit der Formelnotation zu verbessern.

## 2.2 POSITION UND ORIENTIERUNG

Zusätzlich zur reinen Position eines Objektes innerhalb eines Raumes ist es sinnvoll, die Orientierung des Objektes mit anzugeben. Es bietet sich an, die damit verbundene Rotation des Objektes relativ zu den Orthonormalbasen des Objektkoordinatensystems zu definieren. Diese Rotation wird mit einem Einheitsquaternion im Hamilton-Raum  $\mathbb{H}$  beschrieben. Einheitsquaternionen haben die Länge 1 und werden in der Menge  $SO(3)$  zusammengefasst. Die Menge  $SO(3)$  beschreibt eine Einheits-3-Sphäre im vierdimensionalen euklidischen Raum. Es gilt  $SO(3) \subset \mathbb{H}$ .

Ein Einheitsquaternion kann für die Beschreibung einer Rotation im 3-D-Raum verwendet werden: Eine Rotation  $\mathbf{r} \in SO(3)$  beschreibt die Verdrehung zweier Koordinatensysteme zueinander. Die zu einem Rotationsquaternion  $\mathbf{r} \in SO(3)$  korrespondierende Rotationsmatrix wird durch die Funktion  $\text{rot} : SO(3) \rightarrow \mathbb{R}^{3 \times 3}$  definiert (mit den Elementen  $1, i, j, k$  als Basis):

$$\begin{aligned}\mathbf{r} &= r_r + i \cdot r_i + j \cdot r_j + k \cdot r_k \\ \text{rot}(\mathbf{r}) &= \begin{pmatrix} 1 - 2(r_j^2 + r_k^2) & 2(r_i r_j - r_k r_r) & 2(r_i r_k + r_j r_r) \\ 2(r_i r_j + r_k r_r) & 1 - 2(r_i^2 + r_k^2) & 2(r_j r_k - r_i r_r) \\ 2(r_i r_k - r_j r_r) & 2(r_j r_k + r_i r_r) & 1 - 2(r_i^2 + r_j^2) \end{pmatrix} .\end{aligned}$$

Zur vollständigen Beschreibung der Lage eines Objektes im Raum, genannt *Pose*, müssen folglich Position und Orientierung kombiniert werden. Die resultierende Pose  $\mathbf{a} = \langle \mathbf{p}, \mathbf{r} \rangle$  mit  $\mathbf{a} \in \mathbb{R}^3 \times SO(3)$  fasst die benötigten Informationen zusammen. Die vorgestellte Schreibweise lehnt sich an die von Kaess [Kae15] vorgeschlagene Notation an.

Um die Lesbarkeit der Formelnotationen zu erhöhen, werden im Folgenden Schreibweisen definiert, die eine Konvertierung einer Pose  $\mathbf{a} = \langle \mathbf{p}, \mathbf{r} \rangle$  in eine korrespondierende Transformationsmatrix beschreiben. Für Funktion  $\mathbf{T}(\mathbf{a}) : \mathbb{R}^3 \times SO(3) \rightarrow \mathbb{P}^{3 \times 3}$  gilt:

$$\mathbf{T}(\mathbf{a}) = \begin{pmatrix} \text{rot}(\mathbf{r}) & \mathbf{p} \\ \mathbf{0}^T & 1 \end{pmatrix} .$$

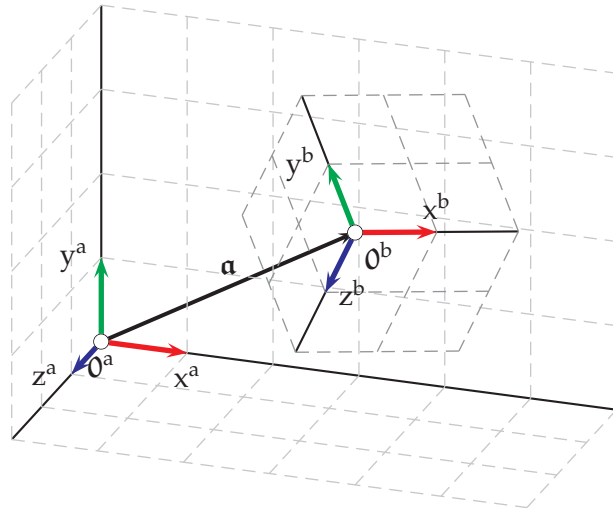


Abbildung 2.2: Beschreibung der relativen Lage zwischen zwei Koordinatensystemen mittels Pose: Das Koordinatensystem a wird mit der Pose  $\mathbf{a}$  in das Koordinatensystem b transformiert.

Für Posen werden Operatoren definiert, die einer vereinfachten Schreibweise bei der Verkettung von Posen dienen:

$$\begin{aligned} \mathbf{a}, \mathbf{b} \in \mathbb{R}^3 \times \text{SO}(3) \quad \mathbf{a} &= \langle \mathbf{p}_a, \mathbf{r}_a \rangle \quad \mathbf{b} = \langle \mathbf{p}_b, \mathbf{r}_b \rangle \\ \mathbf{a} \oplus \mathbf{b} &= \langle \mathbf{p}_a + \text{rot}(\mathbf{r}_a) \cdot \mathbf{p}_b, \mathbf{r}_a \cdot \mathbf{r}_b \rangle \\ \mathbf{a} \ominus \mathbf{b} &= \langle \text{rot}(\bar{\mathbf{r}}_b) \cdot (\mathbf{p}_a + \mathbf{p}_b), \bar{\mathbf{r}}_b \cdot \mathbf{r}_a \rangle \quad . \end{aligned}$$

Mit einer Pose können Koordinaten zwischen Koordinatensystemen transformiert werden (siehe Abbildung 2.2). Die Pose beschreibt in diesem Zusammenhang eine Starrkörpertransformation. Für die benötigten Umrechnungen werden die Operatoren  $\oplus$  und  $\ominus$  verwendet. Die Inverse von  $\mathbf{a}$  ist entsprechend  $-\mathbf{a}$ . Es gilt  $\mathbf{a} \oplus -\mathbf{a} = \mathbf{o}$  mit

$$\mathbf{o} \in \mathbb{R}^3 \times \text{SO}(3), \quad \mathbf{o} = \langle (0, 0, 0)^T, 1 \rangle$$

der Identitätspose.

Die Transformation eines Punktes  $\mathbf{p}^a$  im Koordinatensystem a in das Koordinatensystem b wird wie folgt beschrieben: Sei  $\mathbf{a}$  die Transformation zwischen den Koordinatensystemen a und b (analog zu Abbildung 2.2). Als vereinfachte Schreibweise wird definiert:

$$\mathbf{p}^b = \mathbf{T}(\mathbf{a}) \cdot \mathbf{p}^a = \mathbf{T}_{a \rightarrow b}(\mathbf{p}^a) \quad .$$

### 2.3 DEFINITION MATHEMATISCHER FUNKTIONEN

Zur Vereinfachung von Schreibweisen in mathematischen Ausdrücken werden für die folgenden Ausführungen Funktionen definiert:

Die Funktion  $\text{rotate}(\mathbf{q}_1, \beta, \mathbf{q}_2) : \mathbb{R}^2 \times \mathbb{R} \times \mathbb{R}^2 \rightarrow \mathbb{R}^2$  beschreibt die Rotation des Punktes  $\mathbf{q}_1 \in \mathbb{R}^2$  um den Punkt  $\mathbf{q}_2 \in \mathbb{R}^2$  um den Winkel  $\beta \in \mathbb{R}$  in der Ebene. Es gilt:

$$\text{rotate}(\mathbf{q}_1, \beta, \mathbf{q}_2) = \begin{pmatrix} \cos(\beta) & -\sin(\beta) \\ \sin(\beta) & \cos(\beta) \end{pmatrix} \cdot (\mathbf{q}_1 - \mathbf{q}_2) + \mathbf{q}_2 \quad .$$

Die Rotationsmatrix im 3-D-Raum um eine durch einen Vektor  $\mathbf{a} \in \mathbb{R}^3$  definierte Drehachse um den Winkel  $\beta$  wird durch die Funktion  $\text{angleaxis}(\beta, \mathbf{a})$  mit  $\text{angleaxis} : \mathbb{R} \times \mathbb{R}^3 \rightarrow \mathbb{R}^{3 \times 3}$  beschrieben.

Beschreibe  $\mathbf{q}^\perp : \mathbb{R}^2 \rightarrow \mathbb{R}^2$  das Orthokomplement eines zweidimensionalen Vektors. Es gilt:

$$\mathbf{q}^\perp = \begin{pmatrix} q_x \\ q_y \end{pmatrix}^\perp = \begin{pmatrix} -q_y \\ q_x \end{pmatrix}, \quad \mathbf{q}^\perp \circ \mathbf{q} = 0 \quad .$$

Die Gaußsche Verteilung (auch als Normal-Verteilung bezeichnet) ist ein weitverbreiteter Ansatz, um Wahrscheinlichkeitsverteilungen zu beschreiben. Die Gaußsche Dichteverteilung beschreibt eine kontinuierliche Glockenkurve, die vom Mittelwert  $\mu \in \mathbb{R}$  und der Standardabweichung  $\sigma \in \mathbb{R}^+$  abhängig ist. Als Funktionssymbol für die Gaußsche Glockenkurve wird  $\mathcal{N}_{\mu, \sigma}$  verwendet. Die Gaußsche Dichtefunktion  $\mathcal{N}_{\mu, \sigma} : \mathbb{R} \rightarrow \mathbb{R}$  ist wie folgt definiert:

$$\mathcal{N}_{\mu, \sigma}(x) = \frac{1}{\sigma\sqrt{2\pi}} \cdot e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \quad .$$

## 2.4 FEHLERMASSE

Bei der Beurteilung von Messreihen sind Fehlermaße notwendig, die eine Aussage zur erreichten Qualität treffen.

### 2.4.1 Statistische Fehlermaße

Sei ein eindimensionaler Referenzwert  $s \in \mathbb{R}$  gegeben, zu denen ein Messwert  $s' \in \mathbb{R}$  existiert. Ein Tupel  $(s, s') \in \mathbb{R}^2$  beschreibe ein Paar von Referenz- und Messwert.

Die Abweichung in einer Messung von dem erwarteten Referenzwert wird durch die Differenz beider Werte beschrieben. Die Funktion  $\Delta_E : \mathbb{R}^2 \rightarrow \mathbb{R}$  für das Tupel  $(s, s')$  wird definiert als:

$$\Delta_E(s, s') = s - s' \quad .$$

Die Menge  $A$  über mehrere Tupel sei eine Messreihe mit  $n$  Messungen. Um Fehler über die Messreihe auszudrücken werden die Messtupel in der Menge gemeinsam betrachtet. Es gilt:

$$A = \{(s_i, s'_i) \mid i \in [1; n]\} \quad .$$

Die Fehler aller Tupel in der Menge werden mit  $\Delta_{EA}$  beschrieben:

$$\Delta_{EA} = \{\Delta_E(s_i, s'_i) \mid i \in [1; n]\} = \{s_i - s'_i \mid i \in [1; n]\}$$

Die Menge  $A$  kann verwendet werden, um eine Aussage über die Qualität der Messreihe zu berechnen. Im Einzelnen werden folgende Fehlermaße definiert:

1. *Maximaler absoluter Fehler:*

$$\max |\Delta_{EA}| = \max \{|s_i - s'_i| \mid i \in [1; n]\} \quad .$$

2. *Mittlerer absoluter Fehler:*

$$\overline{|\Delta_{EA}|} = \frac{1}{n} \sum_{i=1}^n |s_i - s'_i| \quad .$$

3. *Varianz des mittleren absoluten Fehlers:*

$$\text{Var}(\overline{|\Delta_{EA}|}) = \frac{1}{n} \sum_{i=1}^n (|s_i - s'_i| - \overline{|\Delta_{EA}|})^2 \quad .$$

#### 2.4.2 Earth Mover's Distance

Die *Earth Mover's Distance (EMD)* [Mon81, RTGoo] ist ein Fehlermaß, das in der Transport-Theorie begründet ist und auf der Lösung des *Transportproblems* [Hit41, Kan42] basiert. Der Name entstammt der bildlichen Vorstellung hinter dem Ansatz: Gegeben ein unebenes Gelände, in dem Löcher und Erdhaufen anzutreffen sind. Ziel ist es, das Gelände einzuebnen, wobei möglichst wenig Erdvolumen bewegt werden soll, d. h. der Fluss und die damit verbundene Arbeit soll minimal bleiben, um den Gesamtaufwand zu reduzieren.

Gegeben zwei Mengen  $A, B$  mit  $n, m \in \mathbb{N}$  und  $|A| = n$  bzw.  $|B| = m$ :

$$A = \{(\mathbf{a}_i, \chi_{\mathbf{a}_i}) \mid i \in \mathbb{N}, i \in [1; n]\}$$

$$B = \{(\mathbf{b}_j, \chi_{\mathbf{b}_j}) \mid j \in \mathbb{N}, j \in [1; m]\} \quad .$$

Weiterhin beschreibe  $d_{ij}$  die Distanz zwischen  $\mathbf{a}_i$  und  $\mathbf{b}_j$  und  $\chi_{\mathbf{a}_i}$  bzw.  $\chi_{\mathbf{b}_j}$  ein jeweiliger Gewichtungsfaktor. Die Kosten bzw. die zu erwartende Arbeit (Work) der Einebnung werden durch Aufsummierung der Produkte aus Distanzen  $d_{ij}$  und dem damit verbundenen Fluss  $f_{ij}$  beschrieben. Die Matrix  $F \in \mathbb{R}^{n \times m}$  beschreibe die Gesamtheit der Flüsse  $f_{ij}$ :

$$\text{Work}(A, B, F) = \sum_{j=1}^m \sum_{i=1}^n d_{ij} f_{ij} \quad .$$



Ziel ist es, die Flüsse  $f_{ij}$  so zu wählen, dass die Arbeit minimiert wird. Es ergibt sich ein Minimierungsproblem:

$$\operatorname{argmin}_{\mathbf{F}} (\operatorname{Work}(A, B, \mathbf{F})) \quad .$$

Für das Optimierungsproblem gelten folgende Nebenbedingungen [RTGoo]:

$$\begin{aligned} f_{ij} &\geq 0 && i \in [1; n], j \in [1; m] \\ \sum_{j=1}^m f_{ij} &\leq \chi_{\mathbf{a}_i} && i \in [1; n] \\ \sum_{i=1}^n f_{ij} &\leq \chi_{\mathbf{b}_j} && j \in [1; m] \\ \sum_{j=1}^m \sum_{i=1}^n f_{ij} &= \min \left( \sum_{i=1}^n \chi_{\mathbf{a}_i}, \sum_{j=1}^m \chi_{\mathbf{b}_j} \right) && . \end{aligned}$$

Nach Lösung des Transportproblems wird die *EMD* unter Verwendung des optimalen Flusses  $\mathbf{F}$  berechnet. Der Wert wird dabei durch den Gesamtfluss normalisiert:

$$\operatorname{EMD}(A, B) = \frac{\sum_{j=1}^m \sum_{i=1}^n d_{ij} f_{ij}}{\sum_{j=1}^m \sum_{i=1}^n f_{ij}} = \frac{\sum_{j=1}^{|B|} \sum_{i=1}^{|A|} d_{ij} f_{ij}}{\sum_{j=1}^{|B|} \sum_{i=1}^{|A|} f_{ij}} \quad .$$

Die *EMD* kann als Distanzmaß für diskrete Eingabemengen unterschiedlicher Größen verwendet werden.



Für das angestrebte Vorhaben müssen eine Vielzahl von bereits geleisteten Vorarbeiten einbezogen werden. Die notwendigen Grundlagen und der aktuelle Stand der Technik werden im folgenden Kapitel dargestellt und umfassen die für beide Großteile dieser Arbeit relevanten Aspekte.

### 3.1 KAMERAS ALS OPTISCHE SENSOREN

Eine Kamera (von lateinisch *camera obscura*, „dunkle Kammer“) bildet Objekte ab, indem sie das von ihnen ausgestrahlte bzw. reflektierte Licht aufnimmt und es auf eine Fläche projiziert. Die Fläche ist meist mit fotosensitiven Materialien ausgestattet, die den Lichteindruck konservieren können.

Bei digitalen Kameras kommen dabei meist flächige Bildsensoren zum Einsatz, die das einfallende Licht in einem Raster abtasten und die gewonnenen Informationen in einem digitalen Format bereitstellen. Das verwendete Raster ist häufig ein äquidistantes orthogonales zweidimensionales Gitter, in dem die einzelnen Bildpunkte, die Pixel (von englisch *picture element*, „Bildelement“) genannt werden, angeordnet sind. Die Dimension dieses Rasters – und damit die Bildauflösung der Kamera – wird mit den Variablen  $w, h \in \mathbb{N}$  beschrieben. Die Variable  $w$  bezeichnet dabei die Breite, die Variable  $h$  die Höhe des abgetasteten Bildes. Die Auflösung einer Kamera entspricht dabei der Anzahl der Pixel im Raster und wird mit  $w \times h$  angegeben.

Zu jedem Pixel, das an einem festen Ort  $\mathbf{q}^i \in \mathbb{R}^2$  auf der Sensorfläche zu finden ist und eine feste (meist rechteckige oder quadratische) Form und Größe hat, werden – durch Messung des Lichts – Intensitätswerte für die eingefallene Strahlung zugeordnet. Folglich ist eine Kamera ein fotosensitives radiometrisches Messinstrument.

Das in dieser Arbeit verwendete Abbildungsmodell ist das *Lochkameramodell*. Dabei wird die Öffnung der Kamera als eine kleine Lochblende im Kameragehäuse angenommen. Das Licht fällt durch die Öffnung ein und projiziert so ein Abbild der beobachteten Objekte auf die Bildebene (siehe Abbildung 3.1). Die Öffnung bzw. das Loch werden dabei als infinitesimal klein, d. h. als Punkt-förmig, angenommen.

Die dabei auftretende Projektion ist perspektivisch. Für die Abbildungsgleichung des Lochkameramodells gilt bei gegebener Brennweite  $F \in \mathbb{R}$ :

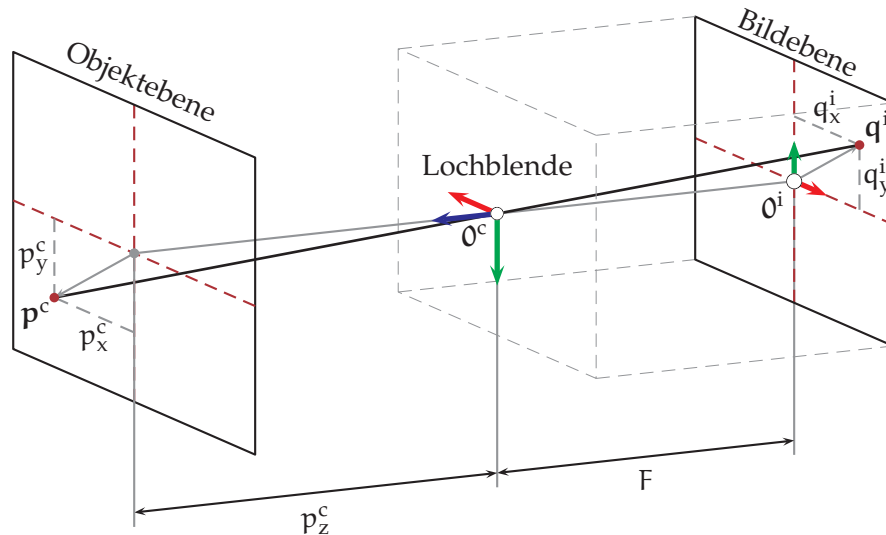


Abbildung 3.1: Projektion eines beobachteten Objektpunktes  $\mathbf{p}^c$  auf die Bildebene nach dem Lochkameramodell

$$\begin{aligned} q_x^i &= F \cdot p_x^c \cdot (p_z^c)^{-1} & \mathbf{p}^c &= \begin{pmatrix} p_x^c & p_y^c & p_z^c \end{pmatrix}^T \in \mathbb{R}^3 \\ q_y^i &= F \cdot p_y^c \cdot (p_z^c)^{-1} & \mathbf{q}^i &= \begin{pmatrix} q_x^i & q_y^i \end{pmatrix}^T \in \mathbb{R}^2 \end{aligned} .$$

In der Kamera wird die dreidimensionale Umgebung auf die zweidimensionale Bildebene projiziert (Dimensionsreduktion). Der Abbildungsprozess kann daher als Abbildungsfunktion wie folgt definiert werden und beschreibt, wie ein beliebiger Punkt in der Umgebung auf die Bildebene abgebildet wird:

$$\mathcal{P}(\mathbf{p}^c) = F \cdot \left( p_x^c \cdot (p_z^c)^{-1}, p_y^c \cdot (p_z^c)^{-1} \right)^T = \mathbf{q}^i \quad \text{mit} \quad \mathcal{P} : \mathbb{R}^3 \rightarrow \mathbb{R}^2 .$$

Bildlich gesprochen wird der Punkt in der 3-D-Welt durch einen Sichtstrahl mit der 2-D-Kamera-Bildebene geschnitten. Die Verschiebung auf der gedachten Sichtgeraden (Vorstellung bei Vernachlässigung von Verzerrungseffekten durch Linsen o. Ä.) entspricht der Normalisierung eines skalaren Verschiebungskoeffizienten entlang der Geraden. Es bietet sich daher an, für die Abbildungsfunktion homogene Koordinaten zu verwenden. Wird der Vektor  $\mathbf{p}^c$  als homogener Vektor niedrigerer Dimension interpretiert, so entspricht  $z^c$  der homogenen Komponente. Eine Normalisierung auf die homogene Komponente 1 entspricht dem Teilen durch  $z^c$  in der Abbildungsfunktion  $\mathcal{P}$ .

### 3.2 KAMERAKALIBRIERUNG

Die in Kameraobjektiven verwendeten Linsen führen zu systematischen geometrischen Abbildungsfehlern, die in der Vorverarbeitung

korrigiert werden müssen, um unerwünschte Effekte zu vermeiden. Zu diesen Effekten gehören Verzeichnungen, die durch die Linsengeometrie bedingt auftreten. Durch die Lichtbrechung werden die geraden Lichtstrahlen abgelenkt und treffen nicht die eigentlich erwarteten Punkte im Bild.

Durch Fertigungsungenauigkeiten bei der Produktion des optischen Systems treten zudem Abweichungen, wie z. B. eine leicht verschobene Positionierung des Chips, auf, die ebenfalls vor der weiteren Verarbeitung korrigiert werden müssen. Es ist daher auch erforderlich, die *intrinsischen* Kameraparameter zu ermitteln.

Da diese Effekte bei unveränderten Parametern (Blende, Position der Linse) als konstant angenommen werden können, kann die Verzerrung durch eine Kalibrierung bestimmt und danach herausgerechnet werden.

Tsai und Lenz haben grundlegende Arbeit im Bereich der Kamerakalibrierung geleistet [Tsa87, LT88]. Nach der Aufnahme von definierten Kalibriermustern, wie z. B. einem Schachbrettmuster, können intrinsische Kameraparameter geschätzt und die Verzeichnung bei der Abbildung bestimmt werden. Eine Symmetriebedingung in Bezug auf die Linse wird angenommen, um die Verzerrungsparameter zu modellieren. Auf Grundlage dieser Arbeiten hat Zhang [Zhao] ein erweitertes Kalibrierverfahren entwickelt, das ebenfalls Aufnahmen von Kalibriermustern als Eingabe verwendet. Heikkilä und Silvén [HS97] schlagen ein weiteres Verfahren für die intrinsische Kalibrierung vor, das ebenfalls Aufnahmen von bekannten Geometrieobjekten nutzt. Beide Verfahren werden in zahlreichen Anwendungen in der Bildverarbeitung eingesetzt.

Die intrinsischen Kameraparameter werden mit einer Kameramatrix  $\mathbf{K} \in \mathbb{P}^{2 \times 2}$  beschrieben:

$$\mathbf{K} = \begin{pmatrix} k_x & s & c_x \\ 0 & k_y & c_y \\ 0 & 0 & 1 \end{pmatrix} .$$

wobei  $(k_x, k_y)^T \in \mathbb{R}^2$  die Kamerakonstanten in  $x$ - bzw.  $y$ -Richtung und  $(c_x, c_y) \in \mathbb{R}^2$  den Hauptpunkt der Kamera beschreiben. Die Kamera-Konstanten skalieren damit die Brennweite mit der Dimension eines einzelnen Pixels, das nicht unbedingt quadratisch sein muss. Die Schiefe der beiden Bildachsen wird durch den Parameter  $s \in \mathbb{R}$  beschrieben und wird oft unter der Annahme, dass die Achsen des Pixelrasters orthogonal verlaufen, mit  $s = 0$  angenommen. Mit  $a_x, a_y \in \mathbb{R}$  den Abmessungen eines Pixels gilt:

$$k_x = \frac{F}{a_x} \quad \text{und} \quad k_y = \frac{F}{a_y} .$$

Die Kameramatrix  $\mathbf{K}$  fasst die Abbildungseigenschaften einer Kamera zusammen. Die Multiplikation der Matrix  $\mathbf{K}$  mit einem als ho-

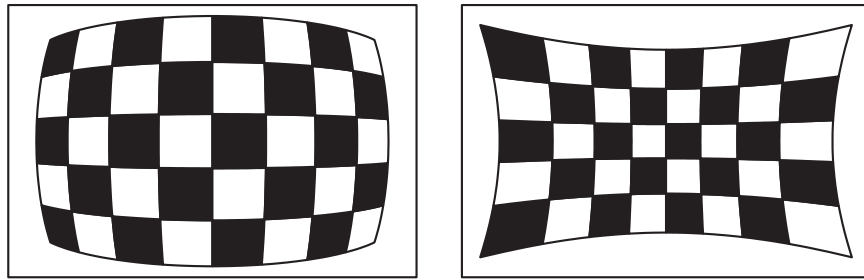


Abbildung 3.2: Beispiele für geometrische Verzerrungen anhand der Abbildung eines Schachbrettmusters. Links: Tonnenverzerrung; Rechts: Kissenverzerrung

mogene Koordinate interpretierten Punkt  $p^c$  bildet den Punkt in Pixelkoordinaten (Koordinatensystem  $p$ ) auf die Bildebene ab und stellt den in Kapitel 3.1 beschriebenen Zusammenhang in Kombination mit der Abbildung auf das Pixelraster dar:

$$\tilde{q}^p = K \cdot p^c \quad .$$

### 3.2.1 Verzerrungskorrektur

Durch die Verwendung von Linsen treten Verzerrungen auf, die korrigiert werden müssen, um die Kamera als Messinstrument verwenden zu können. Dabei werden in der Realität parallele Linien verformt dargestellt. Abbildung 3.2 zeigt Beispiele für Verzerrungen durch Linsen.

Zur Korrektur unterschiedlicher Verzerrungsarten und -grade existieren etablierte Verfahren, u.a. das bereits erwähnte Verfahren nach Tsai und Lenz [Tsa87], Zhang [Zhao] bzw. Heikkilä und Silvén [HS97]. Unter Aufrechterhaltung der Symmetrieannahme wird die Linsenverzerrung (je nach Verfahren) mit radialen und tangentialen Termen beschrieben. Um bestimmte Linsencharakteristika zu beschreiben, nutzen die Autoren Koeffizienten, die in einem Polynom die einzelnen Terme gewichten.

Die vorgestellten Verfahren werden durchgängig durch reelle Verzerrungskoeffizienten parametrisiert. Um die Formulierung der Ausführungen im Folgenden generisch zu halten, werden die Koeffizienten  $\kappa_i \in \mathbb{R}$  im Vektor  $\kappa \in \mathbb{R}^n$  zusammengefasst, ohne weiter auf die genaue Semantik eines einzelnen Koeffizienten einzugehen, da diese vom eingesetzten Verfahren abhängt.

Das Modell nach Zhang [Zhao] modelliert dabei die Linsenverzerrung als radiale Verzerrung. Die Berechnung eines idealen, unverzerrten Bildpunktes  $q' \in \mathbb{R}^2$  aus einem bei der Aufnahme durch die Linse

verzerrten Punkt  $\mathbf{q} = (q_x, q_y)^T$  wird mit folgender Modellgleichung beschrieben [Bro71, GS94]:

$$\begin{aligned} q'_x &= q_x + q_x \left( \kappa_1 \cdot (q_x^2 + q_y^2) + \kappa_2 \cdot (q_x^2 + q_y^2)^2 \right) \\ q'_y &= q_y + q_y \left( \kappa_1 \cdot (q_x^2 + q_y^2) + \kappa_2 \cdot (q_x^2 + q_y^2)^2 \right) . \end{aligned}$$

Das Verfahren nach Heikkilä und Silvén [HS97] kombiniert radiale mit tangentialen Koeffizienten, um die Linsenverzerrung zu modellieren. Eine Modellgleichung für den Fall, dass mindestens zwei radiale Koeffizienten ( $\kappa_i$  mit  $i \in \{1, 2, 3, 4, 5, 6\}$  und je nach Konfiguration für  $i > 2$  mit  $\kappa_i = 0$ ) mit zwei tangentialen Koeffizienten ( $\kappa_7, \kappa_8$ ) kombiniert werden, um die Linsenverzerrung zu modellieren, ist gegeben mit:

$$\begin{aligned} a &= q_x^2 + q_y^2 \\ q'_x &= q_x \cdot \frac{1 + \kappa_1 a^2 + \kappa_2 a^4 + \kappa_3 a^6}{1 + \kappa_4 a^2 + \kappa_5 a^4 + \kappa_6 a^6} + 2 \cdot \kappa_7 q_x q_y + \kappa_8 (a^2 + 2q_x^2) \\ q'_y &= q_y \cdot \frac{1 + \kappa_1 a^2 + \kappa_2 a^4 + \kappa_3 a^6}{1 + \kappa_4 a^2 + \kappa_5 a^4 + \kappa_6 a^6} + \kappa_7 (a^2 + 2q_y^2) + 2 \cdot \kappa_8 q_x q_y . \end{aligned}$$

Die Bestimmung der Verzerrungskoeffizienten kann in einem gemeinsamen Optimierungsproblem zusammen mit der Kameraintrinsik unter Verwendung der Aufnahmen bekannter Geometrien (z. B. Schachbrettmuster) geschätzt werden.

Die bereits vorgestellten Verfahren sind für mäßige Verzerrungen geeignet. Objektive mit stärkeren bzw. anders zu modellierenden Verzerrungen erfordern angepasste Verfahren. Für weitwinklige Objektive und omnidirektionale Kameras eignet sich das Verfahren nach Scaramuzza [Sca07]. In diesem Ansatz wird eine komplexere geometrische Modellierung für die Linsengeometrie zu Grunde gelegt.

Für spezielle Kameramodalitäten wurden eine Vielzahl von unterschiedlichen Kalibrierverfahren veröffentlicht, die die jeweiligen Belange der Modalität abbilden, z. B. für katadioptrische Kameras [GD01] oder plenoptische Kameras [DPW13]. Auf eine detaillierte Darstellung wird an dieser Stelle verzichtet, da erweiterte Kameramodalitäten in der vorliegenden Arbeit keine Anwendung finden.

Die Wahl des für den konkreten Fall richtigen Kalibrierverfahrens ist abhängig von der eingesetzten Optik. Meist ist nicht nur ein Verfahren für das vorliegende Szenario geeignet. An dieser Stelle wird nicht näher auf die Wahl der Kalibriermethode bzw. -modellierung eingegangen. In der vorliegenden Arbeit werden die Verfahren nach Zhang [Zha00], Heikkilä und Silvén [HS97] und nach Scaramuzza [Sca07] angewandt.

## 3.3 PERSPEKTIVISCHE 2-D-VERZERRUNG

Ein weit verbreiteter Ansatz für die Berechnung von virtuellen Kameraansichten beruht auf der Verwendung von perspektivischen Transformationen mittels Homographien, wie u. a. von Hartley und Zisserman [HZ03] beschrieben. Dabei werden 2-D Punkte, die sich auf einer Ebenen (z. B. der Bildebenen der Kamera) befinden, auf eine andere Ebene (z. B. die Bildebene der virtuellen Kamera) transformiert. Als Grundannahme wird dabei verwendet, dass die zu transformierenden Punkte alle auf einer Ebene im 3-D-Raum liegen.

Für die Transformation wird aus mindestens vier Punktkorrespondenzen in den Bildern eine Homographie-Matrix  $\mathbf{H} \in \mathbb{P}^{2 \times 2}$  geschätzt, die dann zur Transformation des gesamten Bildbereichs verwendet wird. Für eine statische Relativpose zwischen den beiden Kameras und bei unveränderten Kameraparametern kann die Homographie-Matrix  $\mathbf{H}$  als konstant angenommen werden. Für die Transformation eines Punktes  $\mathbf{q}^i \in \mathbb{R}^2$  auf der Projektionsebene von Kamera  $i$  auf einen Referenzpunkt  $\mathbf{q}^j$  im Bild von Kamera  $j$  gilt folgender Zusammenhang:

$$\tilde{\mathbf{q}}^j = \mathbf{H} \cdot \tilde{\mathbf{q}}^i \quad .$$

Einige Publikationen beschäftigen sich mit Erweiterungen zur Schätzung der Homographie aus Punktkorrespondenzen. Agarwal, Jawahar und Narayanan [AJN05] stellen verschiedene Ansätze vor und fassen diese zusammen. Einen einfachen Ansatz stellt die *Direkte Lineare Transformation* (DLT) mit Punktkorrespondenzen, Linienkorrespondenzen oder einer Kombination aus beiden dar.

Vincent und Laganière [VL01] stellen ein Verfahren vor, das einen *Random Sample Consensus* (RANSAC) für unkalibrierte Bildpaare nutzt, um geeignete Punktkorrespondenzen zu identifizieren. Der RANSAC sorgt dabei für eine Steigerung der Robustheit gegenüber falsch identifizierter Punktkorrespondenzen.

## 3.3.1 Berechnung einer Homographie

Im Folgenden wird die exemplarische Berechnung einer Homographie anhand einer Direkten Linearen Transformation erläutert. Die Notation lehnt sich dabei an die von Hartley und Zisserman [HZ03] an.

Zur Schätzung einer Homographie-Matrix  $\mathbf{H} \in \mathbb{P}^{2 \times 2}$  sind mindestens vier Punktkorrespondenzen erforderlich. Seien 2-D-Punktkorrespondenzen (in Bildkoordinaten) in der Menge  $A$  gegeben:

$$A = \{(\mathbf{a}_i, \mathbf{b}_i) \mid \mathbf{a}_i, \mathbf{b}_i \in \mathbb{R}^2\} \quad \text{mit} \quad |A| \geq 4 \quad .$$

Wie bereits am Beispiel der Kameratransformation oben beschrie-



ben, wird die Transformation zwischen den Punktkorrespondenzen beschrieben durch den Zusammenhang:

$$\tilde{\mathbf{b}}_i = \mathbf{H}\tilde{\mathbf{a}}_i \quad \text{bzw.} \quad \tilde{\mathbf{b}}_i \times \mathbf{H} \cdot \tilde{\mathbf{a}}_i = \mathbf{0} \quad .$$

Die Matrixmultiplikation lässt sich in eine zeilenweise Multiplikation mit den Zeilen der Homographie-Matrix  $\mathbf{H}$  umformen. Die Vektoren  $\mathbf{b}_1^T, \mathbf{b}_2^T, \mathbf{b}_3^T$  seien die Zeilen der  $3 \times 3$ -Matrix:

$$\mathbf{H} \cdot \tilde{\mathbf{a}}_i = \begin{pmatrix} \mathbf{b}_1^T \cdot \tilde{\mathbf{a}}_i \\ \mathbf{b}_2^T \cdot \tilde{\mathbf{a}}_i \\ \mathbf{b}_3^T \cdot \tilde{\mathbf{a}}_i \end{pmatrix} \quad .$$

Eingesetzt ergibt sich die Gleichung mit  $\tilde{\mathbf{b}}_i = (x_i, y_i, s_i)^T$  zu:

$$\begin{pmatrix} \mathbf{0}^T & -s_i \cdot \tilde{\mathbf{a}}_i^T & y_i \tilde{\mathbf{a}}_i^T \\ s_i \cdot \tilde{\mathbf{a}}_i^T & \mathbf{0}^T & -x_i \cdot \tilde{\mathbf{a}}_i^T \\ -y_i \cdot \tilde{\mathbf{a}}_i^T & x_i \cdot \tilde{\mathbf{a}}_i^T & \mathbf{0}^T \end{pmatrix} \begin{pmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \mathbf{b}_3 \end{pmatrix} = \mathbf{A}_i \cdot \mathbf{b} = \mathbf{0} \quad .$$

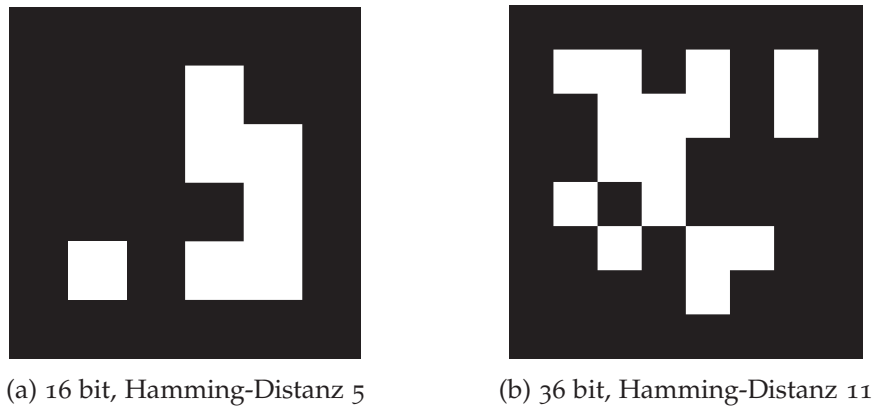
Die Matrix  $\mathbf{A}_i$  hat die Dimension  $3 \times 9$  und ist mit bekannten Werten besetzt. Der 9-D-Vektor  $\mathbf{b}$  enthält die Unbekannten des linearen Gleichungssystems. Da es sich bei den verwendeten Punkten  $\tilde{\mathbf{a}}_i$  und  $\tilde{\mathbf{b}}_i$  um homogene Koordinaten handelt, sind nur zwei der drei Gleichungen linear unabhängig. Die dritte Gleichung wird daher nicht für die Berechnung der Homographie-Matrix  $\mathbf{H}$  verwendet. Als Gleichungssystem ergibt sich mit der  $2 \times 9$ -Matrix  $\mathbf{A}_i$ :

$$\begin{pmatrix} \mathbf{0}^T & -s_i \cdot \tilde{\mathbf{a}}^T & y_i \cdot \tilde{\mathbf{a}}^T \\ s_i \cdot \tilde{\mathbf{a}}^T & \mathbf{0}^T & -x_i \cdot \tilde{\mathbf{a}}^T \end{pmatrix} \cdot \mathbf{b} = \mathbf{0} \quad .$$

Die homogene Komponente kann trivial auf  $s = 1$  gesetzt werden, sodass die x- und y-Komponenten den Pixelkoordinaten des Bildpunktes entsprechen. Mit jeder Punktkorrespondenz kann eine Matrix  $\mathbf{A}_i$  aufgestellt werden, die zwei linear unabhängige Gleichungen definiert. Die einzelnen Matrizen  $\mathbf{A}_i$  werden in Matrix  $\mathbf{A}$  kombiniert:

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \vdots \\ \mathbf{A}_n \end{pmatrix} \quad .$$

Die Unbekannten der Homographie-Matrix  $\mathbf{H}$  werden durch den zusammengesetzten Vektor  $\mathbf{b}$  repräsentiert. Die Homographie-Matrix  $\mathbf{H}$  hat neun Einträge und den Rang 8, da sie einen Skalierungsfaktor enthält. Für den Fall  $n = |\mathbf{A}| = 4$  beschreibt die Matrix  $\mathbf{A}$  acht Gleichungen, sodass die Homographie-Matrix  $\mathbf{H}$  vollständig bestimmt

Abbildung 3.3: Beispiele für *AprilTags* nach Olson [Ols11]

werden kann. Für das Nullraumproblem  $\mathbf{A}\mathbf{b} = 0$  muss eine Lösung gefunden werden, die nicht-trivial (kein Nullvektor) ist.

Ist das Gleichungssystem überbestimmt ( $n > 4$ ), kann eine Lösung für die Homographie im Rahmen eines Optimierungsproblems approximiert werden. Diese erfolgt beispielsweise mit einer Singulärwertzerlegung.

### 3.4 KÜNSTLICHE MARKER

Künstliche Marker sind Objekte, deren Beschaffenheit so gewählt wird, dass sie möglichst robust und recheneffizient erkannt werden können. Künstliche Marker weisen meist zusätzlich Eigenschaften auf, die eine eindeutige Zuordnung vereinfachen, d. h. eine eindeutige Identifikation wird durch den Markeraufbau (z. B. Kodierung von Identifikationsmerkmalen) ermöglicht. Im Allgemeinen werden künstliche Marker nicht nur bei optischen Ansätzen, sondern auch bei anderen bildgebenden Verfahren wie beispielsweise in der Medizin verwendet.

Eine Vielzahl von Veröffentlichungen diskutieren Ansätze und Algorithmen, um solche Marker für Kamerasysteme zu definieren. Dabei kommen oft binär eingefärbte, flächige Marker zum Einsatz.

Kato und Billinghamurst [KB99], Schmalstieg [SFH<sup>+</sup>02] und Fiala [Fia05] haben grundlegende Arbeiten zum Thema künstliche Marker veröffentlicht. Sie schlagen jeweils Definitionen von Markern und einhergehende Detektionsalgorithmen vor.

Die von Olsen [Ols11] eingeführten *AprilTags* kodieren eindeutige Bezeichner in einem binären flächigen Muster. Sie finden in einer Vielzahl von Verfahren Anwendung und können je nach Anwendungsfall kodiert werden. Für Anwendungen, die eine große Anzahl unterscheidbarer Marker erfordern, sind feiner aufgelöste Marker verfügbar. Anwendungen, die eine robuste Erkennung auch auf große Distanzen benötigen, greifen auf gröber aufgelöste Marker zurück. Die

Kodierung der Marker ist dabei redundant, sodass auch bei Teilverdeckung eine korrekte Detektion möglich ist. Olson nutzt die Hamming-Distanz [Ham50], um die Granularität seiner Marker zu beschreiben. Ein Beispiel für die von Olson beschriebenen Marker ist in Abbildung 3.3 zu sehen.

Eine freie Implementierung der *AprilTags* wird von Kaess [K09] angeboten. Sie wird in der vorliegenden Arbeit für die Markererkennung eingesetzt.



#### 4.1 EINLEITUNG

Die Navigation von LKW-Gespansen stellt eine wesentlich komplexere Aufgabe dar als die Steuerung von Einzelfahrzeugen. Die einzelnen Fahrzeugglieder sind über gelenkartige Verbindungen zusammengekoppelt und daher zumindest an diesen Punkten verwindbar. Je nach Anhänger-Konfiguration kommen weitere Möglichkeiten zur Verwindung hinzu, wie beispielsweise bei einem Zweiachs- bzw. Drehschemel-Anhänger.

Da Fahrzeuggespanne folglich über Freiheitsgrade in ihrer geometrischen Konfiguration verfügen (die über die Lenkachse hinausgehen), gehören sie zu den *artikulierten Fahrzeugen* und werden auch als solche bezeichnet.

Das Wissen über die Konfiguration eines artikulierten Fahrzeugs ist eine Grundvoraussetzung für die sichere und zuverlässige Kontrolle über ein Gespann. Beispielsweise ist ein (teil-)autonomes Fahren nur dann sinnvoll möglich, wenn das Verhalten des Gespanns beschrieben und einbezogen wird. Ein System für die Ermittlung und Überwachung der Gespannkonfiguration bzw. des Gespannstatus ist folglich eine zwingende Notwendigkeit, um Computer-gestütztes Navigieren von artikulierten Fahrzeugen zu ermöglichen.

#### 4.2 STAND DER TECHNIK

Im Bereich der Statuserfassung von artikulierten Fahrzeugen gibt es bisher nur wenige Vorarbeiten, die sich lediglich mit Einachsanhängern befassen.

Caup et al. [CSMH13] nutzen eine rückwärtsgerichtete Weitwinkelkamera an einem PKW um die Deichsel eines angekoppelten Einachsanhängers zu detektieren. Die Detektion wird verwendet, um den Knickwinkel bzw. den Freiheitsgrad in der Artikulation des Anhängers zu berechnen. Der Knickwinkel wird dabei als eindimensionale Größe angenommen (Gierwinkel). Die dabei erreichte Genauigkeit beträgt durchschnittlich ca.  $\pm 2^\circ$ .

De Saxe und Cebon [dSC19] nutzen einen Template-Matching-Ansatz, um die Artikulation in einem Gespann mit einem Sattelanhänger zu messen. Bilder des durch eine rückwärtsgerichtete Kamera am Zugfahrzeug aufgenommenen Anhängers werden mit Bildern verglichen, die anhand eines Modells des Anhängers berechnet werden. Die Referenzbilder simulieren die mögliche Anhängerartikulati-

on und werden als Muster für die Zuordnung des aktuellen Kamerabildes verwendet. Auch dieses Verfahren beschränkt sich auf Einachsanhänger. Die Arbeit von De Saxe und Cebon [dSC19] wurde nach Abschluss der hier vorgestellten Ergebnisse veröffentlicht und zitiert die im Rahmen der Arbeiten an dieser Dissertation veröffentlichten Publikationen.

Ein Verfahren zur Schätzung eines einzelnen Artikulationswinkels wird von Chu et al. [CFS<sup>+</sup>10] vorgeschlagen. Die Autoren nutzen Fahrdynamikparameter, um auf den Artikulationsstatus des Gespanns zu schließen, ohne ihn direkt zu messen. Dabei werden auch Umgebungsparameter, wie die Reibung des Untergrunds, modelliert und berücksichtigt.

Einige Patentschriften adressieren die Thematik der Gespannstatusschätzung, enthalten naturgemäß jedoch nur wenige Informationen zur Umsetzung der Verfahren bzw. nur eine vage Beschreibung der eingesetzten Sensoren und Algorithmen. Die im Folgenden dargestellten Patente nennen Aspekte hinsichtlich des technischen Aufbaus:

Ein von Buckley patentiertes Verfahren [Buc10] nutzt rückwärtige Abstandssensoren, um darin wiederkehrende Signaturen eines Einachsanhängers zu erkennen. Die gewonnenen Informationen werden mit einer kinematischen Modellierung kombiniert, um damit ein Warnsystem für den Fahrer zu speisen. Die Patentschrift enthält keine genaueren Details über die Vorgehensweise, lässt jedoch den Einsatz von Ultraschallsensoren vermuten.

In einem Patent von Tenckhoff [Ten16] werden Magnetfeldsensoren an einer Kugelkopfanhängerkupplung eingesetzt, um deren Verdrehung zu messen. Dabei ist die Sensorik direkt im Kugelkopf verbaut. Burkhart und Rösch [BR16] nutzen am Anhänger montierte Magnetfeldsensoren, um den Winkel innerhalb des Gespanns zu ermitteln. Sensoren am Zugfahrzeug schätzen die Artikulation anhand der gemessenen Feldstärken. Im Beschreibungstext werden auch Zweiaachsanhänger erwähnt. Die genaue Zerlegung der Stärkemessung hinsichtlich einer kinematischen Modellierung mit mehreren Winkeln wird in der Patentschrift jedoch nicht genau thematisiert.

Lu et al. [LGJH12] nutzen eine rückwärtsgerichtete Kamera an einem PKW, um einen angehängten Einachsanhänger zu verfolgen. Dabei setzen sie künstliche Marker ein, um die horizontale Auslenkung im Kamerabild mit einem Histogrammverfahren zu bestimmen. Zusätzlich werden Farbmarkierungen an der Deichsel angebracht, um als Detektionspunkte zu dienen. Dabei befassen sie sich auch mit der automatischen Detektion des Kupplungspunktes, der über das Fahrzeug nach hinten hinausstehen muss. Das Verfahren nutzt die Annahme des ebenen Bodens, da lediglich die horizontale Auslenkung des Markers berücksichtigt wird.

Xu et al. [XTE<sup>+</sup>14] nutzen in ihrer Patentschrift zwei unabhängige

Gierstatusschätzungen auf Zugfahrzeug und Anhänger, aus denen ein Knickwinkel in einer Einachsanhänger-Konfiguration geschätzt wird. Auch hier wird das genauere Vorgehen nicht erläutert.

In einem Patent von Gaedke et al. [GJK<sup>+</sup>10] wird ein kamerabasiertes System beschrieben, das als Grundlage für eine Rangierhilfe für Gespanne mit Einachsanhängern dienen soll. Im Patent wird die Beobachtung eines flachen Anhängers von schräg oben in Kombination mit Erkennungsmechanismen für die *hinteren* Anhängerecken beschrieben, auf deren Grundlage Fahrempfehlungen generiert werden sollen. Weitere Details sind im Patent nicht beschrieben. Im Kontext der Nutzfahrzeuge scheint dieses Verfahren aufgrund der anderen Geometrieverhältnisse und der damit verbundenen anderen Sichtmöglichkeiten der Kamera nicht übertragbar zu sein.

Lee et al. [LK11] haben ein Verfahren patentiert, in dem mehrere Ultraschall- und Radarsensoren für die Artikulationsschätzung in PKW-Einachsanhänger-Gespansen verwendet werden. Die Sensoren sind am Heck des Fahrzeugs verteilt und messen den Abstand zum Anhänger. Die unterschiedlichen Tiefenwerte werden dann für die Schätzung der Auslenkung verwendet.

Verschiedene Fahrzeughersteller aus dem PKW-Bereich haben kürzlich Fahrerassistenzsysteme für das Rückwärtsfahren mit Anhängern angekündigt bzw. vorgestellt. Diese Assistenzsysteme adressieren lediglich den einfachen Fall des Einachsanhängers und nutzen einen speziellen Drehsensor, der im Kugelkopf der Anhängerkuppelung verbaut ist. Zu diesen Systemen sind keine wissenschaftlichen Publikationen bekannt.

#### 4.2.1 2-D-Verfahren zur Statusschätzung

Auch an der Universität Koblenz-Landau wurden bereits Arbeiten im Bereich der Statusschätzung für artikulierte Fahrzeuge geleistet. Neben Arbeiten für Gespanne mit Einachsanhängern (u. a. in einer Publikation von Schikora, Berg und Zöbel [SBZ09] prototypisch umgesetzt), wurden bereits erweiterte Verfahren für komplexere Gespannkinematiken entwickelt. Im Wesentlichen sind diese in einem Patent von Balcerak, Zöbel und Weidenfeller [BZW06] veröffentlicht.

Das dort vorgestellte Verfahren adressiert Zweiachsanhänger und nutzt eine Kamera, um ein künstliches Muster zu beobachten, das am Anhänger angebracht ist. Dabei wird die Einschränkung getroffen, dass sich das Fahrzeug auf ebenem Grund bewegt, sodass eine 2-D-Annahme für die Fahrzeuggeometrie getroffen werden kann. Die Artikulation des Gespanns wird dabei auf zwei Gierwinkel reduziert.

Das Muster hat die Form eines gleichschenkligen Dreiecks, das an den Eckpunkten eine Kennzeichnung (in einer Anwendung beispielsweise ein weißer Kreis auf schwarzem Grund, siehe Abbildung 4.1) enthält, die mit Hilfe der Kamerabilder erkannt wird. In Kombinati-

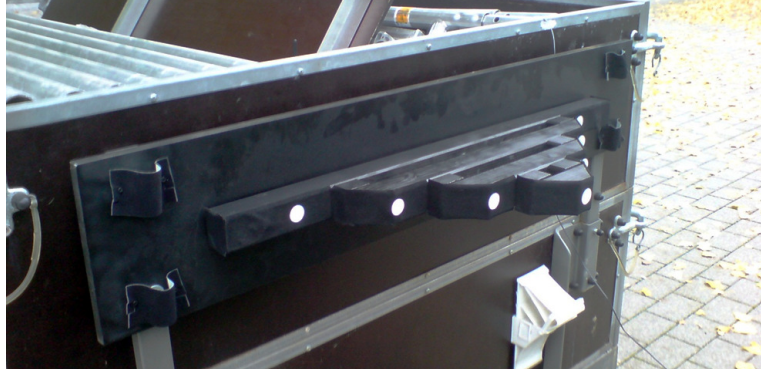


Abbildung 4.1: Verwendetes künstliches Muster für den Testaufbau  
(Bildquelle: Forschungsgruppe Echtzeitsysteme,  
Universität Koblenz-Landau)

on mit dem Öffnungswinkel der Kamera wird dabei der horizontale Blickwinkel interpoliert, um mit Hilfe von Winkelrelationen und trigonometrischen Funktionen die Knickwinkel zu errechnen.

Der gewählte Aufbau ist in Abbildung 4.2 zu sehen und stellt die relevanten geometrischen Eigenschaften schematisch dar. An den beiden Kupplungs- bzw. Artikulationspunkten  $k_1$  und  $k_2$  beschreibt jeweils ein Gierwinkel ( $\xi_1$  und  $\xi_2$ ) die Artikulation des Gespanns. Die Kamera auf dem Fahrzeug beobachtet die (mindestens) drei Punkte des Musters am Anhänger ( $M_1$ ,  $M_2$  und  $M_3$ ). Die Dimensionen des gleichschenkligen Dreiecks werden mit den Parametern  $w_\Delta, h_\Delta \in \mathbb{R}$  beschrieben. Aus den beobachteten Bildkoordinaten werden zusammen mit den geometrischen Relationen des Gespanns ( $l_1, l_2, l_3 \in \mathbb{R}$ ) und der Kamerageometrie die Winkel  $\xi_1$  und  $\xi_2$  berechnet:

Seien  $q_i = \mathcal{P}(M_i)$  mit  $q_i \in \mathbb{R}^2$  und  $i \in \{1, 2, 3\}$  die im Bild der Kamera detektierten Abbildungen der Markerpunkte  $M_i$ . Die relativ zur Blickrichtung der Kamera resultierenden Blickwinkel  $\alpha_i$  sind mit Hilfe der Kamera-Kalibrierdaten bestimmbar.

Die folgenden Formeln beschreiben die geometrischen Zusammenhänge für den Fall  $|\alpha_1 - \alpha_2| < |\alpha_2 - \alpha_3|$  [BZW06]. Die zyklische Definition von Winkeln im Allgemeinen stellt an dieser Stelle keine Einschränkung dar, da die Werte der Winkel  $\alpha_i$  durch den Kameraöffnungswinkel begrenzt sind und kein zyklischer Übersprung des Wertebereichs zu erwarten ist.

Es werden Winkelbeziehungen, trigonometrische Funktionen und Sinussätze genutzt, um die Gierwinkel  $\xi_1$  und  $\xi_2$  bei bekannten Parametern zu errechnen. Für die Fälle mit anderen Verhältnissen zwischen  $\alpha_1, \alpha_2$  und  $\alpha_3$  werden die Beziehungen analog mit gespiegelten Beziehungen durchgeführt [BZW06]. Seien im Folgenden  $\beta_i \in \mathbb{R}$  Hilfsvariablen für Winkel und  $a_i \in \mathbb{R}^2$  Hilfsvektoren (mit  $i \in \mathbb{N}$ ).

Beschreibe  $d_\Delta$  die Schenkellänge des gleichschenkligen Dreiecks, das durch die Punkte  $M_1$ ,  $M_2$  und  $M_3$  aufgespannt wird. Der Win-



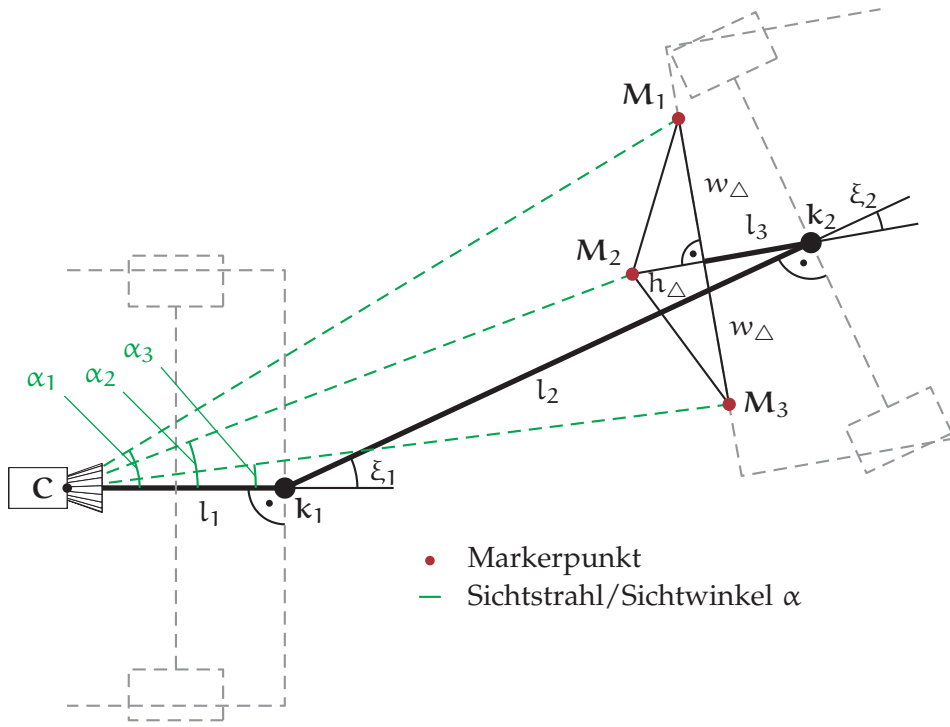


Abbildung 4.2: Mustergeometrie und Winkelrelationen im Aufbau nach [BZW06] (Draufsicht)

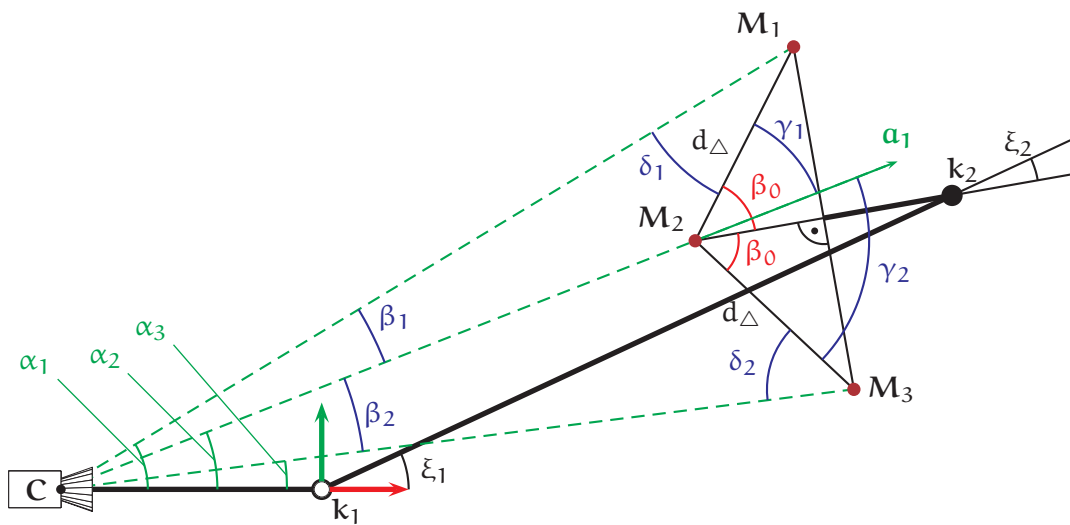


Abbildung 4.3: Winkelrelationen, Hilfspunkte und Koordinatensystem für den Aufbau nach [BZW06]

kel  $\beta_0$  beschreibt den halben Innenwinkel des gleichschenkligen Dreiecks.

$$d_{\Delta} = \sqrt{w_{\Delta}^2 + h_{\Delta}^2} \quad \beta_0 = \text{atan}(w_{\Delta} \cdot h_{\Delta}^{-1}) \quad \mathbf{C} = (l_1, 0)^T$$

Daraus lassen sich die Winkel  $\beta_1$  und  $\beta_2$  zwischen den Sichtstrahlen bestimmen:

$$\beta_1 = \alpha_1 - \alpha_2 \quad \beta_2 = \alpha_2 - \alpha_3 \quad .$$

Für die Winkel  $\gamma_1$  und  $\gamma_2$  zwischen den gleichschenkligen Seiten der Markerpyramide und dem verlängerten Sichtstrahl durch  $\mathbf{M}_2$  (siehe Abbildung 4.3) gilt:

$$2 \cdot \beta_0 = \gamma_1 + \gamma_2 \quad .$$

Aus den Winkelrelationen ergibt sich für  $\delta_1$  und  $\delta_2$ :

$$\delta_1 = \gamma_1 - \beta_1 \quad \delta_2 = \gamma_2 - \beta_2 \quad \Rightarrow \quad \delta = \delta_1 + \delta_2 \quad .$$

Zusammengefasst ergibt sich:

$$\begin{aligned} \delta &= 2 \cdot \beta_0 - \beta_1 - \beta_2 \\ \delta_1 &= \text{atan} \left( \frac{\sin \beta_1 \cdot \sin \delta}{\sin \beta_2 + \sin 4\beta_1 \cdot \cos \delta} \right) \quad . \end{aligned}$$

Daraus kann die Distanz zwischen  $\mathbf{C}$  und  $\mathbf{M}_2$  und die Lage von  $\mathbf{M}_2$  berechnet werden:

$$\begin{aligned} \|\mathbf{M}_2 - \mathbf{C}\| &= d_\Delta \cdot \frac{\sin \delta_1}{\sin \beta_1} \\ \mathbf{M}_2 &= \mathbf{C} + \|\mathbf{M}_2 - \mathbf{C}\| \cdot (\cos \alpha_2, \sin \alpha_2)^T \quad . \end{aligned}$$

Der Punkt  $\mathbf{a}_1$  beschreibt die Verlängerung des Sichtstrahls durch  $\mathbf{M}_2$  um die Länge  $d_\Delta$ :

$$\mathbf{a}_1 = \mathbf{M}_2 + d_\Delta \cdot (\cos \alpha_2, \sin \alpha_2)^T \quad .$$

Die Punkte  $\mathbf{M}_1$  und  $\mathbf{M}_3$  werden durch Drehung des Punktes  $\mathbf{a}_1$  um den Punkt  $\mathbf{M}_2$  um die Winkel  $\gamma_1$  bzw.  $\gamma_2$  erreicht bzw. durch Drehung des jeweils zuerst berechneten Punktes um  $\pm 2 \cdot \beta_0$ :

$$\begin{aligned} \mathbf{M}_1 &= \text{rotate}(\mathbf{a}_1, \gamma_1, \mathbf{M}_2) = \text{rotate}(\mathbf{a}_1, \delta_1 + \beta_1, \mathbf{M}_2) \\ \mathbf{M}_3 &= \text{rotate}(\mathbf{M}_1, -2 \cdot \beta_0, \mathbf{M}_2) \quad . \end{aligned}$$

Der Punkt  $\mathbf{a}_2$  beschreibt die Richtung von  $\mathbf{M}_2$  zum Mittelpunkt der Strecke zwischen  $\mathbf{M}_1$  und  $\mathbf{M}_3$  ( $\|\mathbf{a}_2\| = d_\Delta$ ):

$$\mathbf{a}_2 = \text{rotate}(\mathbf{M}_1, -\beta_0, \mathbf{M}_2) \quad .$$

Die Lage von  $\mathbf{k}_2$  lässt sich durch die bekannten Größen  $h_\Delta$  und  $l_3$  berechnen:

$$\mathbf{k}_2 = \mathbf{M}_2 + \mathbf{a}_2 \cdot \frac{h + l_3}{d_\Delta} \quad .$$

Die Gierwinkel  $\xi_1$  am Zugfahrzeug und  $\xi_2$  am Anhänger ergeben sich mit:

$$\xi_1 = \text{atan} \left( \frac{(0, 1)^T \circ \mathbf{k}_2}{(1, 0)^T \circ \mathbf{k}_2} \right)$$

$$\xi_2 = \arccos \frac{(\mathbf{M}_2 - \mathbf{k}_2) \circ (-\mathbf{k}_2)}{\|\mathbf{M}_2 - \mathbf{k}_2\| \cdot \|\mathbf{k}_2\|} .$$

Für den Fall  $(-\mathbf{k}_2)^\perp \circ \mathbf{M}_2 < 0$  wird  $\xi_2 = -\xi_2$  gesetzt. Für andere Winkelrelationen hinsichtlich  $\alpha_0, \beta_1$  und  $\beta_2$  werden analoge Vorgehensweisen beschrieben [BZW06].

Genauere Evaluationsdaten wurden für das Verfahren bisher nicht veröffentlicht. Im Patent ist lediglich eine Tabelle mit einer Abschätzung der Genauigkeit enthalten, die nur gewisse händisch gemessene Winkelkonstellationen beinhaltet.

#### 4.2.2 Kinematikmodelle artikulierter Fahrzeuge

Das Messen und die Extrapolation der aktuellen Gespannkonfiguration erfordert die Kenntnis über die geometrischen und kinematischen Grundlagen von artikulierten Fahrzeugen.

Laumond [Lau90] betrachtet rigide Multi-Körper-Systeme und beschreibt deren Bewegung auf einer Ebene unter der Annahme, dass kein Schlupf auftritt. Fahrzeuggespanne werden dabei als Verkettung von Starrkörpern beschrieben, die in der Modellierung nach Laumond immer am Zentrumspunkt einer Achse zusammengekuppelt sind (sogenannte *standard-n-trailer*, wobei  $n$  die Anzahl der Starrkörper beschreibt).

Altafini [Alto1] erweitert dieses Modell um Kupplungspunkte, die nicht zwangsweise auf den Achsen liegen. Dieses Modell erlaubt die Beschreibung von gewöhnlichen Fahrzeuggespannen, die z.B. eine Deichsel für die Verbindung der Fahrzeugglieder nutzen. Altafini definiert in seiner Veröffentlichung den *general-n-trailer*, wobei  $n$  für die Anzahl der starren Elemente im Gespann steht. Ein Zugfahrzeug, das mit einem Zweiachsanhänger verbunden ist, wird beispielsweise entsprechend als *general-3-trailer* bezeichnet (Starre Elemente sind Zugfahrzeug, Deichsel und Anhänger.).

Die mit der Erweiterung um Kupplungspunkte jenseits der Achsenmitten beschriebenen Freiheitsgrade erlauben vielfältige kinematische Verhaltensweisen innerhalb der *general-n-trailer*-Klasse, die bei der Konfigurationsschätzung für das Gespann berücksichtigt werden müssen.

Die kinematischen Belange eines Fahrzeuggespanns werden bereits in verschiedenen Publikationen diskutiert: Verschiedene Autoren schlagen Lösungen für das rückwärtige Navigieren vor [Alto2, LCKSo4, KK92, ASW01, ASJo2]. Andere Publikationen spezialisieren sich auf passiv gesteuerte Anhänger [Mic12, CPY<sup>+</sup>11, DW97] und aktiv gesteuerte Anhänger mit lenkbaren Achsen [OGABV02, NETCo1].

Wie gezeigt, beschäftigen sich bereits einige Publikationen mit der Fragestellung der Steuerung und Regelung von artikulierten Fahrzeugen. Interessanterweise zeigt keine dieser Veröffentlichungen eine

Lösung für die Konfigurationsschätzung auf, die eine unumgängliche Voraussetzung für das korrekte Berechnen und Überwachen der Gespann-Kinematik ist. Sie nutzen angenommene Ursprungszustände, die nicht durch entsprechende Einrichtungen verifiziert werden. Die Schätzung des Gespannzustands ist jedoch Grundlage für die Entwicklung von Fahrerassistenzsystemen, die zur Steigerung des Komforts und der Operationssicherheit von Fahrzeuggespannen beitragen können.

#### 4.2.3 Trajektorienbeurteilung

Der von Fahrzeugen zurückgelegte Weg wird als Trajektorie bezeichnet. Um Trajektorien bewerten bzw. den Unterschied zwischen Trajektorien quantifizieren zu können, sind spezielle Fehlermaße notwendig. Dies ist beispielsweise für den Vergleich einer optimal geplanten und einer tatsächlich gefahrenen Trajektorie sinnvoll.

Der Abstand zwischen Trajektorien ist mit gewöhnlichen euklidischen Abstandsmaßen nicht sinnvoll beschreibbar, da die Abweichung komplexer ist. Die *Fréchet*-Distanz [AG95] ist ein Ansatz, der für die Berechnung der Distanz zwischen parametrisierten Kurven genutzt wird. Dabei werden sowohl die Reihenfolge als auch die Positionen innerhalb der Trajektorie für die Berechnung verwendet. Die Adaption der *Fréchet*-Distanz auf diskrete Datensätze wird von Eiter und Mannila [EM94] diskutiert. Die Ergebnisse zeigen jedoch, dass die *Fréchet*-Distanz anfällig für Ausreißer in den Daten ist.

Die in Kapitel 2.4.2 vorgestellte *Earth Mover's Distance (EMD)* [Mon81, RTG00] kann adaptiert werden, um den Abstand zwischen Trajektorien zu beschreiben. Dabei werden Trajektorien als Kurven interpretiert. Eine diskrete Trajektorie kann dabei als diskrete Wahrscheinlichkeitsverteilung konditioniert werden. Boem et al. [BFP11] beschreiben ein Verfahren, das dynamische Rasterkarten für die Berechnung der Diskretisierung verwendet. Die Berechnung der *EMD* basiert dann auf der Lösung des Transportproblems nach Hitchcock [Hit41] bzw. Kantorovich [Kan42].

Die *EMD* wird als Abstandsmaß zwischen zwei Trajektorien verwendet, um die Qualität von Trajektorien zu beschreiben.

### 4.3 MODELLBILDUNG

Für die Statusbeschreibung eines artikulierten Gespanns ist ein Modell notwendig, das die möglichen Freiheitsgrade beschreibt und in einen Kontext setzt. Die hier aufgeführten Betrachtungen beziehen sich auf Zweiachsanhänger mit Drehschemel, die in der Kategorisierung nach Altafini [Alto1] als *general-3-trailer* bezeichnet werden. Da Zugfahrzeug und Anhänger voneinander getrennt werden können, werden die einzelnen Fahrzeugglieder zunächst isoliert betrachtet.

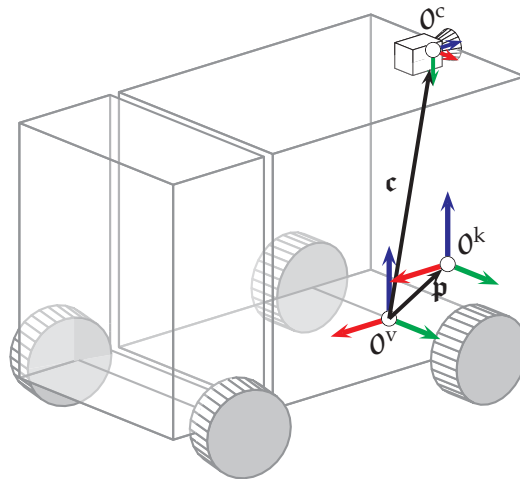


Abbildung 4.4: Koordinatensysteme auf dem Zugfahrzeug

#### 4.3.1 Zugfahrzeug

Das Zugfahrzeug, das als gewöhnlicher Lastkraftwagen angenommen wird, verfügt über eine Steuerung, die nach dem Ackermann-Prinzip bzw. der Achsschenkellenkung funktioniert und über eine lenkbare Vorderachse verfügt. Das Fahrzeug verfügt zusätzlich über eine oder mehrere starre Hinterachsen. Innerhalb des Fahrzeugs wird ein kartesisches Koordinatensystem  $v$  definiert, dessen Ursprung und Orientierung auf den Mittelpunkt der Fahrzeughinterachse gelegt wird.

Die Bodenebene liegt dabei parallel zur  $xy$ -Ebene des kartesischen Koordinatensystems, die  $x$ -Achse verläuft entlang in Richtung der vorwärtigen Fahrtrichtung. Die  $z$ -Achse zeigt nach oben in Richtung des Himmels. Die  $y$ -Achse beschreibt entsprechend die Richtung nach links im Hinblick auf die Fahrtrichtung.

Auf dem Zugfahrzeug wird eine Kamera installiert, die den Fahrzeuganhänger beobachtet. Die Kamera spannt das Koordinatensystem  $c$  auf, das als mit dem Koordinatensystem  $v$  des Fahrzeugs rigide verbunden angenommen wird. Für die Transformation von  $v$  nach  $c$  wird die Pose  $\mathbf{c} \in \mathbb{R}^3 \times SO(3)$  definiert. Die Zusammenhänge sind in Abbildung 4.4 zu sehen.

Das Zugfahrzeug verfügt über eine Anhängerkupplung, die eine normierte Schnittstelle zur Befestigung von Anhängern bereitstellt. Bei Lastkraftwagen sind insbesondere Maulkupplungen (siehe Abbildung 4.5) weit verbreitet. Die Lage der Anhängerkupplung relativ zum Ursprung des Fahrzeugkoordinatensystems  $v$  wird durch die Pose  $\mathbf{p} \in \mathbb{R}^3 \times SO(3)$  beschrieben.



Abbildung 4.5: Maulkupplung an einem Zugfahrzeug

#### 4.3.2 Anhänger

Die in diesem Abschnitt betrachteten Anhänger sind Zweiachsanhänger, die über eine Drehschemel-Lenkung verfügen (*general-3-trailer*).

Anhänger verfügen über eine Deichsel, mit der sie an das Zugfahrzeug gekoppelt werden. Die Deichsel ist am Zweiachsanhänger an einem Drehschemel befestigt, auf dem der eigentliche Anhänger aufsitzt. Im hinteren Bereich verfügt der Anhänger zusätzlich über mindestens eine starre Achse. Analog zum Zugfahrzeug wird der Ursprung des Anhängerkoordinatensystems auf den Mittelpunkt der Hinterachse gelegt. Der Drehschemel und damit auch sein Drehzentrum haben eine feste Position am Anhänger. Die Transformation vom Anhängerkoordinatensystem zum Drehpunkt wird mit der Pose  $\mathbf{t} \in \mathbb{R}^3 \times SO(3)$  beschrieben. Bei konstanter Geometrie kann diese Pose als konstant angenommen werden. Das Koordinatensystem, das seinen Ursprung im Drehschemel hat und *nicht* durch die dort mögliche Artikulation beeinflusst wird, wird mit dem Buchstaben  $\mathbf{d}$  bezeichnet.

Die Artikulation des Drehschemels wird als eigenständige Pose  $\mathbf{g} \in \mathbb{R}^3 \times SO(3)$  betrachtet, die die geometrischen Abhängigkeiten innerhalb des Anhängers nicht berücksichtigt. Da die Artikulation lediglich als Rotation zu erwarten ist, wird die Translationskomponente trivial auf den Nullvektor gesetzt.

Für den Fall, dass die Deichsel geradegerichtet ist bzw. sich der Anhänger in Vorfahrt auf ebenem Grund befindet, gilt  $\mathbf{g} = \mathbf{o}$ . Der Kupplungspunkt wird vom Drehschemel aus durch Transformation mit der Pose  $\mathbf{d}$  erreicht. Für die hier vorgenommene Modellierung wird diese Pose als konstant angenommen. Manche Anhänger verfügen je nach Ausführung über eine Deichsel, die sich marginal in ihrer Länge verändern kann, z. B. durch eine Auflaufeinrichtung, die jedoch nur in kleinen bzw. leichten und mittelgroßen Anhängern An-

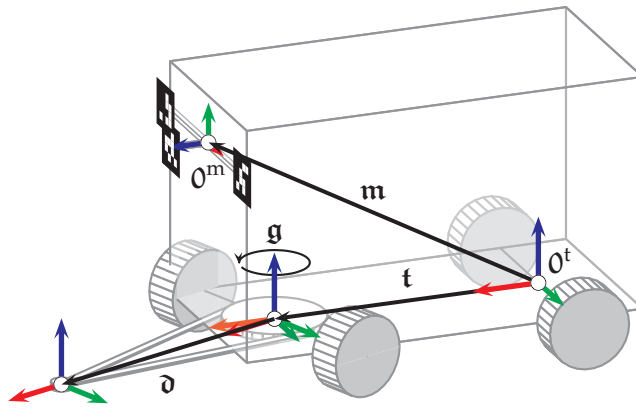


Abbildung 4.6: Koordinatensysteme im Zweiachsanhänger

wendung findet. Diese mögliche Dynamik wird in der vorgenommenen Modellierung aufgrund des maximal möglichen geringen Einflusses bewusst ausgeblendet. Für die Transformation vom Ursprung des Anhängerkoordinatensystems zum Kupplungsauge ergibt sich:

$$\mathbf{t} \oplus \mathbf{g} \oplus \mathbf{d} .$$

Die einzelnen Koordinatensysteme und deren Zusammenhänge sind in Abbildung 4.6 ersichtlich.

Für die im Folgenden vorgestellten Verfahren wird ein System aus passiven, künstlichen Markern verwendet, das an der Vorderseite des Anhängers montiert wird, sodass es durch eine Kamera an der Rückseite des Zugfahrzeugs erfasst werden kann. Die Pose des Systems relativ zum Koordinatensystem des Anhängers  $t$  wird durch die Pose  $\mathbf{m} \in \mathbb{R}^3 \times \text{SO}(3)$  beschrieben. Für das Muster wird das Koordinatensystem  $m$  aufgespannt. Innerhalb dieses Koordinatensystems werden die Posen der einzelnen Marker beschrieben. Hierbei sind unterschiedlichste Anordnungen denkbar. Um eine Vergleichbarkeit mit dem Verfahren nach Balcerak, Zöbel und Weidenfeller [BZW06] zu ermöglichen, wird die dort spezifizierte und für den Algorithmus notwendige Pyramidenform adaptiert. Für die vorgestellten neu entwickelten Verfahren (siehe Kapitel 4.5 und 4.6) ist diese Form nicht zwingend erforderlich.

### 4.3.3 Fahrzeuggespann

Die in den Kapiteln 4.3.1 und 4.3.2 entwickelten Modelle können nun zusammengesetzt werden, um ein Fahrzeuggespann zu erhalten. Dazu wird das Kupplungsauge der Anhängerdeichsel mit der Anhängerkupplung des Zugfahrzeugs verbunden. Dadurch wird die Position des Kupplungsauges am Anhänger auf die Position der Anhängerkupplung verschoben. Bei gerader Vorfahrt auf ebenem Grund sind die beiden Posen für das Kupplungsauge und die Anhänger-

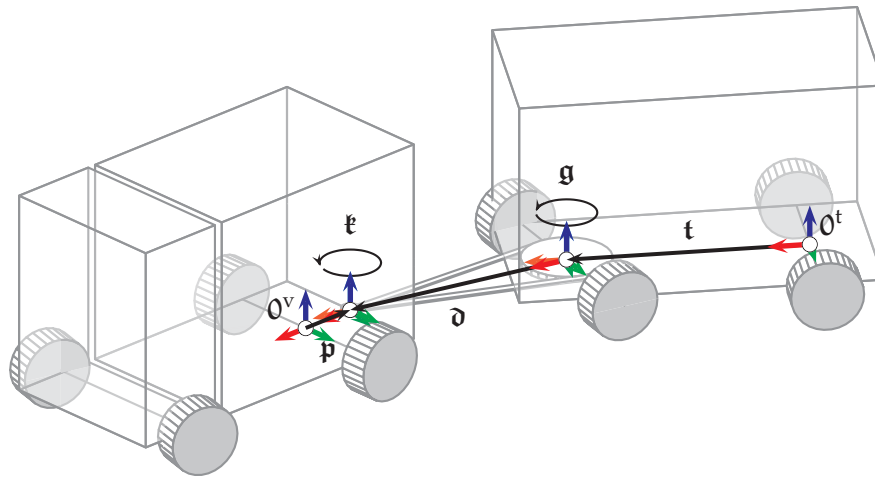


Abbildung 4.7: Koordinatensysteme im zusammengekuppelten Fahrzeuggespann

kupplung (im Weltkoordinatensystem) äquivalent.

Die Artikulation am Kupplungspunkt wird als Rotation modelliert, da das an der Kupplung auftretende Spiel im Hinblick auf die Position als vernachlässigbar klein angesehen wird. Die Pose  $\mathbf{k} \in \mathbb{R}^3 \times SO(3)$  beschreibt die Artikulation, wobei die Translationskomponente als Nullvektor gesetzt wird. Für den Fall der geraden Vorfahrt auf ebenem Grund ist  $\mathbf{k} = \mathbf{o}$ .

Für das zusammengekuppelte Gespann kann die Transformation vom Ursprung des Zugfahrzeugkoordinatensystems hin zum Ursprung des Anhängerkoordinatensystems durch Verkettung der definierten Posen beschrieben werden (Die Artikulationsposen sind farbig hervorgehoben):

$$\mathbf{p} \oplus \mathbf{k} \ominus (\mathbf{t} \oplus \mathbf{g} \oplus \mathbf{d}) \quad .$$

Der Zusammenhang ist in Abbildung 4.7 dargestellt. Der Gespannzustand wird folglich durch die beiden Posen  $\mathbf{k}$  und  $\mathbf{g}$  beschrieben, die die Artikulationen der jeweiligen Gelenke enthalten. Die weiteren für die Transformationen relevanten Posen können durch präzises Messen bzw. geeignete Kalibrierverfahren bestimmt werden.

#### 4.4 2-D-ANSATZ

Das von Balcerak, Zöbel und Weidenfeller [BZW06] (siehe Kapitel 4.2.1) beschriebene Verfahren dient als Grundlage für die Implementierung der 2-D-Statusschätzung. In Bezug auf die Modellierung des Fahrzeuggespanns (siehe Kapitel 4.3.3) werden die Artikulationsposen  $\mathbf{k}$  und  $\mathbf{g}$  aufgrund der für das Verfahren geltenden Voraussetzungen eingeschränkt. Durch die planare Annahme werden die Rotationen in den Artikulationsposen auf jeweils einen Winkel reduziert.



Pose  $\mathfrak{k}$  wird durch den Winkel  $\xi_1$  und Pose  $\mathfrak{g}$  wird durch den Winkel  $\xi_2$  repräsentiert.

### *Software-Komponenten*

Die Schaffung eines Systems für die Schätzung des Gespannzustands stellt Sicherheitsanforderungen für den Betrieb. Bereits in der Software-Architektur müssen daher Vorkehrungen getroffen werden, welche die Beherrschbarkeit einzelner Bausteine gewährleisten und potentiellen Fehlersituationen vorbeugen. Eine adäquate Architektur für das Vorhaben wurde entwickelt und auf einer Konferenz veröffentlicht und vorgestellt [1]. Die für diese Arbeit wichtigen Aspekte werden im Folgenden dargestellt.

Für die Statusschätzung des Gespanns sind mehrere Bausteine notwendig, die das Software-System für den Sensor definieren:

1. Bildaufnahme
2. Bildverarbeitung
3. Winkelschätzung

Der Baustein „Bildaufnahme“ beschäftigt sich mit der Ansteuerung der Kamera und der Bereitstellung der Bilddaten für die weitere Verarbeitung. Die für die Aufnahme benötigte Zeit muss bei der weiteren Verarbeitung betrachtet werden, um für die zeitliche Aktualität der Daten zu sorgen. Nach der Aufnahme der Bilder müssen diese weiterverarbeitet werden. Im zweiten Baustein „Bildverarbeitung“ werden zunächst Bildvorverarbeitungsschritte (z. B. Entzerrung, Normalisierung etc.) durchgeführt bevor die Extraktion der Markerpunkte des eingesetzten Musters erfolgt. Im dritten Baustein „Winkelschätzung“ werden die aus dem Bild gewonnenen Informationen genutzt, um eine Rückrechnung auf die benötigte Statusinformation (bei der 2-D-Variante die Gierwinkel  $\xi_1$  und  $\xi_2$ ) durchzuführen. Die Bezeichnung „Winkelschätzung“ meint das konkrete Ermitteln der Artikulation anhand der aus den Kamerabildern extrahierten Informationen. Diese Bezeichnung ist gewählt, um eine Abgrenzung zum Begriff „Statusschätzung“ zu finden, der für das Gesamtverfahren verwendet wird.

Die einzelnen Bausteine können prinzipiell hintereinandergeschaltet werden. Es ist jedoch sinnvoll, die Daten an den entsprechenden Übergabestellen auf Validität und Aktualität zu prüfen. In Abbildung 4.8 ist ein Schaubild zur beschriebenen Software-Architektur zu sehen. Die einzelnen Bausteine (in der Abbildung als *Workers* bezeichnet), werden mit einer Kontrollschicht unter Verwendung des *Facade-Patterns* nach [GHJV94] gekapselt. Die Kontrollschicht (*Controllers & Guards*) ermöglicht die Integration von Validierungsmechanismen, die die ein- und ausgehenden Daten auf Validität prüfen. Die

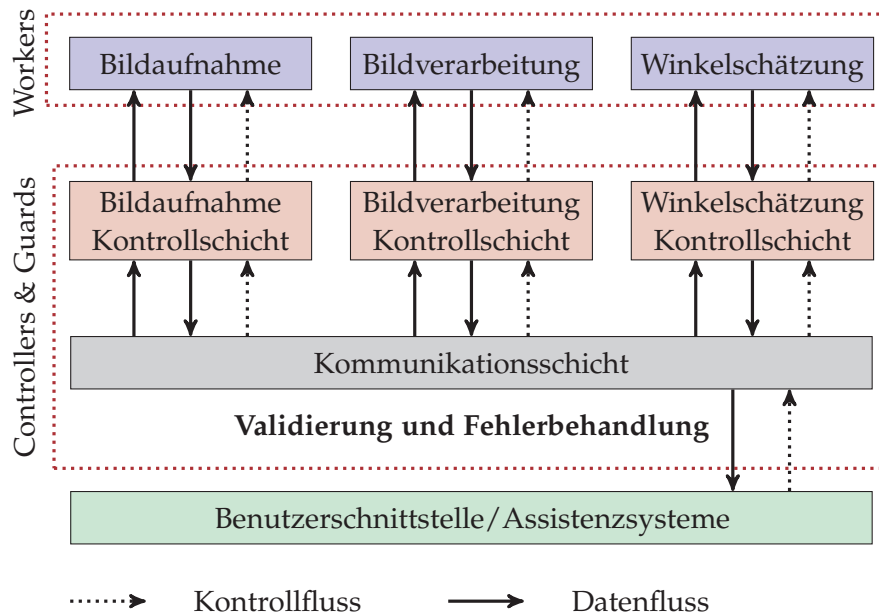


Abbildung 4.8: Software-Architektur für das Messsystem

Software-Architektur adressiert die Belange der funktionalen Sicherheit sowie die Adaptier- und Kontrollierbarkeit der einzelnen Komponenten. Für die Anbindung der einzelnen Bausteine wird eine *Observer*-Struktur nach [GHJV94] angewandt, um eine flexible Integration zu gewährleisten. Diese Struktur ermöglicht zudem eine Schnittstelle für die Auswertung der einzelnen Schritte.

#### 4.4.1 Virtuelle Testumgebung

Die Veröffentlichung von Balcerak, Zöbel und Weidenfeller [BZW06] enthält lediglich einzelne Referenzwerte, die mittels eines einfachen Modellaufbaus händisch ermittelt wurden. Eine Evaluation über den zu erwartenden Messbereich und die zu erwartende Präzision ist noch nicht verfügbar.

Im Rahmen der Weiterentwicklung ist es unabdingbar, eine Aussage über die vom Sensorsystem zu erwartende Genauigkeit treffen zu können. Ohne entsprechendes Referenzsystem ist eine qualitative Evaluation nicht möglich. Insbesondere für den *general-3-trailer* ist kein Messsystem (allgemein) verfügbar, das beide Gierwinkel erfasst. Die im Folgenden dargestellten Ergebnisse wurden bereits auf einer internationalen Konferenz veröffentlicht [2].

Der für die Evaluation gewählte Ansatz beruht auf einer Simulationsumgebung, die die Geometrie des Gespanns abbildet und ein durch den jeweiligen Konfigurationsvektor definiertes synthetisches Bild erzeugt. Durch die Simulation wird eine Evaluation möglich, die rein aus Software-Komponenten besteht. Der Versuchsaufbau ist

folglich ein *Software-in-the-Loop*-System, das über frei konfigurierbare Parameter in Hinblick auf Geometrie und virtueller Kamera verfügt. Klassische Evaluationsmethoden mit Testaufbauten auf Fahrzeugge spannen implizieren die Notwendigkeit für eine Vielzahl von unterschiedlichen Gespannkonfigurationen, was den für die Evaluation notwendigen Aufwand stark erhöhen würde.

Ziel der vorgestellten Evaluation ist eine Aussage hinsichtlich des von der Statusschätzung zu erwartenden Verhaltens. Kernpunkte sind: Statistische Fehler (siehe Kapitel 2.4), Numerische Stabilität und Fehlertoleranz gegenüber Systemparametern. Die Kombination dieser Punkte führt zur Festlegung der Parametergrößen für die virtuelle Testumgebung. Relevante Größen sind:

1. *Fahrzeuggespann*  
Dimension, Deichsellänge, Gelenkpositionen
2. *Kamera*  
Auflösung, Öffnungswinkel, Installationsposition
3. *Markersystem*  
Größe der einzelnen Marker, Dimension der Markerpyramide, Installationsposition
4. *Künstliche Fehler*  
Inkorrekte Positionierung der Komponenten, Sensorrauschen, falsche Kalibrierung

Die Simulation der Kamera stellt einen inversen Prozess im Hinblick auf die Funktionsweise der Statusschätzung dar. Auf der einen Seite wird aus den gesetzten Modellparametern ein Bild der virtuellen Kamera erzeugt. Auf der anderen Seite verarbeitet die Statusschätzung dieses Bild wieder zurück zu den geschätzten Modellparametern. Dieser Zusammenhang ist in Abbildung 4.9 dargestellt.

Die einzelnen Bausteine haben ihren jeweiligen Einfluss auf die Qualität der Ergebnisse. Die virtuelle Testumgebung kann daher in zwei verschiedenen Tiefen eingesetzt werden (siehe Abbildung 4.9). Bei Simulationstiefe 1 werden die Kamerabilder direkt an den Baustein Bildverarbeitung übergeben. Um den Effekt der Bildverarbeitung zu neutralisieren ist es möglich, die perfekten Markerpositionen (Subpixel-genau) im Bild direkt an den Baustein „Winkelschätzung“ zu übergeben (Simulationstiefe 2). Damit lassen sich die Einflüsse in beiden Konfigurationen betrachten. Es kann somit identifiziert werden, wie stark der Einfluss der Bausteine ist. Durch die Parametrierung der virtuellen Kamera können potentielle Einflüsse des Bausteins „Bildaufnahme“ bzw. der Hardware gemessen werden.

Um die Qualität der Statusschätzung zu beurteilen, werden je Testlauf mit gegebenem Konfigurationssatz der jeweilige Eingabestatus mit dem gemessenen Status protokolliert und verglichen.

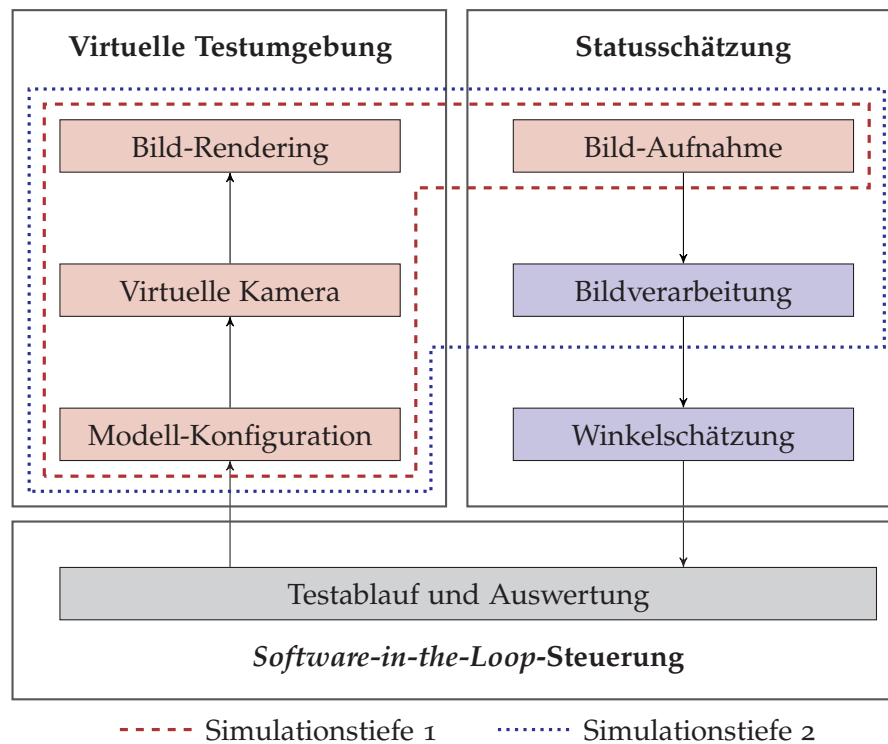


Abbildung 4.9: Die virtuelle Testumgebung als Inverse zur Statusschätzung (nach [2])

#### 4.4.2 Evaluation

Die virtuelle Testumgebung (siehe Kapitel 4.4.1) ist Grundlage für die durchgeführte Evaluation. Beide Simulationsstufen (siehe Abbildung 4.9) werden für verschiedene Kamerakonfigurationen getestet. Erwartetes Verhalten ist ein numerischer Fehler in der Berechnung der Winkel bei der Anwendung des Verfahrens nach [BZW06]. Der Einfluss der Markererkennung unter Verwendung der Implementierung [9] der Markererkennung nach [Ols11] und der Bildverarbeitungs-komponente wird untersucht.

Die Testergebnisse wurden mit unterschiedlichen Kameraparametern ermittelt. Diese Parameter sind wichtig für die Wahl der korrekten Kamerahardware bzw. der Objektive für das Endsystem. Es wird erwartet, dass die Bildbreite Einfluss auf die Qualität der Ergebnisse hat, da die Sichtwinkel entsprechend feiner bzw. gröber quantisiert werden.

Parameter	$l_1$	$l_2$	$l_3$	$w_\Delta$	$h_\Delta$	$w_m$	Parameter	$\alpha$
Wert in cm	18.4	16.4	3.5	6.25	4.3	2.0	Wert in Grad	67

Tabelle 4.1: Modellparameter für die Evaluation des Verfahrens (siehe Abbildung 4.2)

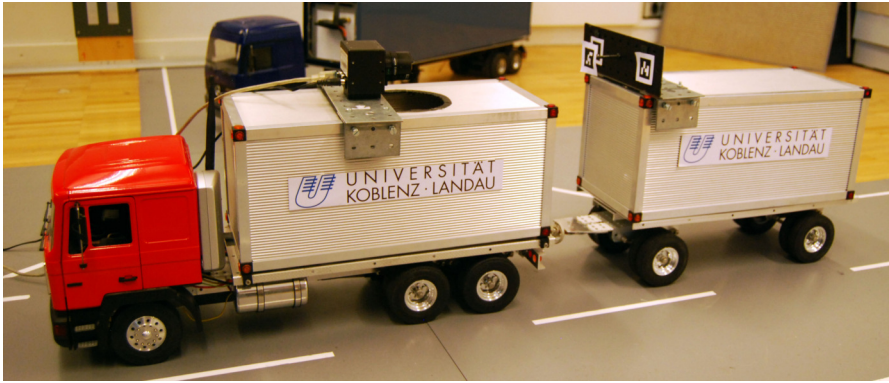


Abbildung 4.10: Laboraufbau im Maßstab 1 : 16

Die Geometrieparameter für die Evaluation entstammen einem maßstabsgetreuen Laboraufbau. Der Laboraufbau ist in Abbildung 4.10 zu sehen. Die für die präsentierte Evaluation verwendeten Parameter sind in Tabelle 4.1 ersichtlich.

Das Modell wird mit Winkeln  $\xi_1, \xi_2 \in [-75^\circ; 75^\circ]$  mit einer Schrittweite von  $1.5^\circ$  abgetastet (10 000 Werte). Das jeweilige synthetische Bild wird als Eingabe für die Statusschätzung verwendet. Der berechnete Status mit den Winkeln  $\xi'_1$  und  $\xi'_2$  wird zusammen mit den Eingabewinkeln  $\xi_1$  und  $\xi_2$  protokolliert.

Die Fehlermaße *maximaler absoluter Fehler*, *mittlerer absoluter Fehler* und *Varianz des mittleren absoluten Fehlers* (siehe Kapitel 2.4) werden für die Auswertung der Testergebnisse angewendet. Da die Winkelfehler sehr klein erwartet werden, ist eine Anwendung dieser Maße nichtsdestotrotz valide. In dem zu erwartenden Wertebereich verhält sich der zyklische Raum quasi linear. Die Ergebnisse der Testläufe sind in Tabelle 4.2 aufgeführt.

Die Genauigkeit im Hinblick auf die Schätzung der Winkel bei den einzelnen Konfigurationen ist in den Abbildungen 4.12 und 4.13 visualisiert. Es ist zu erkennen, dass eine niedrigere Kameraauflösung (in Abbildung 4.12:  $640 \text{ px} \times 480 \text{ px}$ ) zu signifikant höheren Fehlern führt als eine höhere Auflösung (in Abbildung 4.13:  $1920 \text{ px} \times 1080 \text{ px}$ ). Die Informationen hinsichtlich der Grenzen der Genauigkeit des geschätzten Fahrzeugstatus können direkt aus den Ergebnissen abgelesen werden.

Selbstverständlich ist die Statusschätzung mit dem Verfahren nach [BZW06] nur dann möglich, wenn die Kamera alle Markerpunkte erfasst und diese auch korrekt detektiert werden können. Bestimmte Winkelkonstellationen bedingen, dass sich die Markerpyramide (teilweise) außerhalb des für die Kamera sichtbaren Bereichs befindet. Der für die Statusschätzung mögliche Konfigurationsraum ist abhängig von den geometrischen Parametern des Aufbaus und den Eigenschaften der Kamera (Öffnungswinkel etc.). Die in der vorgestellten Konfiguration erfassbare Konfigurationsraum wurde ebenfalls ermittelt

**Simulationstiefe 1** (Ideale Markerpositionen)

Bildbreite (px)		<b>640</b>	<b>1024</b>	<b>1920</b>
Bildhöhe (px)		480	768	1080
$\max  \Delta_E $	$\xi_1$	0.080	0.078	0.009
	$\xi_2$	0.080	0.078	0.009
$\overline{ \Delta_E }$	$\xi_1$	0.029	0.029	0.004
	$\xi_2$	0.029	0.029	0.004
$\text{Var}( \Delta_E )$	$\xi_1$	$4 \cdot 10^{-4}$	$3 \cdot 10^{-4}$	$5 \cdot 10^{-6}$
	$\xi_2$	$4 \cdot 10^{-4}$	$3 \cdot 10^{-4}$	$5 \cdot 10^{-6}$

**Simulationstiefe 2** (Detektierte Markerpositionen)

Bildbreite (px)		<b>640</b>	<b>1024</b>	<b>1920</b>
Bildhöhe (px)		480	768	1080
$\max  \Delta_E $	$\xi_1$	0.386	0.283	0.187
	$\xi_2$	0.532	0.342	0.236
$\overline{ \Delta_E }$	$\xi_1$	0.111	0.070	0.048
	$\xi_2$	0.118	0.078	0.063
$\text{Var}( \Delta_E )$	$\xi_1$	0.004	0.002	0.001
	$\xi_2$	0.008	0.003	0.002

Tabelle 4.2: Evaluierung des 2-D-Verfahrens nach [BZW06]

und ist in Abbildung 4.11 ersichtlich. Jeder Punkt in der Darstellung markiert eine Konfiguration, für die die Statusschätzung berechnet werden konnte.

Um die Toleranz des Verfahrens gegenüber falscher Positionierung der Komponenten zu ermitteln, wird ein künstlicher Fehler für die Positionierung der Kamera definiert. Der sogenannte *künstliche Positionierungsfehler (KPF)* beschreibt die Verschiebung der Kamera orthogonal zur Mittelachse des Zugfahrzeugs bzw. entlang der  $x$ -Achse des Kamera-Koordinatensystems. Eine Bewegung in der Höhe ( $y$ -Achse des Kamera-Koordinatensystems) hätte keinen nennenswerten Einfluss auf die Berechnung, da im Verfahren nach [BZW06] die Annahme des ebenen Bodens verwendet wird. In Tabelle 4.3 sind die Ergebnisse unter Verwendung des *KPF* aufgeführt. Der *KPF* wird dabei in cm angegeben und relativ zur Breite des Zugfahrzeugs, um die Relation im Hinblick auf das Fahrzeuggespann zu verdeutlichen.

Interessant ist in diesem Zusammenhang die offensichtliche Kohärenz der Fehler von  $\xi_1$  und  $\xi_2$ . Bei der Summierung der Winkel im Hinblick auf das unterliegende geometrische Modell heben sich die Fehler auf und es entsteht ein Fehlerverhalten, das von der Evaluati-

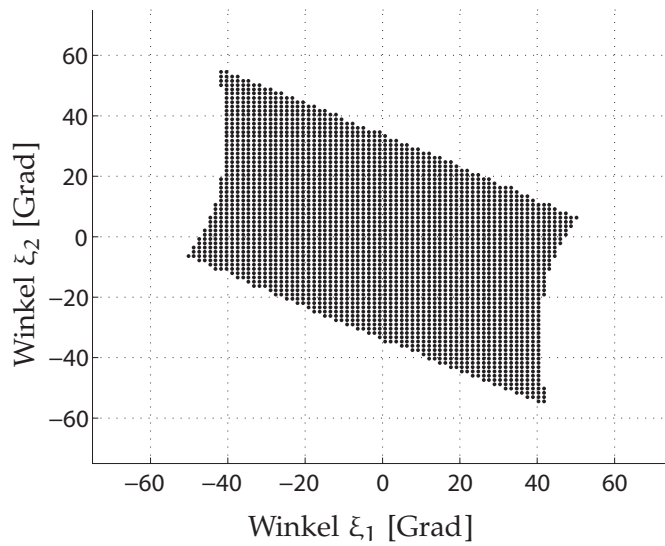


Abbildung 4.11: Konfigurationsraum für eine vollständig detektierbare Markerpyramide und folglich die Berechnung der Statusschätzung bei einer Kameraauflösung von  $640 \text{ px} \times 480 \text{ px}$

on ohne *KPF* bekannt ist. Der Zusammenhang der Fehler ist in Abbildung 4.14 dargestellt. Der durch den *KPF* erzeugte Fehler bedingt eine (virtuelle) relative Verschiebung des Anhängers, die durch die trigonometrischen Beziehungen im Verfahren nach [BZW06] zu erklären ist.

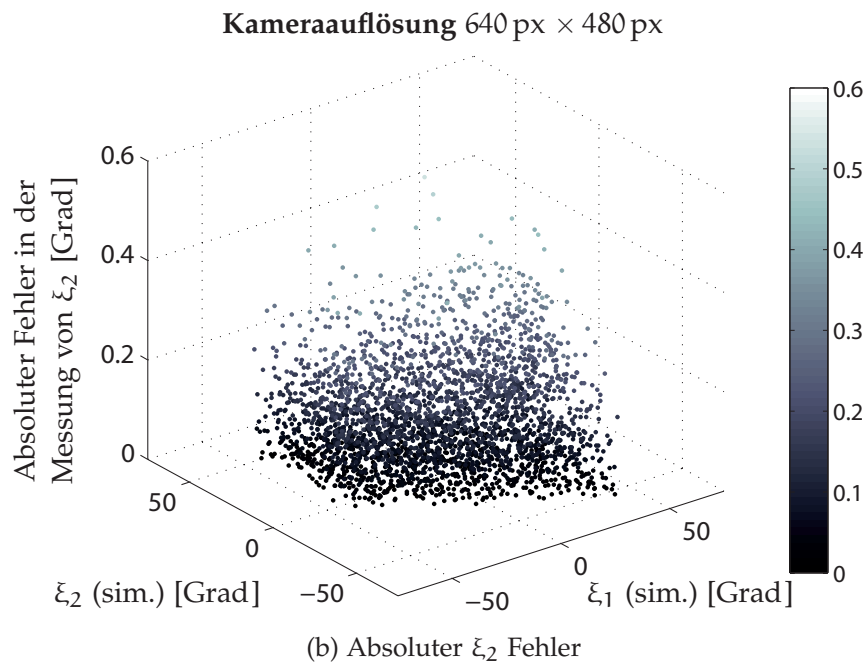
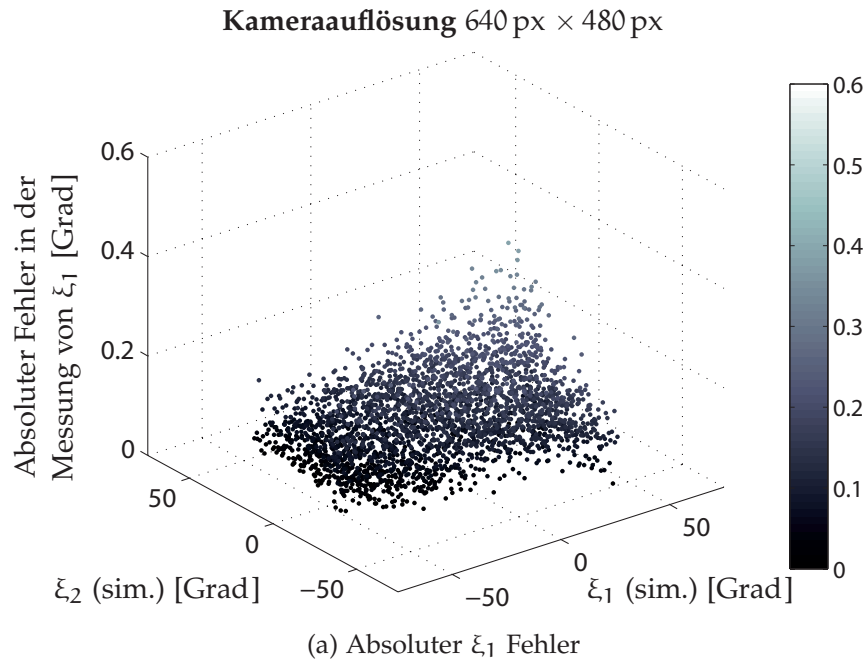


Abbildung 4.12: Absolute Fehler in der Schätzung der Gierwinkel  $\xi_1$  und  $\xi_2$  bei einer Kameraauflösung von 640 px × 480 px mit Markerdetektion



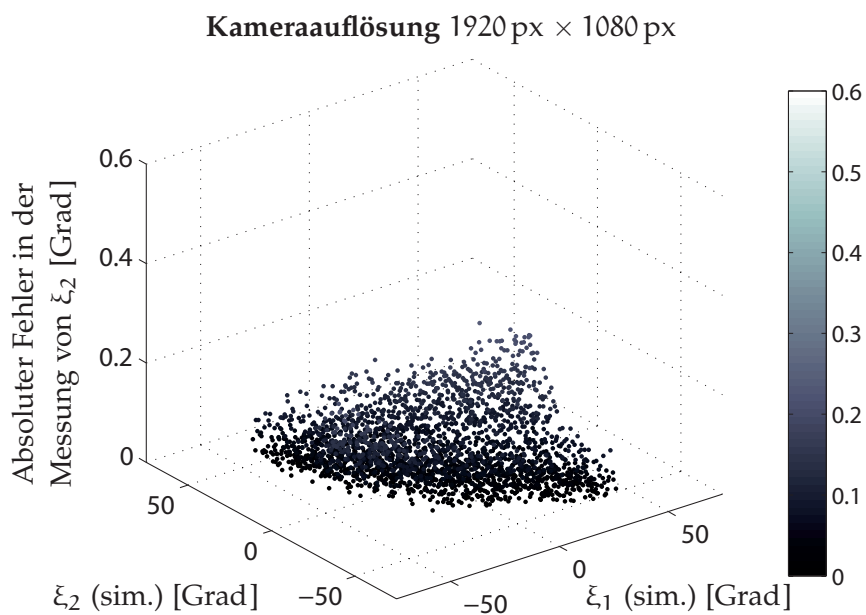
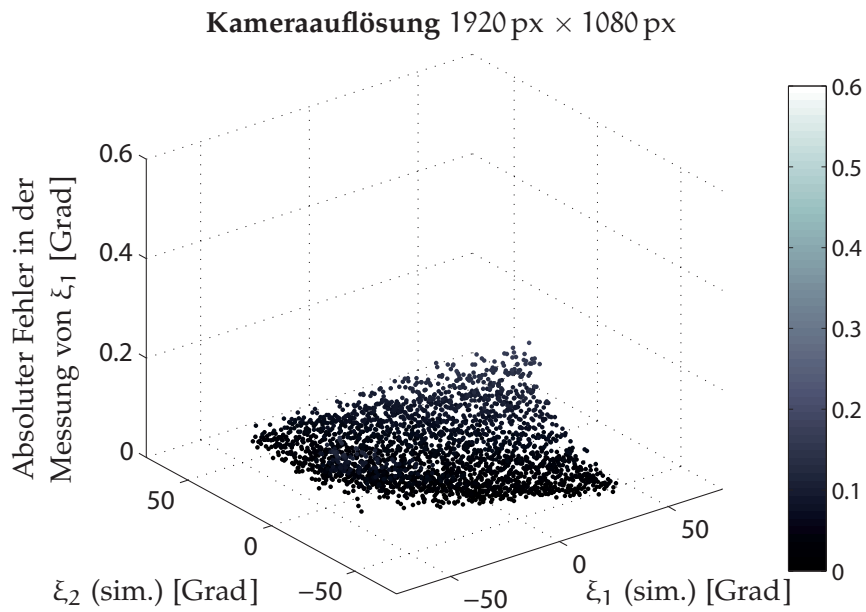


Abbildung 4.13: Absolute Fehler in der Schätzung der Gierwinkel  $\xi_1$  und  $\xi_2$  bei einer Kameraauflösung von 1920 px × 1080 px mit Markerdetektion

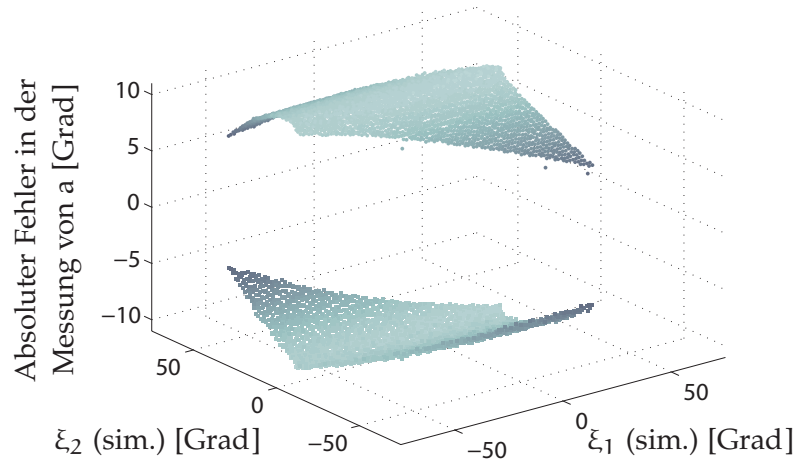


Abbildung 4.14: Fehlerverhalten für einen *KPF* von 3 cm bei Auflösung 640 px × 480 px

**Kameraauflösung 640px × 480px, Markererkennung**

<i>KPF</i> in cm		-5	-4	-3	-2	-1	0
<i>KPF</i> in %		31.3%	25.0%	18.8%	12.5%	6.3%	0%
max $ \Delta_E $	$\xi_1$	17.92	14.31	10.80	7.17	3.73	0.39
	$\xi_2$	18.12	14.49	10.84	7.35	3.90	0.53
$\overline{ \Delta_E }$	$\xi_1$	16.23	12.90	9.60	6.36	3.12	0.12
	$\xi_2$	16.28	12.95	9.66	6.41	3.17	0.12
Var ( $\overline{ \Delta_E }$ )	$\xi_1$	2.69	1.65	0.90	0.40	0.10	0.01
	$\xi_2$	2.70	1.67	0.93	0.41	0.11	0.01
$\overline{ \Delta_{E\xi_1} + \Delta_{E\xi_2} }$		0.12	0.13	0.13	0.13	0.13	0.13

<i>KPF</i> in cm		+1	+2	+3	+4	+5
<i>KPF</i> in %		6.3%	12.5%	18.8%	25.0%	31.3%
max $ \Delta_E $	$\xi_1$	3.80	7.38	10.86	14.43	18.12
	$\xi_2$	3.90	7.42	10.96	14.53	18.10
$\overline{ \Delta_E }$	$\xi_1$	3.32	6.56	9.81	13.09	16.42
	$\xi_2$	3.28	6.50	9.77	13.04	16.37
Var ( $\overline{ \Delta_E }$ )	$\xi_1$	0.10	0.40	0.92	1.67	2.71
	$\xi_2$	0.12	0.42	0.93	1.68	2.74
$\overline{ \Delta_{E\xi_1} + \Delta_{E\xi_2} }$		0.13	0.13	0.13	0.12	0.12

Tabelle 4.3: Evaluationsergebnisse mit *KPF*. Die prozentualen Angaben des *KPF* beziehen sich auf die Fahrzeugbreite.

## 4.5 3-D-ANSATZ

Im Algorithmus nach [BZW06] steckt die Grundannahme, dass sich das artikuliert Fahrzeug auf einer Ebene bewegt. Dies bedeutet, dass lediglich Verschränkungen in Gier-Richtung modelliert werden.

Die Gierwinkel beschreiben die für die Kinematik primär entscheidende Artikulation, jedoch treten auf unebenem Grund auch Verschränkungen in Nick- und Roll-Richtung des Fahrzeugs auf. Werden diese bei der Statusmessung nicht miteinbezogen, liegt ein systematischer Fehler vor, der die ermittelten Gierwinkel verfälscht. Für Fahrzeuge, die auf unebenem Grund – beispielweise in Forst-, Agrar- oder Baustellenszenarien – eingesetzt werden, ist folglich eine Erweiterung erforderlich.

Im Rahmen dieser Problemstellung wurden die in diesem Abschnitt vorgestellten Ergebnisse erarbeitet und publiziert [12]. Die im Folgenden verwendete Notation wurde an die auf Posen beruhende Modellierung angepasst und im Hinblick auf die Veröffentlichung erweitert.

In Bezug auf die in Kapitel 4.3.3 vorgenommene Modellierung des Fahrzeuggespanns lässt sich das Problem der Gespannstatusermittlung auf das Berechnen bzw. Ermitteln der Artikulationsposen  $\mathbf{k}$  und  $\mathbf{g}$  reduzieren. Dabei wird die Kamera bzw. werden die Kamerabilder zur Schätzung der Pose  $\mathbf{e} \in \mathbb{R}^3 \times SO(3)$  des Musters im Kamerakoordinatensystem genutzt. Es lässt sich der folgende Zusammenhang aufstellen (siehe Abbildung 4.15):

$$-\mathbf{c} \oplus \mathbf{p} \oplus \mathbf{k} \ominus \mathbf{d} \ominus \mathbf{g} = \mathbf{e} \ominus \mathbf{m} \oplus \mathbf{t} \quad .$$

Diese Gleichung beinhaltet zwei unbekannte Posen  $\mathbf{k}$  und  $\mathbf{g}$ , die die

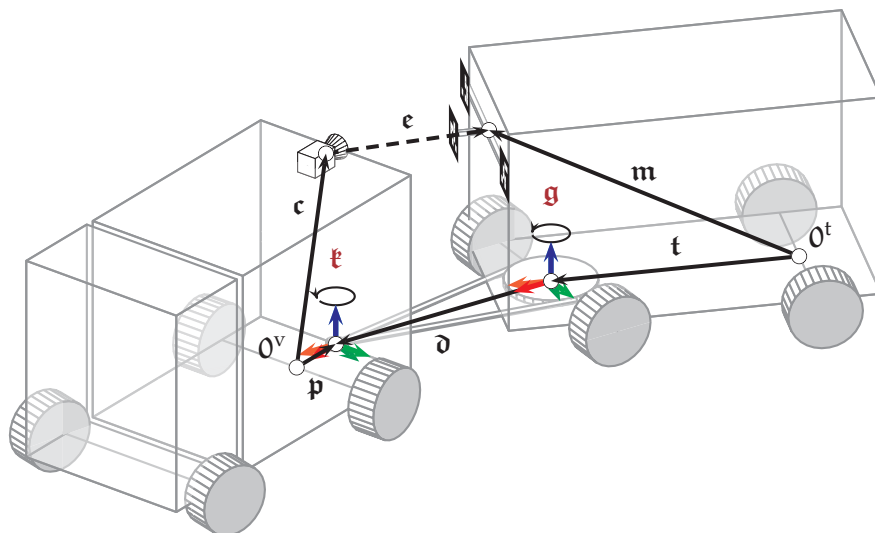


Abbildung 4.15: Modellbildung des Gespannstatus in Verbindung mit der Posenschätzung des Musters am Anhänger durch die Kamera am Zugfahrzeug

Artikulation des Gespanns beschreiben. Da diese Gleichung unterbestimmt ist, lässt sie sich nicht geschlossen und eindeutig lösen.

Werden die beiden Artikulationspunkte vereinfacht als Kugelgelenke modelliert, so treten Rotationen auf, die mit jeweils drei Freiheitsgraden beschrieben werden können. Für die Artikulationsposen  $\mathbf{f}$  und  $\mathbf{g}$  (siehe Kapitel 4.3.3) bedeutet dies, dass die Einschränkung der Rotation um nur eine Achse (siehe Kapitel 4.4) aufgehoben wird.

Bei Zerlegung der Rotationskomponente einer Pose  $\mathbf{a}$  in ihre Eulerwinkel beschreibt der Gierwinkel  $\xi_{\mathbf{a}}$  die Rotation um die z-Achse des Koordinatensystems (siehe Abbildung 4.7). Die Rotation in Nick-Richtung um die y-Achse im Gespann wird durch den Winkel  $\theta_{\mathbf{a}}$  beschrieben, die Rotation in Roll-Richtung um die x-Achse durch den Winkel  $\phi_{\mathbf{a}}$ . Abbildung 4.16 zeigt die Winkel, die an den Kugelgelenken anliegen mit ihrer geometrischen Bedeutung. Die Zuordnung der Achsen zu den entsprechenden Artikulationen entspricht dabei der in Kapitel 4.3 beschriebenen Standard-Ausrichtung des Fahrzeugkoordinatensystems.

Das dargestellte Modell beschreibt die Rotationen an den Gelenken mit sechs Freiheitsgraden. Insbesondere beim Drehschemel ( $\mathbf{g}$ ) liegt die eigentliche Bewegung lediglich im Gierwinkel vor. Die Deichsel als solche verfügt über ein Gelenk in Nick-Richtung, das Höhenunterschiede zum Kupplungsmaul ausgleicht und sich daher im gekuppelten Zustand nur bei unebenem Untergrund bewegt. Sie wird wie in Kapitel 4.3 beschrieben als starr im Hinblick auf ihre Länge angenommen. Da die Bewegungen sowohl in Nick- als auch in Roll-Richtung nur im Fall einer Verscherung des Gespanns auftreten, wird die begründete Annahme getroffen, dass die beiden Winkel in Nick- und Roll-Richtung zu jeweils einem Freiheitsgrad zusammengefasst werden können. Es ergibt sich wie in Abbildung 4.16b und Abbildung 4.16c dargestellt folgende Vereinfachung:

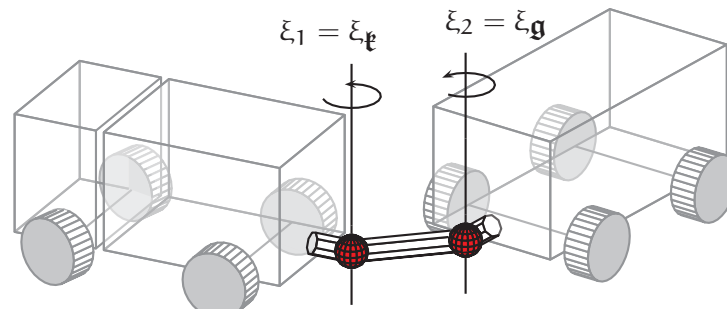
$$\theta = \theta_1 + \theta_2 = \theta_{\mathbf{f}} + \theta_{\mathbf{g}} \quad \phi = \phi_1 + \phi_2 = \phi_{\mathbf{f}} + \phi_{\mathbf{g}} \quad .$$

Die Anzahl der Freiheitsgrade verringert sich somit auf vier.

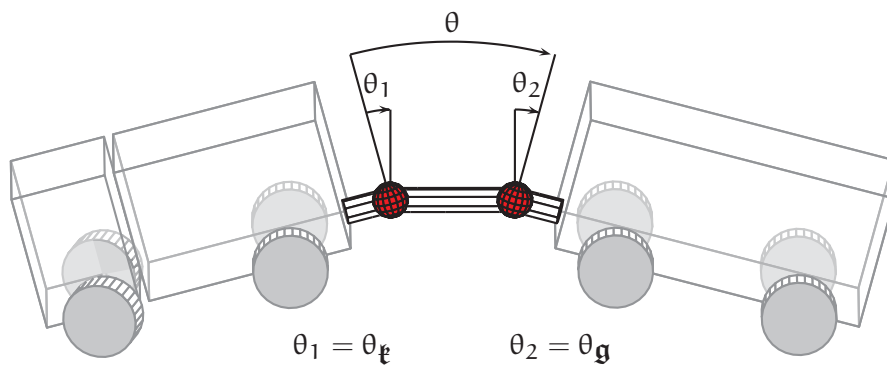
#### 4.5.1 Musterpose und Winkeldekomposition

Für die Schätzung des Gespannzustands ist die Ermittlung der Pose  $\mathbf{e}$  (siehe Abbildung 4.15) des Musters relativ zur Kamera notwendig. Das Vorgehen wird im Folgenden erläutert.

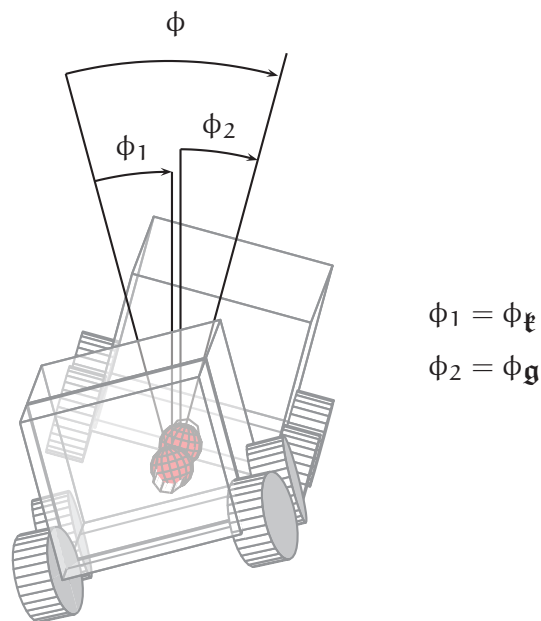
Mittels projektiver Geometrie kann der Zusammenhang zwischen der 3-D-Gestalt des künstlichen Musters und der Abbildung auf die Kameraebene durch den Bildentstehungsprozess hergestellt werden. Das dabei zugrundeliegende Lochkameramodell (siehe Kapitel 3.1) in Kombination mit Kamerakalibrierung (siehe Kapitel 3.2) werden verwendet, um die Projektion zu beschreiben.



(a) Gierwinkel



(b) Nickwinkel



(c) Rollwinkel

Abbildung 4.16: Modellierung der Fahrzeugartikulation im *general-3-trailer* durch Kugelgelenke (vereinfachte Darstellung)

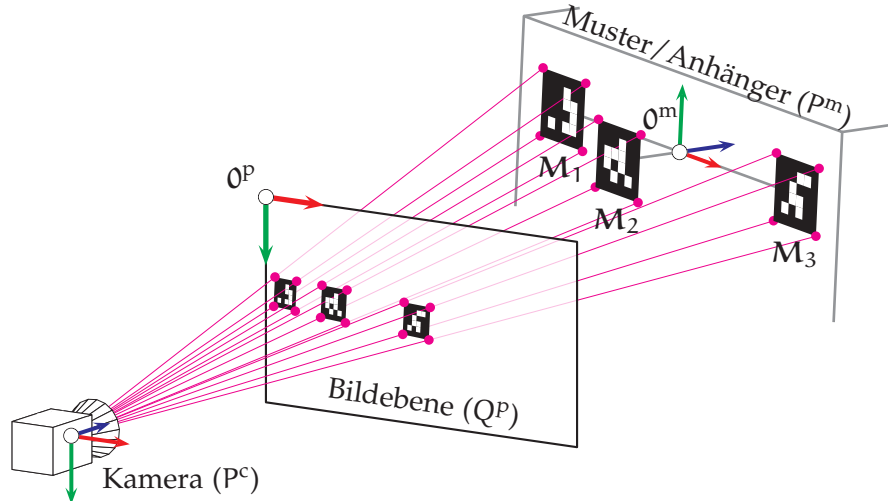


Abbildung 4.17: Projektion des künstlichen Musters auf die Bildebene der Kamera

In Abbildung 4.17 ist die Abbildung der Marker auf die Bildebene der Kamera zu sehen. Die Eckpunkte der Marker im Bild werden bezeichnet durch:

$$Q^P = \left\{ \mathbf{q}^P \mid \mathbf{q}^P = (q_x, q_y)^T \in \mathbb{R}^2 \right\} .$$

Die 3-D-Modellpunkte des Musters werden durch die 3-D-Koordinaten relativ zum Ursprung des Musters beschrieben:

$$P^m = \left\{ \mathbf{p}^m \mid \mathbf{p}^m = (p_x, p_y, p_z)^T \in \mathbb{R}^3 \right\} .$$

Die Modellpunkte  $P^m$  sind durch die Parameter der Markerpyramide und der Markergröße bestimmt. Die Markerpunkte  $P^m$  werden im Abbildungsprozess auf die Bildpunkte  $Q^P$  projiziert. Es gilt:

$$Q^P = \mathcal{P}(P^m) .$$

Die Rückprojektion  $\mathcal{P}^{-1}$  ist somit ein inverses Problem. Eine 3-D-Rückprojektion eines einzelnen Markers  $M_i$  ist möglich, da die Seitenlänge eines Markers bekannt ist. Dieses Vorgehen ist jedoch nicht optimal: Die zu erwartende Rekonstruktionsgenauigkeit ist geringer als die *gemeinsame* Rückprojektion aller Marker im System. Ziel ist die Bestimmung der Relativpose des Musters zur Kamera.

Mit der Formulierung eines gemeinsamen Optimierungsproblems kann die Position des Markersystems relativ zur Kamera direkt bestimmt werden. Die Menge der im Kamerabild detektierten Markereckpunkte  $\mathbf{d}_j$ , d. h. die tatsächlichen Beobachtungen im Kamerabild, wird beschrieben durch

$$D^P = \left\{ \mathbf{d}_j \mid \mathbf{d}_j \in \mathbb{R}^2, j \in \mathbb{N}, 1 \leq j \leq 4 \cdot n_M \right\}$$

mit  $n_M \in \mathbb{N}$  als Anzahl der im Muster angebrachten künstlichen Marker (in Abbildung 4.17 ist  $n_M = 3$ ).

Das Optimierungsproblem minimiert den Abstand zwischen den gemessenen Punkten  $D^P$  und der Projektion  $Q^P$  der Musterpunkte  $P^m$ , die sich an der gesuchten Pose  $\epsilon$  befinden:

$$\epsilon = \underset{\mathbf{a}}{\operatorname{argmin}} \sum_{1 \leq j \leq 4 \cdot n_M} (\|\mathbf{d}_j - \mathcal{P}(\mathbf{T}(\mathbf{a}) \cdot \mathbf{p}_j^m)\|)^2 .$$

Zur Lösung wird ein Optimierungsalgorithmus verwendet, der auf der Methode der minimalen Quadrate basiert [Mar63, Lev44]. Mit Kenntnis der Gespanngeometrie kann die berechnete Relativpose auf die gesuchten vier Freiheitsgrade zurückgeführt werden:

Die Position des Kupplungspunkts am Fahrzeug  $\mathbf{k}_1^c$  ist durch die Pose der Kamera im Fahrzeug ( $\mathbf{c}$ ) und die Pose der Kupplung ( $\mathbf{p}$ ) bestimmt (siehe Abbildung 4.4):

$$\mathbf{k}_1^c = \mathbf{T}(-\mathbf{c} \oplus \mathbf{p}) \cdot \mathbf{0}^T .$$

Gegeben die relative Pose des Musters  $\mathbf{m} \in \mathbb{R}^3 \times SO(3)$  im Anhänger (siehe Abbildung 4.6). Das Zentrum des Drehschemels  $\mathbf{k}_2$  wird durch Kombination der Relativposen des Musters  $\mathbf{m}$  und des Drehschemels  $\mathbf{t}$  im Anhänger beschrieben:

$$\mathbf{k}_2^m = \mathbf{T}(-\mathbf{m} \oplus \mathbf{t}) \cdot \mathbf{0}^T .$$

Aus der geschätzten Pose wird die relative Rotation der Kamera zum Kupplungskoordinatensystem und die relative Rotation des Musters zum Drehschemel herausgerechnet, um die Schätzung auf den reinen Fahrzeugzustand zu normalisieren.

Die geschätzte Musterpose  $\epsilon$  beschreibt die Pose relativ zur Kamera. Die Montagepose der Kamera und die damit einhergehende Rotation muss zunächst herausgerechnet werden.

Der Vektor  $\mathbf{up} \in \mathbb{R}^3$  beschreibe im Folgenden die „nach oben“ (im Sinne des jeweiligen Koordinatensystems) zeigende Referenznormale (engl. *up-vector*) und wird in z-Richtung als Einheitsvektor definiert:

$$\mathbf{up} = (0, 0, 1)^T .$$

Die Pose des Anhängers relativ zum Koordinatensystem des Fahrzeugs wird beschrieben durch die bekannten Posen der Anhängerkupplung  $\mathbf{p}$ , der Kamera am Zugfahrzeug  $\mathbf{c}$ , des Musters im Anhänger  $\mathbf{m}$  und der Lage des Drehschemels im Anhänger  $\mathbf{t}$  sowie der mit dem Optimierungsproblem geschätzten Pose  $\epsilon$ :

$$\mathbf{a} = -\mathbf{p} \oplus \mathbf{c} \oplus \epsilon \ominus \mathbf{m} \oplus \mathbf{t} .$$

Das Koordinatensystem an der Anhängerkupplung sei mit  $\mathbf{k}$  bezeichnet, das Koordinatensystem am Drehschemel mit  $\mathbf{d}$ . Sei nun  $\mathbf{up}^k$  die Referenznormale an der Anhängerkupplung mit

$$\mathbf{up}^k = \mathbf{up} = (0, 0, 1)^T .$$

Die transformierte Referenznormale im Koordinatensystem des Drehschemels ergibt sich durch Transformation mit der zusammengesetzten Pose  $\mathbf{a}$ . Dabei wird lediglich die Rotationskomponente genutzt, da es sich um eine Normale handelt:

$$\begin{aligned}\mathbf{a} &= \langle \mathbf{p}_{\mathbf{a}}, \mathbf{r}_{\mathbf{a}} \rangle \\ \mathbf{up}^d &= \text{rot}(\mathbf{r}_{\mathbf{a}}) \cdot \mathbf{up}^k \quad .\end{aligned}$$

Die akkumulierte Rotation  $\text{rot}(\mathbf{r}_{\mathbf{a}})$  zwischen den Vektoren  $\mathbf{up}^k$  und  $\mathbf{up}^d$  beinhaltet die Artikulation des Fahrzeuggespanns und wird gemäß der vorgestellten Modellierung zerlegt. Die akkumulierte Rotation entspricht der kombinierten Rotationskomponenten der Posen  $\mathbf{k}$  und  $\mathbf{g}$ .

Die Vektoren  $\mathbf{up}^k$  und  $\mathbf{up}^d$  werden im ersten Schritt verwendet, um die Nick- und Rollwinkel zu bestimmen. Die jeweiligen Winkel werden bestimmt, indem die Vektoren auf die jeweilige Ebene projiziert werden. Für den Nickwinkel ist dies die  $xz$ -Ebene, für den Rollwinkel die  $yz$ -Ebene (jeweils im Hinblick auf die Ausrichtung des Fahrzeugkoordinatensystems). Es gilt für die Projektion zur Ermittlung des Nickwinkels  $\theta$ :

$$\mathbf{up}_{\theta}^k = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \mathbf{up}^k \quad \mathbf{up}_{\theta}^d = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \mathbf{up}^d \quad .$$

Analog gilt für die Projektion zur Ermittlung des Rollwinkels  $\phi$ :

$$\mathbf{up}_{\phi}^k = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \mathbf{up}^k \quad \mathbf{up}_{\phi}^d = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \mathbf{up}^d \quad .$$

Die Nick- und Rollwinkel ergeben sich aus:

$$\theta = \text{acos} \left( \frac{\mathbf{up}_{\theta}^k \circ \mathbf{up}_{\theta}^d}{\|\mathbf{up}_{\theta}^k\| \cdot \|\mathbf{up}_{\theta}^d\|} \right) \quad \phi = \text{acos} \left( \frac{\mathbf{up}_{\phi}^k \circ \mathbf{up}_{\phi}^d}{\|\mathbf{up}_{\phi}^k\| \cdot \|\mathbf{up}_{\phi}^d\|} \right) \quad .$$

Um nun die beiden Gierwinkel  $\xi_1$  und  $\xi_2$  zu berechnen, muss die Rotationsinformation um die bereits berechneten Winkel  $\theta$  und  $\phi$  bereinigt werden. Beschreibe die Matrix  $\mathbf{R}_{\theta\phi} \in \mathbb{R}^{3 \times 3}$  eine Rotationsmatrix, die die beiden ermittelten Rotationen in Roll- und Nick-Richtung akkumuliert.

Die Position  $\mathbf{k}_2$  des Zentrums des Drehschemels wird in das Koordinatensystem der Anhängerkupplung  $\mathbf{k}$  transformiert und mittels der Inversen der Rotationsmatrix  $\mathbf{R}_{\theta\phi}$  um die Rotation in Roll- und Nick-Richtung bereinigt:

$$\mathbf{a}_1 = \mathbf{R}_{\theta\phi}^{-1} \cdot \mathbf{k}_2 \quad .$$



Der Differenzvektor  $\mathbf{a}_2$  zwischen  $\mathbf{k}_1$  und  $\mathbf{k}_2$  auf der Ebene approximiert die Deichsel und ergibt:

$$\mathbf{a}_2 = \mathbf{k}_1^k - \mathbf{a}_1$$

Um die Gierwinkel zu berechnen, ist die Normalausrichtung bzw. die in Geradeausfahrt zu erwartende Nullstellung als Richtungsvektor erforderlich. Gemäß den in den Abbildungen 4.4 und 4.6 gezeigten Ausrichtungen der Koordinatensysteme  $\mathbf{k}$  und  $\mathbf{d}$  ergibt sich die rückwärtige Normalausrichtung am Zugfahrzeug  $\mathbf{b}_1 \in \mathbb{R}^3$  und die vorwärtige Ausrichtung am Drehschemel  $\mathbf{b}_2 \in \mathbb{R}^3$ :

$$\mathbf{b}_1^k = (-1, 0, 0)^T \quad \mathbf{b}_2^d = (1, 0, 0)^T \quad .$$

Die Rotationen in Nick- und Roll-Richtung werden aus der Neutralrichtung des Anhängers im Koordinatensystem des Zugfahrzeugs entfernt:

$$\mathbf{a}_3 = \mathbf{R}_{\theta\phi}^{-1} \cdot \mathbf{b}_2^k \quad .$$

Die Gierwinkel  $\xi_1$  und  $\xi_2$  lassen sich durch Bestimmung der Winkel zwischen diesen Vektoren ermitteln:

$$\xi_1 = \text{acos} \left( \frac{\mathbf{b}_1^k \circ \mathbf{a}_2}{\|\mathbf{b}_1^k\| \cdot \|\mathbf{a}_2\|} \right) \quad \xi_2 = \text{acos} \left( \frac{\mathbf{a}_3 \circ \mathbf{a}_2}{\|\mathbf{a}_3\| \cdot \|\mathbf{a}_2\|} \right)$$

#### 4.5.2 Evaluation

Um das 3-D-Verfahren zu evaluieren wird die Simulationsumgebung aus Kapitel 4.4.2 um 3-D-Gespannkonfigurationen erweitert. Die ermittelten Winkelgenauigkeiten werden mit den Genauigkeiten des 2-D-Ansatzes verglichen. Die Nick- und Rollwinkel  $\theta$  und  $\phi$  werden in das virtuelle Modell integriert und zur Ergänzung der Schnittstelle genutzt.

Um die Intrinsik der virtuellen Kamera zu beschreiben, muss ein Konfigurationsvektor definiert werden. Sei  $(w, h)^T \in \mathbb{N}^2$  die Bildauflösung der virtuellen Kamera. Der Hauptpunkt kann entsprechend auf

$$c_x = \frac{w}{2} \quad \text{bzw.} \quad c_y = \frac{h}{2}$$

gesetzt werden.

Der Verscherungskoeffizient ist trivial 0, da die Pixel der virtuellen Kamera als rechteckig festgelegt werden. Die Kamerakonstanten werden in beide Richtungen als äquivalent definiert ( $k = k_x = k_y$ ). In Abhängigkeit vom horizontalen Kameraöffnungswinkel  $\alpha$  und der Bildbreite  $w$  gilt:

$$k = \left( \frac{2}{w} \cdot \tan \left( \frac{\alpha}{2} \right) \right)^{-1} \quad .$$

Für die Intrinsic-Matrix ergibt sich:

$$\mathbf{K} = \begin{pmatrix} k & 0 & c_x \\ 0 & k & c_y \\ 0 & 0 & 1 \end{pmatrix} .$$

Um die Vergleichbarkeit mit den in Kapitel 4.4.2 vorgestellten Ergebnissen zu ermöglichen, werden die dort verwendeten Parameter für die Spann- und Mustergeometrie adaptiert (siehe Tabelle 4.1). Die Winkel-Parametrierung der Simulation wird durch den Vektor

$$\mathbf{g} = (\xi_1, \xi_2, \theta, \phi)^T$$

beschrieben. Das Ergebnis der Winkelschätzung wird durch den Spannungszustandsvektor  $\mathbf{s} \in \mathbb{R}^4$  repräsentiert:

$$\mathbf{s} = (\xi'_1, \xi'_2, \theta', \phi')^T .$$

Die Ermittlung der Genauigkeit der Verfahren wird durch den Vergleich der konfigurierten ( $\mathbf{g}$ ) und geschätzten ( $\mathbf{s}$ ) Werte errechnet. Zur Anwendung kommen dabei die in Kapitel 2.4 beschriebenen Fehlermaße *maximaler absoluter Fehler*, *mittlerer absoluter Fehler* und die *Varianz des mittleren absoluten Fehlers*. Die genannten Maße sind für euklidische Räume definiert. Die Aussagen bezüglich der Korrektheit der Anwendung dieser Fehlermaße im Hinblick auf Winkel gelten entsprechend wie in Kapitel 4.4.2 beschrieben. Die berechneten Qualitätsmaße sind:

Fehlermaß	Bezeichnung
$\max  \Delta_{E\beta} $	Maximaler absoluter Fehler
$\overline{ \Delta_{E\beta} }$	Mittlerer absoluter Fehler
$\text{Var}(\overline{ \Delta_{E\beta} })$	Varianz des mittleren absoluten Fehlers

Für den Winkel  $\beta$  gilt dabei  $\beta \in \{\xi_1, \xi_2, \theta, \phi\}$ . Für die Evaluation wird in jedem Schritt die Geometrieconfiguration  $\mathbf{g}$  sowie der durch das Verfahren ermittelte Zustandsvektor  $\mathbf{s}$  gespeichert.

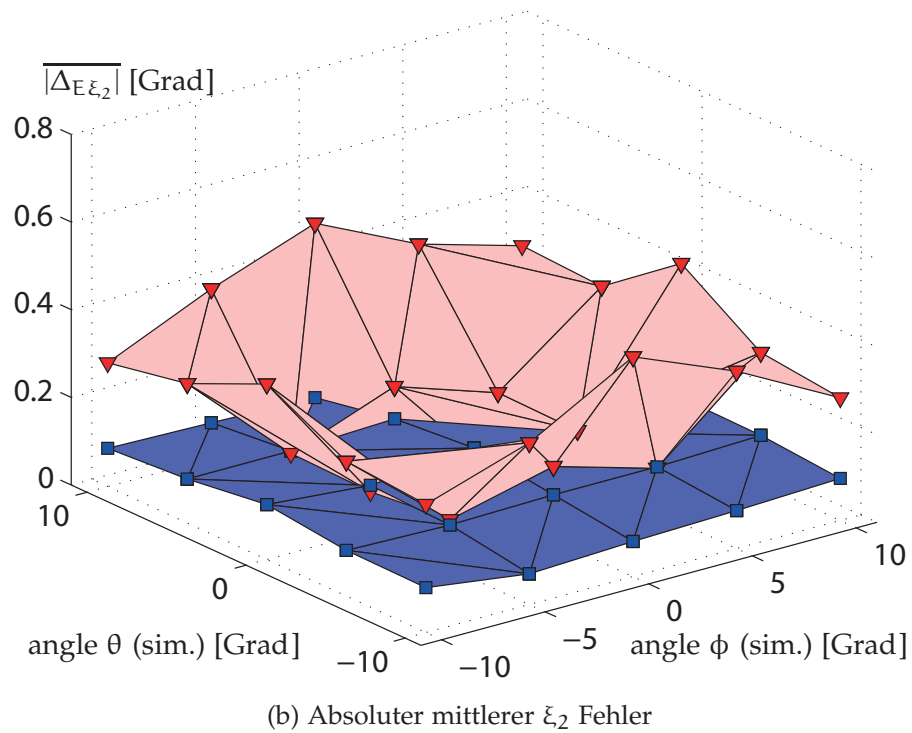
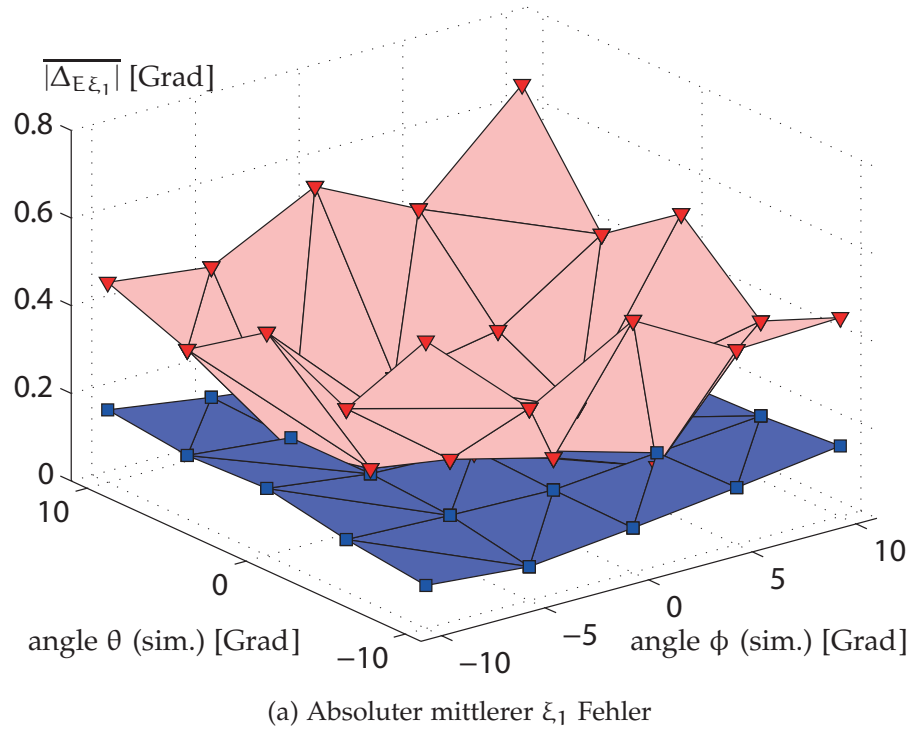
In einem ersten Testdurchlauf wird die Genauigkeit der Winkelschätzung bei einer Bewegung auf ebenem Grund ( $\theta = 0 = \phi$ ) evaluiert. Die Gierwinkel  $\xi_1$  und  $\xi_2$  werden jeweils im Bereich  $[75^\circ; 75^\circ]$  mit je 100 Werten (Schrittweite  $1.5^\circ$  mit 10 000 Abtastungen pro Durchlauf) abgetastet. Die Ergebnisse unter Verwendung unterschiedlicher Kameraauflösungen sind in Tabelle 4.4 aufgeführt. Die Werte zeigen, dass eine höhere Auflösung eine bessere Stabilität in Bezug auf die Fehlermaße aufweist.

In einem zweiten Testdurchlauf werden Roll- und Nickzustände konfiguriert. Die Genauigkeit der Rekonstruktion der Roll- und Nickzustände durch das 3-D-Verfahren ist in Abbildung 4.19 dargestellt.

Bildbreite (px) Bildhöhe (px)		640		1024		1920	
		480		768		1080	
Methode		2-D	3-D	2-D	3-D	2-D	3-D
$\max  \Delta_E $	$\xi_1$	0.386	0.411	0.283	0.422	0.187	0.448
	$\xi_2$	0.532	0.678	0.342	0.559	0.236	0.444
	$\theta$	-	0.939	-	0.169	-	0.158
	$\phi$	-	0.378	-	0.098	-	0.088
$\overline{ \Delta_E }$	$\xi_1$	0.111	0.109	0.070	0.114	0.048	0.117
	$\xi_2$	0.118	0.118	0.078	0.145	0.063	0.140
	$\theta$	-	0.060	-	0.044	-	0.140
	$\phi$	-	0.001	-	0.009	-	0.032
$\text{Var}( \Delta_E )$	$\xi_1$	0.004	0.004	0.002	0.006	0.001	0.008
	$\xi_2$	0.008	0.008	0.003	0.012	0.002	0.001
	$\theta$	-	0.001	-	0.000	-	0.000
	$\phi$	-	0.000	-	0.000	-	0.000

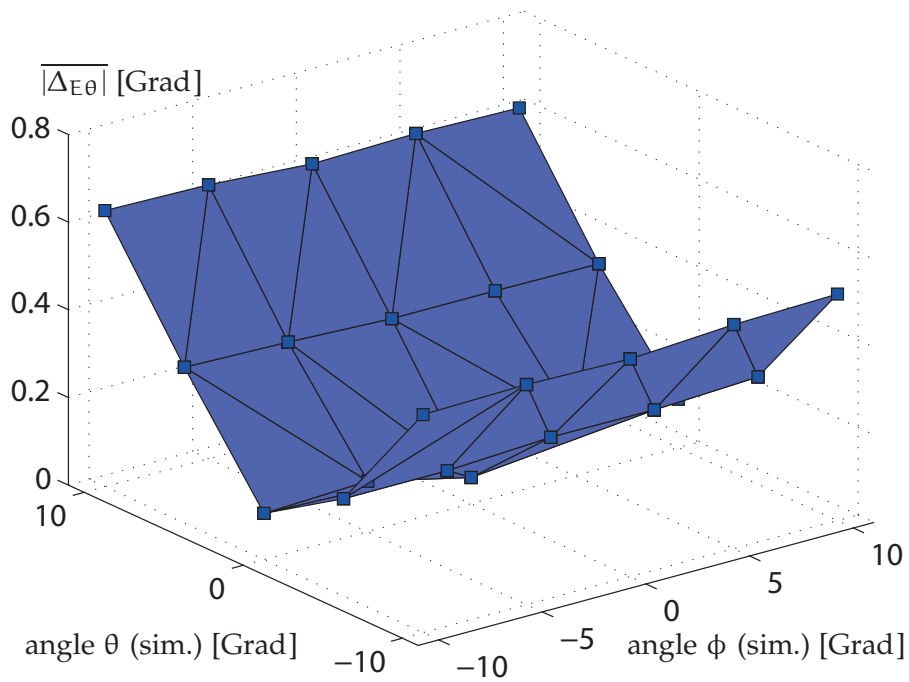
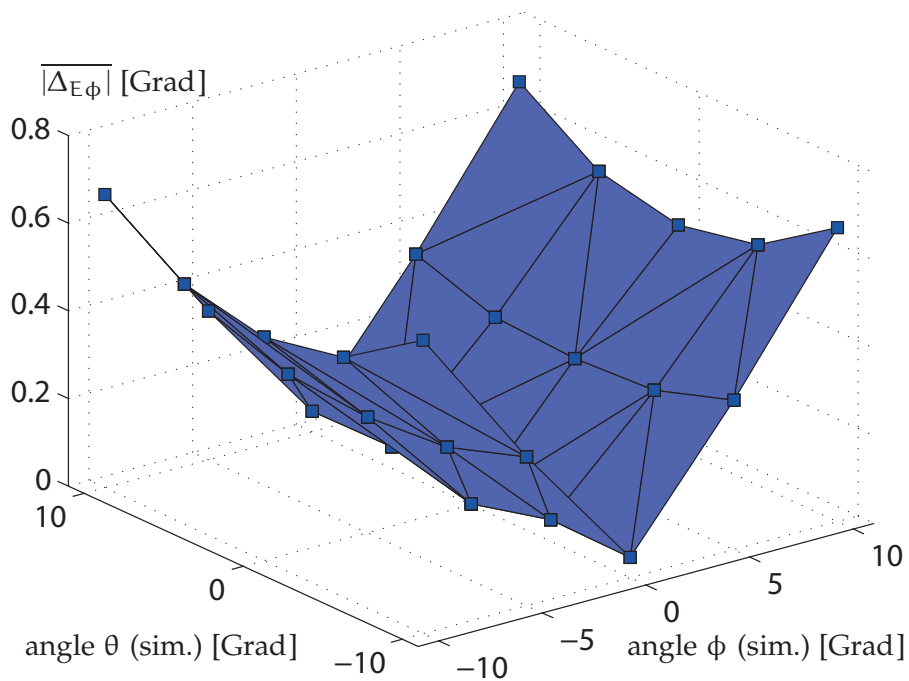
Tabelle 4.4: Evaluationsergebnisse (in Grad) für  $\theta = \phi = 0$ 

Die Ergebnisse bzgl. der Gierwinkel zeigen deutliche Verbesserungen in der Präzision, sobald Roll- und Nickzustände eintreten. Die Verletzung der Annahme des ebenen Bodens im 2-D-Verfahren bedingt Verfälschungen in den Ergebnissen für die Gierwinkel in diesen Fällen. Der Sachverhalt ist in Abbildung 4.18 zu sehen: Mit zunehmendem Roll- bzw. Nick-Winkel steigt der Fehler in der Messung der Gierwinkel  $\xi_1$  und  $\xi_2$ . Gilt jedoch  $\phi \approx 0 \approx \theta$ , zeigt der 2-D-Ansatz geringfügig bessere Ergebnisse. Dies ist der Tatsache geschuldet, dass durch das Zutreffen der Vereinfachung hinsichtlich der ebenen Bodenannahme eine höhere numerische Präzision erreicht wird, da die Roll- und Nickwinkel nicht mitgeschätzt werden.



▼ 2-D Sensor      ■ 3-D Sensor

Abbildung 4.18: Fehler in der Gierwinkelschätzung in Abhängigkeit zu den Nick- und Rollwerten  
(Bildauflösung 640 px × 480 px)

(a) Absoluter mittlerer  $\theta$  Fehler(b) Absoluter mittlerer  $\phi$  Fehler

■ 3-D Sensor

Abbildung 4.19: Fehler in der Nick- und Rollwinkelschätzung in Abhängigkeit zu den konfigurierten Nick- und Rollzuständen („unebener“ Boden)  
(Bildauflösung 640 px × 480 px)

## 4.6 TEMPORALE KOHÄRENZ

Der in Kapitel 4.5 vorgestellte Ansatz zur Ermittlung des 4-D-Gespannzustands [12] nutzt einzelne Kamerabilder für die jeweils voneinander unabhängige Schätzung der Fahrzeugartikulation.

Da die Schätzung des Gespannzustands fortlaufend vorgenommen wird, kann die Bildsequenz genutzt werden, um vorangegangene Messungen zur Verbesserung der Zustandspräzision zu nutzen. Die im Folgenden vorgestellten Erweiterungen wurden bereits auf internationalen Konferenzen vorgestellt (Integration temporaler Kohärenz [7]; Integration des Ansatzes und Evaluation gegen andere Verfahren [6]).

Die kontinuierliche Aufnahme des Musters am Anhänger impliziert einen kurzen zeitlichen Abstand zwischen den einzelnen Momentaufnahmen bzw. Kamerabildern. Der zeitliche Abstand ist dadurch durch die Bildrate der Kamera (beispielsweise 20 Hz mit  $\Delta_t = 0.05$  s) vorgegeben. Im kurzen Zeitraum zwischen zwei Einzelbildern ist zu erwarten, dass die Bewegung des Musters in jeweils kleinen Schritten erfolgt, sodass beim kontinuierlichen Betrachten ein Film nach dem Prinzip des Daumenkinos entsteht.

Die jeweils relativen, zwischen zwei Einzelbildern vollzogenen kleinen Bewegung des Musters in Kombination mit dem Zusammenhang in der Bildsequenz kann genutzt werden, um ein Bewegungsmodell aufzustellen. Das Bewegungsmodell wird genutzt, um die zu erwartenden Veränderungen aus den vorangegangenen Einzelbewegungen zu schätzen bzw. eine Bewegungskontinuität zu implizieren.

Rudolph E. Kálmán hat mit der Beschreibung des nach ihm benannten Kalman-Filters [Kal60] grundlegende Arbeit für die Integration von Messreihen mit einer zugrunde liegenden Bewegungsmodellierung geleistet. Das Verfahren nach Kalman findet im Bereich der Objektverfolgung (engl. *tracking*) verbreitete Anwendung.

Eine angepasste Variante des Kalman-Filters findet für die erweiterte Gespannstatusschätzung Anwendung. Die genaue Anpassung und die zugrundeliegende Bewegungsmodellierung werden im Folgenden erörtert:

Der Kalman-Filter nutzt einen Systemzustand  $x$ , der als mehrdimensionaler Vektor die für die Messung relevanten Größen beschreibt. Zusätzlich beschreibt ein Bewegungsmodell den Einfluss der Messgrößen aufeinander bei fortschreitender Zeit und eine Kovarianzmatrix  $W$  die Konfidenz hinsichtlich des Zustands. Ein einfaches Beispiel ist die Veränderung einer zurückgelegten Strecke, die sich durch einen Zeitschritt in Kombination mit einer Geschwindigkeit errechnet. Das Bewegungsmodell wird genutzt, um eine Zustandsdynamik zu realisieren. Im klassischen Kalman-Filter [Kal60] ist das Bewegungsmodell linear. Eine Erweiterung auf nicht-lineare Bewegungsmodelle [JU04] wird u. a. durch den *Extended Kalman Filter* vor-

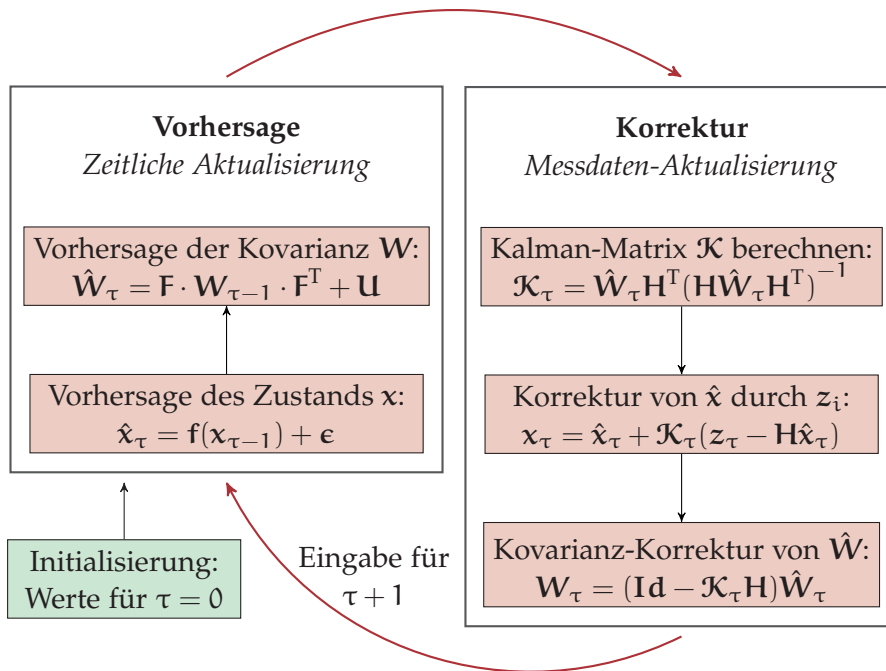


Abbildung 4.20: Schematischer Aufbau des Kalman-Filters

genommen.

Der Kalman-Filter besteht aus zwei Phasen: einer Vorhersage- und einer Korrekturphase. Ein neuer Systemzustand zum Zeitpunkt  $\tau + 1$  kann durch Anwendung des Bewegungsmodell auf den Systemzustand  $x$  zum Zeitpunkt  $\tau$  und unter Einbeziehung aller vorangegangenen Messwerte vorhergesagt werden (engl. *prediction*). Dabei wird auch eine Vorhersage der Kovarianz  $W$  berechnet.

Sobald eine neue Messung vorliegt, wird diese genutzt, um den Systemzustand zu korrigieren (engl. *correction/update*) und die Kovarianzen der Messung neu zu bestimmen. Die Kalman-Matrix  $\mathcal{K}$  (in der Literatur auch als *Kalman-Gain* bezeichnet) beschreibt dabei die Gewichtung zwischen Prädiktion und Messung und wird ebenfalls mit jeder Iteration neu berechnet.

Die beiden Hauptschritte des Kalman-Filters mit ihren Unterschritten sind in Abbildung 4.20 schematisch dargestellt. Der Kalman-Filter wird als Grundlage für die Integration temporaler Kohärenz für die Schätzung des Gespannzustands verwendet. Die genauen Schritte und die Übertragung auf das vorliegende Problem werden im Folgenden erläutert.

Das lineare Bewegungsmodell des Kalman-Filters ist für 6- $D$ -Pose-schätzung nicht generisch anwendbar, da im damit einhergehenden Systemzustand eine zyklische Komponente für die Modellierung der Orientierungskomponente enthalten ist.

Auf Basis der Erweiterung zum *Unscented Kalman Filter* [JU04], der

die Integration von Quaternionen in den Kalman-Filter beschreibt, erweitern Hertzberg et al. [HWFS13] das mathematische Modell hinter dem Kalman-Filter um Mannigfaltigkeiten (engl. *manifolds*). Diese Erweiterung ermöglicht die mathematisch fundiertere Formulierung der Posenschätzung:

Die Autoren erweitern die Vektorrauminterpretation des  $SO(3)$  (siehe Kapitel 2.2). Insbesondere Orientierungen im 3-D-Raum, wie z. B. als Teil einer Pose, verhalten sich lediglich lokal wie Vektoren, nicht jedoch global bzw. allgemein betrachtet. Hertzberg et al. schlagen eine Kapselung des  $SO(3)$  vor, die eine Behandlung analog zu einem klassischen Vektorraum ( $\mathbb{R}^n$ ) ermöglicht, wobei eine entsprechende Verwendung mathematisch korrekt erfolgt. Die Erweiterung nach Hertzberg et al. [HWFS13] wird in der vorliegenden Arbeit verwendet.

#### 4.6.1 Modellierung des Systemzustands

Der Systemzustand des Kalman-Filters wird mit dem Vektor  $\mathbf{x} \in \mathbb{R}^{13}$  zusammengefasst. Dieser enthält die geschätzte Pose des künstlichen Musters  $\mathbf{e} \in \mathbb{R}^3 \times SO(3)$  und die Geschwindigkeiten entsprechend dem Bewegungsmodell. Zur Vereinfachung wird dabei die Pose  $\mathbf{e} = \langle \mathbf{p}_e, \mathbf{r}_e \rangle$  als 7-D-Vektor mit sechs Freiheitsgraden (wegen des Einheitsquaternions) aufgefasst. Das enthaltene Quaternion wird mit seinen Komponenten als 4-D-Vektor notiert.

Die korrekten Rechenregeln werden bei dieser Darstellung durch die Integration von Manifolds [HWFS13] entsprechend beachtet. Der Systemzustand wird formuliert als:

$$\mathbf{x} = (\mathbf{e}^T, \mathbf{v}^T, \boldsymbol{\omega}^T)^T = (\mathbf{p}_e^T, \mathbf{r}_e^T, \mathbf{v}^T, \boldsymbol{\omega}^T)^T$$

mit

$$\begin{array}{ll} \mathbf{e} \in \mathbb{R}^3 \times SO(3) & \text{Pose des künstlichen Musters relativ zur Kamera,} \\ \mathbf{v} \in \mathbb{R}^3 & \text{Geschwindigkeit (Translation),} \\ \boldsymbol{\omega} \in \mathbb{R}^3 & \text{Winkelgeschwindigkeit (Orientierung).} \end{array}$$

Der Vorhersage-Schritt (engl. *prediction step*) zur Prädiktion des nächsten Zeitschritts  $\tau + 1$  nutzt die Funktion  $f : \mathbb{R}^{13} \rightarrow \mathbb{R}^{13}$  zur Schätzung des neuen Status basierend auf dem Bewegungsmodell.

Als Bewegungsmodell wird an dieser Stelle ein Modell mit der Annahme von konstanter Geschwindigkeit verwendet (engl. *constant velocity assumption*). Dies bedeutet, dass von keiner Geschwindigkeitsänderung zwischen den einzelnen Zeitschritten ausgegangen wird. Übertragen auf das Problem der Posenschätzung wird davon ausgegangen, dass diese konstant bleibt bzw. sich konstant weiterbewegt. Lenkbewegungen bzw. die Veränderung des Fahrzeugzustands werden als „Störungen“ in den Kalman-Filter integriert und bilden somit die Bewegungseigenschaften ab.



Der Zeitschritt wird mit der Variable  $\tau \in \mathbb{N}$  bezeichnet, die vergangene Zeit seit dem letzten Zeitschritt durch  $\Delta_t \in \mathbb{R}^+$ . Es ergibt sich:

$$\hat{\mathbf{x}}_{\tau+1} = \mathbf{f}(\mathbf{x}_\tau) = \begin{pmatrix} \mathbf{p}_{\mathbf{e}_\tau} + \Delta_t \cdot \mathbf{v}_\tau \\ \mathbf{r}_{\mathbf{e}_\tau} \cdot \mathcal{Q}(\Delta_t \cdot \boldsymbol{\omega}_\tau) \\ \mathbf{v}_\tau \\ \boldsymbol{\omega}_\tau \end{pmatrix} + \boldsymbol{\epsilon} = \mathbf{F} \cdot \mathbf{x}_\tau + \boldsymbol{\epsilon} \quad .$$

Die Funktion  $\mathcal{Q}$  extrapoliert die neue Orientierung  $\mathbf{r}_{\mathbf{e}}$  anhand des Zeitschritts  $\Delta_t$  und der Winkelgeschwindigkeit  $\boldsymbol{\omega}$ . Der Vektor  $\boldsymbol{\epsilon}$  beschreibt das additive Rauschen.

Die Unsicherheit des Status  $\mathbf{x}$  wird mit der Kovarianzmatrix  $\mathbf{W}$  beschrieben. Sie wird mit der Unsicherheit des anfänglichen Zustands initialisiert. Die Matrix  $\mathbf{W}$  wird während des Trackings aktualisiert und enthält die Konfidenz der geschätzten Pose. Zusätzlich beinhaltet die Prozess-Rausch-Kovarianzmatrix  $\mathbf{U}$  Einschränkungen für unerwartete Statusänderungen (z.B. plötzliche Windböen oder ähnliche Einflüsse auf den Anhänger). Nach jeder Vorhersage wird die Matrix  $\mathbf{W}$  wie folgt aktualisiert:

$$\hat{\mathbf{W}}_{\tau+1} = \mathbf{F} \cdot \mathbf{W}_\tau \cdot \mathbf{F}^T + \mathbf{U}$$

mit  $\mathbf{F}$  als die linearisierte Veränderung des Status durch  $\mathbf{f}$  zum Zeitpunkt  $\tau$ .

Das Messmodell  $\mathbf{h}_i$  berechnet die Rückprojektion des  $i$ -ten Markereckpunkts ( $\zeta$  ist das Messrauschen) mit  $T_{w \rightarrow c}$  und  $\mathcal{P}$ :

$$\mathbf{h}_i(\mathbf{x}) = \mathcal{P}(T_{w \rightarrow c}(\mathbf{p}_i^c)) + \zeta \quad .$$

Die Pixelkoordinaten der Eckpunkte der im Bild extrahierten Marker werden als rohe Messungen interpretiert. Diese werden aus dem Kamerabild als Eingabe extrahiert und mit den vorhergesagten Pixelkoordinaten verglichen. Das Messmodell  $\mathbf{h}_i$  wird im Korrekturschritt des Kalman-Filters verwendet, da die Beobachtungen im 2-D-Raum genutzt werden, um die Pose im Status  $\mathbf{x}$  zu verifizieren. Die Messmatrix  $\mathbf{H}$  beschreibt die Gesamtheit der im Hinblick auf den Status  $\mathbf{x}$  gemessenen Werte. Die Kalman-Matrix  $\mathcal{K}_\tau$  wird berechnet durch:

$$\mathcal{K}_\tau = \mathbf{W}_\tau \cdot \mathbf{H}^T \cdot (\mathbf{H} \cdot \mathbf{W}_\tau \cdot \mathbf{H}^T)^{-1} \quad .$$

Die Schätzung wird nun mit den detektierten Markereckpunkten  $\mathbf{q}_j \in \mathcal{Q}^i$  aktualisiert, die in der Matrix  $\mathbf{Z}_\tau$  zusammengefasst werden. Mit diesen Werten wird die Kalman-Innovation  $\mathbf{v}$  berechnet:

$$\mathbf{v}_\tau = \mathbf{Z}_\tau - \mathbf{H} \cdot \hat{\mathbf{x}}_\tau \quad .$$

Die Kalman-Innovation  $\mathbf{v}$  wird zusammen mit der Kalman-Matrix  $\mathcal{K}$  als Gewichtung genutzt, um den geschätzten Systemzustand  $\hat{\mathbf{x}}_t$  zu korrigieren:

$$\mathbf{x}_\tau = \hat{\mathbf{x}}_\tau + \mathcal{K}_\tau \cdot \mathbf{v} \quad .$$

Letztendlich wird die Fehlerkovarianz  $\mathbf{W}$  mittels der Kalman-Matrix  $\mathcal{K}$  aktualisiert:

$$\mathbf{W}_\tau = (\mathbf{Id} - \mathcal{K}_\tau \cdot \mathbf{H}) \cdot \hat{\mathbf{W}}_\tau \quad .$$

Die vorangegangenen Schritte werden mit den durch die Messsequenz eintreffenden Messwerte für jede Iteration neu durchgeführt. Durch das iterative Anpassen der Kovarianzen und Konfidenzen wird ein insgesamt stabileres Systemverhalten erreicht.

Die im Systemzustand des Kalman-Filters enthaltene Pose  $\epsilon$  kann nun genutzt werden, um den Fahrzeugzustand zu beschreiben. Die Zerlegung in die einzelnen Komponenten des 4-D-Gespannzustands erfolgt analog zu der Darstellung in Kapitel 4.5.

#### 4.6.2 Evaluation

Um die Vergleichbarkeit mit den Ergebnissen in den Kapiteln 4.4 und 4.5 zu gewährleisten wird zur Evaluation des Systems ein analoger Aufbau im Hinblick auf die Marker- und Gespannkonfiguration gewählt. Der vorgestellte Algorithmus mit dem Kalman-Filter setzt die Anordnung der Marker in einer Pyramidenform nicht zwingend voraus. Vielmehr ist jegliche Markerkonfiguration mit einem oder mehreren Markern anwendbar.

Da der Kalman-Filter eine Bildsequenz zur Nutzung der temporalen Kohärenz voraussetzt, werden für die Ansteuerung der Simulationsumgebung (siehe Kapitel 4.4) realistische Bewegungen des Gespanns benötigt. Es werden daher kontinuierliche Funktionen für jeden der Artikulationsparameter definiert, die jeweils äquidistant abgetastet werden.

Für einen gegebenen Zeitpunkt  $t \in \mathbb{R}^+$  im Simulationslauf berechnet eine Funktion  $\varphi : \mathbb{R}^+ \rightarrow \mathbb{R}^4$  den Artikulationszustand als Eingabe für die Simulationsumgebung. Die Funktion  $\varphi$  berechnet einen Konfigurationsvektor:

$$\varphi(t) = (\varphi_{\xi_1}(t), \varphi_{\xi_2}(t), \varphi_\theta(t), \varphi_\phi(t))^T \quad .$$

Für die vorgestellte Evaluation wurden folgende Funktionen für die einzelnen Komponenten verwendet:

$$\begin{aligned} \varphi_{\xi_1}(t) &= 30 \cdot \sin\left(\frac{1}{6} \cdot t\right) \\ \varphi_{\xi_2}(t) &= 15 \cdot \cos\left(\frac{1}{6} \cdot t\right) \\ \varphi_\theta(t) &= 80 \cdot (\mathcal{N}_{25,2}(t) - \mathcal{N}_{25,4}(t)) \\ \varphi_\phi(t) &= -40 \cdot (\mathcal{N}_{60,2}(t) - \mathcal{N}_{60,6}(t)) \quad . \end{aligned}$$

Eine graphische Darstellung der Funktionsverläufe ist in Abbildung 4.21 ersichtlich. Für die Gierwinkel wird eine kontinuierliche

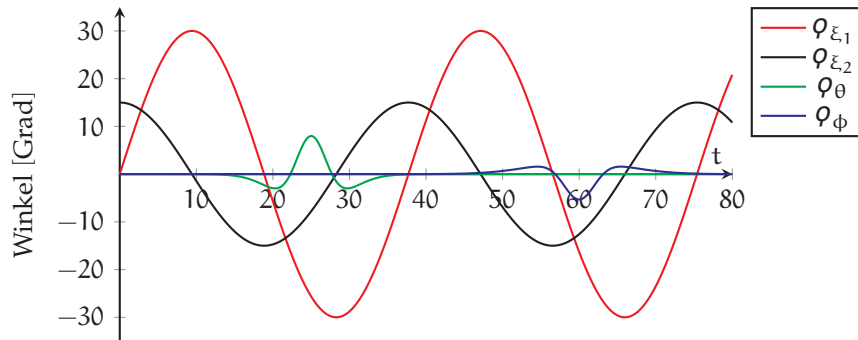


Abbildung 4.21: Funktionen zur Modellierung der kontinuierlichen Fahrzeugbewegung

Kurvenfahrt simuliert. Für Roll- und Nickwinkel wird eine abrupte Veränderung, wie durch ein Schlagloch bzw. eine Steigung verursacht, modelliert.

Die Funktionen werden äquidistant mit einer Rate von 25 Hz ( $\Delta_t = 25 \text{ Hz}^{-1}$ ) abgetastet. Für den vorgestellten Evaluationslauf wurden Daten über einen Zeitraum von 80 s bzw. 2 000 Frames aufgenommen. Um den Einfluss der Kalman-Filter-Initialisierung zu neutralisieren, wurden die ersten 3 Sekunden nicht in die Berechnung der Genauigkeiten einbezogen. Innerhalb dieser 75 Frames ist sichergestellt, dass sich der Kalman-Filter vollständig initialisiert hat (insbesondere Status und Vorhersage-Modell).

Für jeden Zeitschritt wird der geschätzte Fahrzeugzustand mit dem Konfigurationsvektor der Simulation aufgezeichnet und im Folgenden mit den Ergebnissen der in den Kapiteln 4.4 und 4.5 beschriebenen Verfahren verglichen.

In einem zusätzlichen Durchlauf wird die Robustheit der Verfahren gegenüber unpräziser Detektion der Markerecken geprüft. Dies wird durch Hinzufügen von zufälligem Rauschen auf die Koordinaten der Markereckpunkte im Bildraum erreicht. Der Zufallsvektor  $\eta \in [-1,5 \text{ px}; 1,5 \text{ px}] \times [-1,5 \text{ px}; 1,5 \text{ px}]$  beschreibt die Abweichung der Markereckpunkt-Koordinate und wird für jeden Eckpunkt neu erzeugt.

Die Ergebnisse sind in Abbildung 4.23 sowohl für den unverrauschten als auch für den verrauschten Durchlauf ersichtlich. Sie zeigen eine Überlegenheit des mit dem Kalman-Filter erweiterten Verfahrens gegenüber dem 2-D- und 3-D-Verfahren. Die Roll- und Nickzustände können in diesem Zusammenhang natürlich lediglich mit dem Kalman-erweiterten und dem ursprünglichen 3-D-Verfahren ermittelt werden.

Die genauen Abweichungen über die Messreihe hinweg sind in Abbildung 4.22 abgebildet. Dort sind die systematischen Fehler im 2-D-Verfahren erkennbar, die bei verletzter Annahme des planaren Untergrunds auftreten.

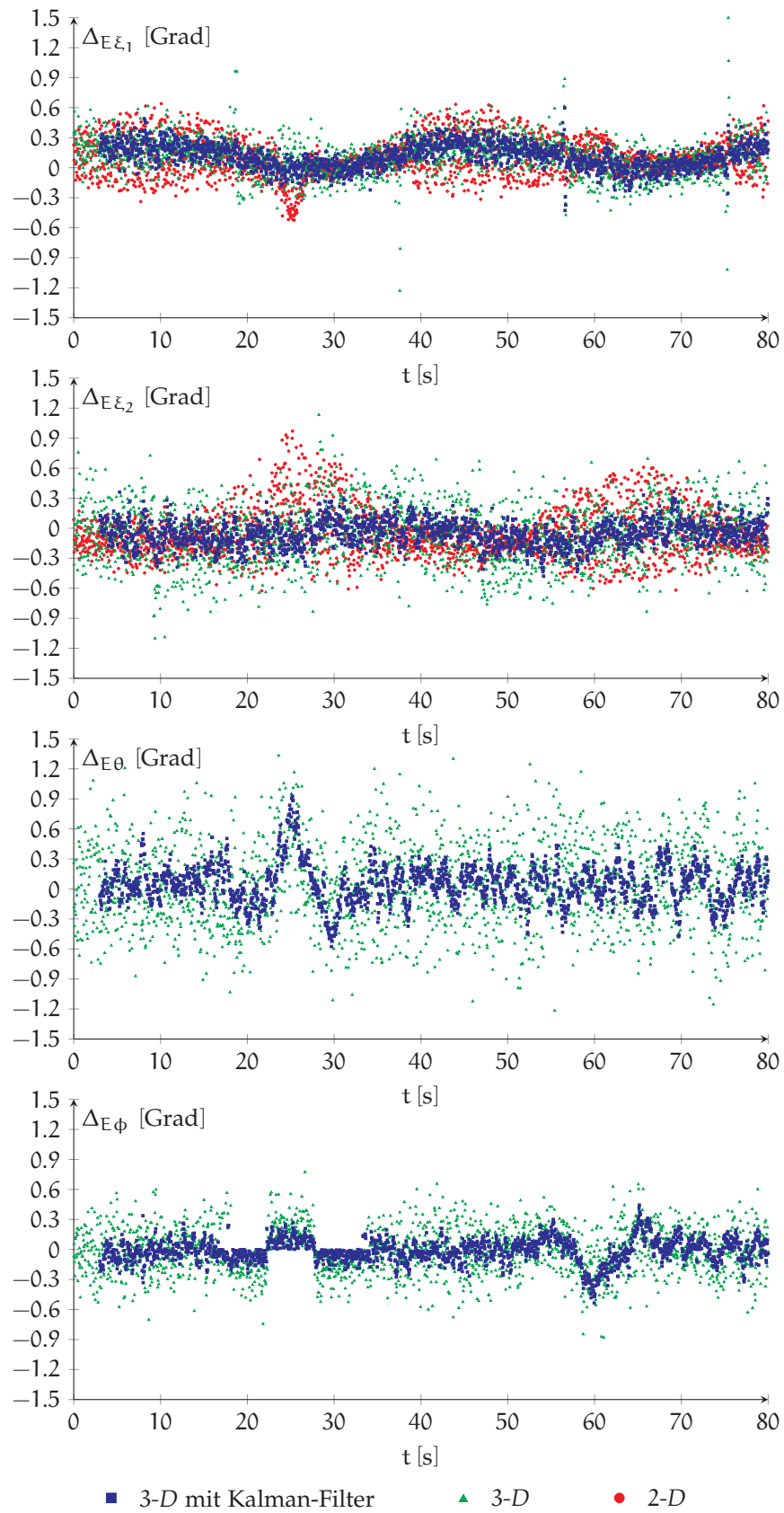


Abbildung 4.22: Fehler  $\Delta_E$  für die einzelnen Gespannstatuswinkel

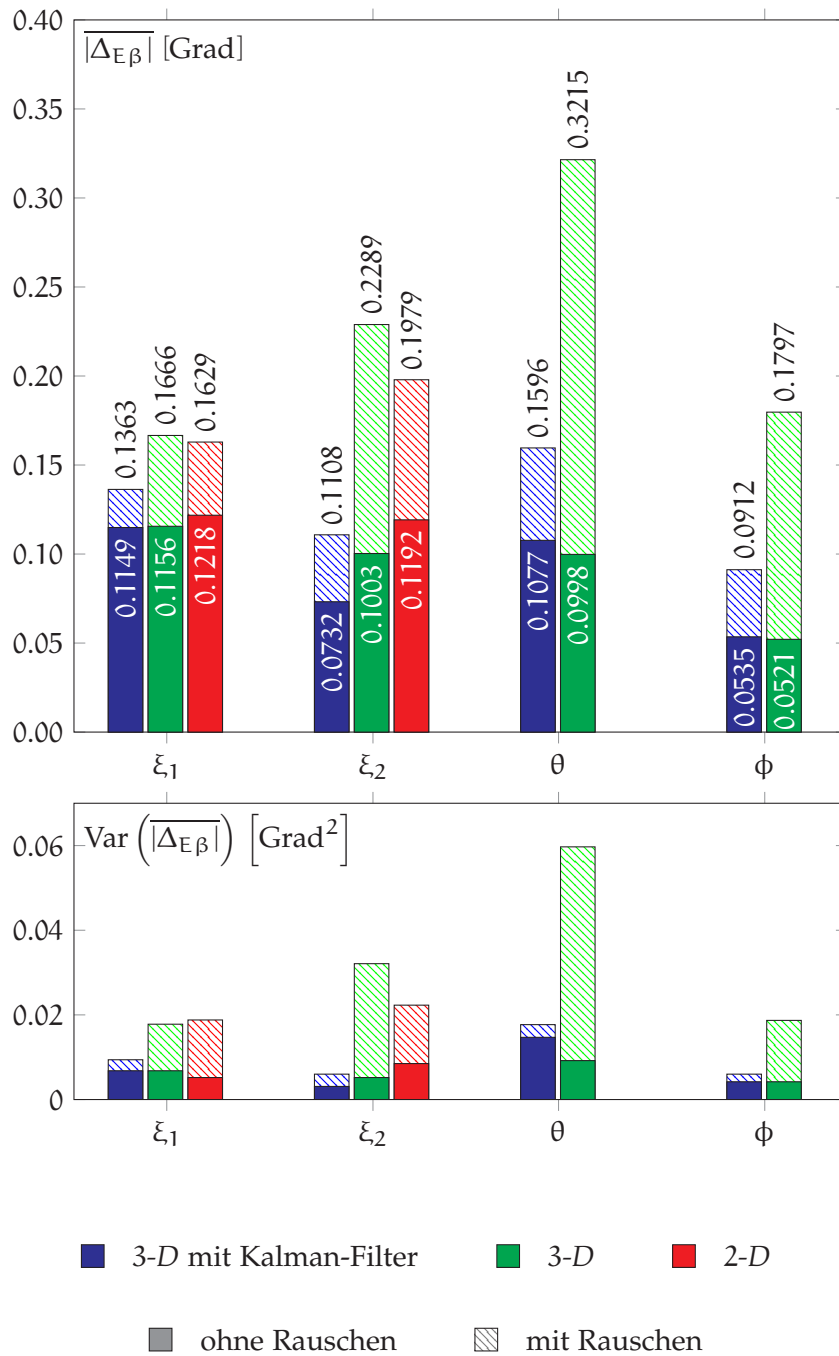


Abbildung 4.23: Ergebnisse für den mittleren absoluten Fehler  $|\overline{\Delta_{E\beta}}|$  und die Varianz des mittleren absoluten Fehlers  $\text{Var}(|\overline{\Delta_{E\beta}}|)$

Die Präzision des verbesserten 3-D-Verfahrens mit Kalman-Filter ist der des 3-D-Verfahrens und der des 2-D-Verfahrens beim Auftreten von Rauschen deutlich überlegen. Die Integration des Kalman-Filters und damit das Ausnutzen temporaler Integration bedeutet eine deutliche Verbesserung der Genauigkeiten.

## 4.7 AUSWIRKUNGEN AUF DIE TRAJEKTORIENQUALITÄT

In den vorangegangenen Kapiteln werden drei verschiedene Verfahren zur Ermittlung des Gespannzustands im Hinblick auf die Artikulation eines Fahrzeuggespanns vorgestellt. Die dort vorgestellten Auswertungen beziehen sich auf die Präzision des geschätzten Fahrzeugzustands.

Wünschenswert ist es, auch eine Aussage zur Auswirkung von potentiell unpräzisen Schätzungen auf die Bewegung des Fahrzeuggespanns treffen zu können. Die bereits vorgestellten Schätzverfahren für den Artikulationszustand eines Gespanns werden hierzu in eine *Software-in-the-Loop*-Umgebung integriert, die ein einfaches Fahrermodell beinhaltet, das sich strikt an generierte Fahrweisungen hält. Die im Folgenden dargestellten Ergebnisse wurden im Rahmen einer internationalen Konferenz vorgestellt [4].

Als Beispielszenario für die Auswirkung der Sensordatenqualität auf das Fahrscenario wurde das Heranfahen an eine zur initialen Fahrzeugausrichtung  $90^\circ$  ausgerichtete Laderampe gewählt. Ziel ist es, dass das Gespann vom Startpunkt eine Kreisbahn (genauer: einen Viertelkreis) fährt, sodass der Anhänger so vor der Laderampe zum Stehen kommt, dass die Ausrichtung der Anhängerrückwand parallel zur Laderampenkante ist. Das Fahrscenario ist in Abbildung 4.24 dargestellt.

4.7.1 *Gespannzustand und Zustandsschätzung*

Ein Anhänger bewegt sich in gewissen Freiheitsgeraden, wie in den vorangegangenen Kapiteln beschrieben. Für die vorgestellte Simulation wird die Einschränkung getroffen, dass sich das Gespann auf ebenem Grund bewegt, um den Parameterraum für die Auswertung beherrschbar zu halten. Eine Simulation für unebenen Grund bedingt eine komplexere Simulation der Fahrdynamik, die hier nicht im Fokus der Arbeit steht. Die Definition des Gespannzustands folgt der mit vier Freiheitsgraden (siehe Kapitel 4.5) [12]. Der verwendete geo-

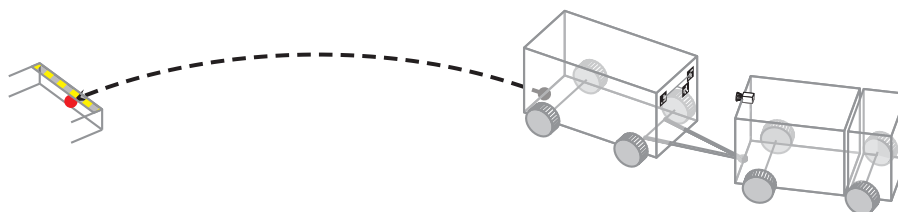


Abbildung 4.24: Anschauungsfall für die Evaluation der Trajektorienqualität: Das Gespann soll rückwärtsfahrend an eine Laderampe navigiert werden. Die ideale Trajektorie wird dabei durch einen Viertelkreis beschrieben.

metrische Aufbau entspricht dabei dem in Kapitel 4.3 dargestellten Aufbau.

Der Gespannzustandsvektor  $\mathbf{s} \in \mathbb{R}^4$  wird definiert als:

$$\mathbf{s} = (\xi_1, \xi_2, \theta, \phi)^T .$$

Der Gespannzustand  $\mathbf{s}$  kann sowohl für den *general-3-trailer* (Zweiachsanhänger mit Drehschemel) als auch für *general-2-trailer* (Einachsanhänger) verwendet werden. Zu erwarten ist, dass der gemessene Gierwinkel am Anhänger  $\xi_2 \approx 0$  ist für den Fall, dass ein *general-2-trailer* verwendet wird.

Für den Test werden die drei vorgestellten Verfahren gegeneinander getestet:

- 3-D mit Kalman-Filter (siehe Kapitel 4.6),
- 3-D (siehe Kapitel 4.5) und
- 2-D (siehe Kapitel 4.4).

#### 4.7.2 Fahrermodell-basierte Simulation

Die folgenden Abschnitte erläutern das Vorgehen hinsichtlich des *Software-in-the-Loop*-Systems und der Modellierung hinsichtlich des Fahrermodells.

##### *Komponenten der Testumgebung*

Um die Auswirkung der verschiedenen Ansätze zur Statusschätzung auf ein reales Szenario zu ermitteln, wird ein Fahrermodell genutzt, das die *Software-in-the-Loop*-Analyse mit einer Rückfahrasistenz (RAS) ermöglicht.

Die RAS implementiert eine Extrapolation des Fahrzeugzustands und nutzt Sensoren zur Statusschätzung des Gespanns als Eingabe, die (wie alle Messdaten) Unsicherheiten hinsichtlich der Genauigkeit der eingesetzten Sensorik/Methode beinhalten. Für diesen Fall bedeutet dies, dass die verrauschte Gespannstatus-Extrapolation die Eingabedaten für das Fahrermodell (siehe Kapitel 4.7.2) liefert. Die im Folgenden präsentierten Ergebnisse nutzen einen *general-2-trailer* (Einachsanhänger) als Konfiguration für das Gespann, um die Freiheitsgrade in den Steuerungsmodellen möglichst einfach zu halten. Selbstverständlich ist eine Erweiterung für den *general-3-trailer* möglich, wird jedoch an dieser Stelle nicht durchgeführt.

Um ein Fahrscenario inklusive der Bewegung/Trajektorie beschreiben zu können, werden neben dem Artikulationszustand weitere Parameter benötigt. Die hier vorgenommene Zustandsbeschreibung wird definiert durch:

- *Position und Orientierung des Fahrzeugs*  $\mathbf{q} = (q_x, q_y)^T, \rho$ ,  
Definiert als „Bicycle Model“ [Raj12], in dem u. a. 2-D-Position und Orientierung (als Ausrichtungswinkel auf der Ebene) enthalten sind
- *Artikulationszustand*  $\xi_1$ ,  
Aufgrund der Vereinfachung durch die Bewegung auf der Ebene und die Reduzierung auf den *general-2-trailer*-Trailer mit den Gierwinkeln  $\xi_1$  modelliert
- *Lenkwinkel des Zugfahrzeugs*  $\psi$  und  
Modellierung der Artikulation der Lenkachse (Richtungsänderung) ( $\psi$ )
- *Geschwindigkeit des Gespanns*  $v$ .

Diese Parameter definieren einen Konfigurationsvektor  $\zeta$ , der den Gespannzustand zum Zeitschritt  $\tau$  beschreibt:

$$\zeta_\tau = (q_x, q_y, \rho, \psi, \xi_1)^T .$$

Die Komponenten  $v$  (Geschwindigkeit) und  $\psi$  (Lenkwinkel) stellen die Einflussmöglichkeiten des Fahrers auf das Fahrzeug dar. Je nach Wahl und dem damit verbundenen zeitlichen Ablauf wird das Fahrzeug gesteuert. Die Schnittstelle bei realen Fahrern wird durch klassische Elemente wie Lenkrad bzw. Gaspedal und Bremse bedient. In der Simulationsumgebung werden diese Parameter auf konkrete übergebene numerische Werte abgebildet.

Der Konfigurationsvektor ist diskret definiert, d. h. er wird in einem zeitlichen Ablauf von Schritt zu Schritt bestimmt. Die Funktion  $\mathfrak{A}$  beschreibt den Übergang des Zeitschritts  $\tau$  zu  $\tau + 1$ :

$$\mathbf{u} = (v, \psi)^T, \quad \mathfrak{A}(\zeta_\tau, \mathbf{u}) = \zeta_{\tau+1} .$$

Sowohl die Aktuator- als auch die Sensordaten sind verrauscht und damit fehlerbehaftet. Daher ist es sinnvoll, den aktuellen Zustand nicht nur durch die vorangegangenen Werte und den aktuellen Steuerzustand  $\mathbf{u}$  zu bestimmen. Um ein konstantes Driften des Gespanns zu vermeiden wird der Gespannstatus regelmäßig während der Fahrt gemessen. Da der Gespannzustand unabhängig geschätzt wird, kann ein kumulativer Fehler vermieden werden. Der Effekt von Ungenauigkeiten im Bildabbildungsprozess bzw. der darauffolgenden Verarbeitung wird durch den entwickelten Testprozess untersucht.

### *Fahrermodell*

Fahrerszenarien können in sogenannte Manöver zerlegt werden. In diesem Kontext sind Manöver als Verknüpfung von Bewegungs-Primitiven, wie z. B. dem Fahren auf einem Kreisbogen oder dem



Fahren auf einer geraden Linie, zu verstehen. Manöver können verwendet werden, um Standard-Fahrsituationen, wie z. B. das Wenden oder das Rückwärtsfahren, zu beschreiben.

Für die Simulationsumgebung wurde eine Fahrer-Steuerung adaptiert und implementiert, die das Rückwärtsfahren an einen rückwärtigen und seitlich verschobenen Zielpunkt abbildet. Die Steuerung ahmt das typische Verhalten eines echten Fahrers nach und versucht den Zielpunkt in einem aufgestreckten Gespannzustand (d. h.  $\xi_1 = 0$ ) zu erreichen. Zu diesem Zweck nutzt der virtuelle Fahrer die Unterstützung einer Rückfahrassistentz und folgt deren Anweisungen.

#### *Fahrerassistenzsystem zum Rückwärtsfahren*

Die verwendete Rückfahrassistentz wurde an der Universität Koblenz-Landau entwickelt [BZ06, ZPW03] und wird für die vorgenommene Evaluation adaptiert.

Da die Entwicklung der Assistentz nicht im Rahmen der vorliegenden Arbeit erfolgt ist und diese als Software-Komponente angebunden ist, erfolgt die Darstellung im Folgenden als kurze Erläuterung zur Funktionsweise und den vorliegenden Schnittstellen. Auf die genaue Ausführung der Methoden und Algorithmen wird aus genanntem Grund an dieser Stelle verzichtet. Die zu erwartende Trajektorie wird auf Grundlage der Ausgabe des virtuellen Sensor- bzw. Schätzsystems und der Steuerdaten berechnet. Sie wird in eine Echtzeit-Kameraansicht in rückwärtiger Richtung des Anhängers eingeblendet. Die Ansicht zeigt dem Fahrer, wie sich das Fahrzeug verhält und lässt unmittelbare Steuerhinweise einfach ableiten.

Dabei werden dem Fahrer zwei unterschiedliche Trajektorien angezeigt: Die erste Trajektorie der beiden hinteren Anhängerräder wird extrapoliert im Hinblick auf den aktuellen Gespannzustand, wie er aus der letzten Iteration der Kinematiksimulation berechnet wurde. Die zweite Trajektorie visualisiert die Trajektorie des Achsenmittelpunkts der Anhängerachse unter Berücksichtigung des aktuellen Lenkwinkels. Solange sich die Trajektorie der Anhängerachse in mitten der Status-abhängigen Trajektorie der beiden Räder befindet, bewegt sich das Fahrzeug auf einer stabilen Kurve. Sobald sich das Fahrzeug aus der reaktiven Trajektorie bewegt, ändert sich der Kurvenradius. Der Grad der Auslenkung ist invers proportional zur Stabilität des artikulierten Fahrzeugs. Die optisch eingeblendeten Trajektorien unterstützen den Fahrer bei der Planung seiner Lenkbewegungen. Dabei kann der Fahrer die Trajektorien intuitiv kombinieren, um das Fahrzeug in stabiler Fahrt zum Zielpunkt zu navigieren. Da das Fahrerassistenzsystem auf einem fehlerbehafteten, geschätzten Gespannzustand beruht, der vom physikalisch korrekten Zustand abweicht, ist die Information des Fahrerassistenzsystems unvollkommen. Die dynamische Komponente des Statusvektors  $\zeta$  wird von verschiedenen Sensorsystemen aufgenommen, deren Rauschcharakteristika die

Gesamtqualität des Systems beeinflussen.

#### *Auswahl des Anschauungsfalls*

Um die Methoden zur Schätzung des Gespannzustands in einem Standard-Fahrszenario zu beurteilen, wurde ein Fahrermodell-basiertes Steuerungsmodul (*FSM*) als exemplarisches Anwendungsbeispiel für den Ansatz entwickelt.

Die Aufgabe des *FSM* besteht darin, das Fahrzeuggespann rückwärts zu einer Laderampe zu navigieren, deren Lage um  $90^\circ$  zur ursprünglichen Fahrzeugausrichtung gedreht ist (siehe Abbildung 4.24).

Die Komponente ist so parametrisiert, dass sie eine ähnliche Reaktionszeit wie ein menschlicher Fahrer abbildet (siehe Kapitel 4.7.3). Die relevanten Messgrößen zur Evaluation des Systems (Kontroll- und Statusvektoren) werden für jeden Iterationsschritt gespeichert. Das vorgestellte Standard-Fahrszenario wird bei gleicher Parametrierung jeweils für die drei Varianten der Gespannstatusschätzung in Kombination mit verschiedenen Stärken von zusätzlichem künstlichem Rauschen auf die geschätzten Status durchgeführt. In einer ersten Testreihe werden die Auswirkungen auf eine Verschiebung im Abbildungsprozess, wie z. B. durch starke Vibration innerhalb des Gespanns möglich, simuliert und untersucht.

#### 4.7.3 *Evaluation*

Die Komplexität der Evaluation und der damit verbundenen Interpretation der Ergebnisse steigt signifikant mit der Anzahl der Fahrzeugglieder im Gespann. Daher nutzt die durchgeführte Evaluation die Konfiguration eines *general-2-trailer* (Einachsanhänger), obwohl die genutzten Verfahren zur Statusschätzung des Gespanns für einen *general-3-trailer* entwickelt wurden. Vielmehr soll die Evaluation die methodische Herangehensweise für die Auswertung darstellen.

Um die Qualität einer im Fahrszenario gefahrenen Trajektorie auswerten zu können, muss das Fahrermodell (siehe Kapitel 4.7.2) mit der Kamera-Simulationsumgebung (siehe Kapitel 4.6) verbunden werden. Die Integration der beiden Komponenten bedient sich des *Software-in-the-Loop*-Ansatzes und erlaubt dadurch das einfache Konfigurieren der notwendigen Parameter und das Durchführen von Regressionstests. Die Fahrmanöver werden unter verschiedenen Aspekten bewertet:

- Abweichung zur optimalen Trajektorie,
- Notwendige Lenkkorrekturen bzw. -bewegungen,
- Genauigkeit hinsichtlich des Erreichens des Endpunkts.

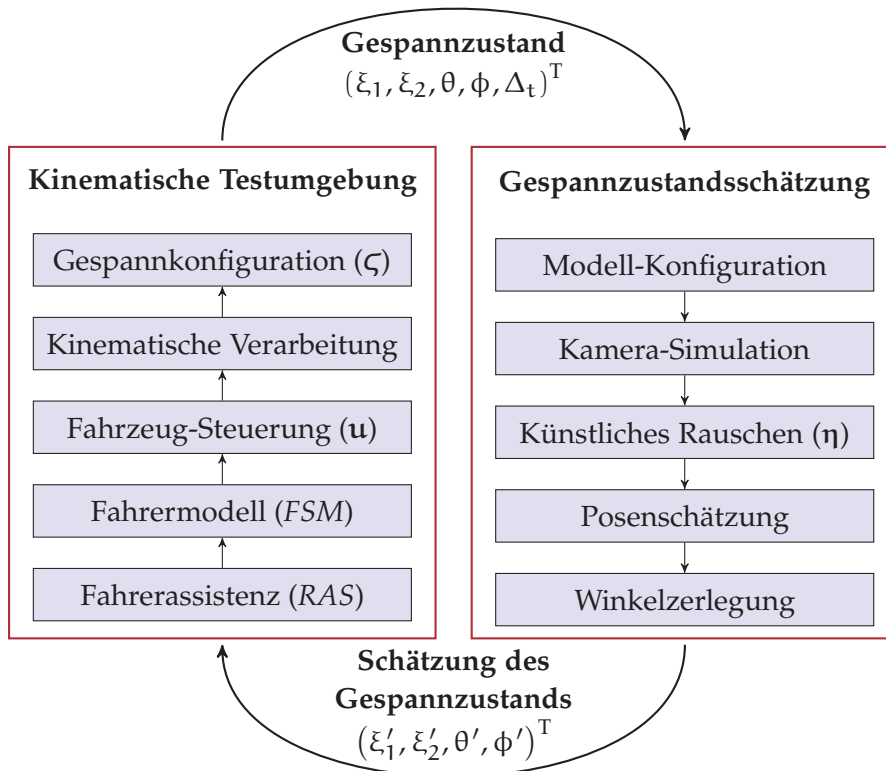


Abbildung 4.25: Datenfluss zwischen den Komponenten

Die kinematische Simulation berechnet die tatsächliche Gespannbe-  
 wegung und die fehlerbehaftete Modellierung hinsichtlich des Fahr-  
 rerassistenzsystems. Auf Basis der Ausgabe des Fahrerassistenzsys-  
 tems wird in jeder Iteration ein Steuerkommando ( $\mathbf{u}$ ) durch das Fahr-  
 ermodell *FSM* erzeugt. Die Reaktionsrate des Fahrers wird dabei mit  
 25 Hz angenommen. Ein Systemdiagramm des *Software-in-the-Loop*-  
 Tests ist in Abbildung 4.25 zu sehen.

Die geometrische Konfiguration des Fahrzeuggespanns ist analog  
 zu der in Kapitel 4.4.2 vorgestellten. Da in diesem Kontext ein *general-  
 2-trailer*-Gespann statt eines *general-3-trailer*-Gespanns genutzt wird,  
 ist der Drehschemel als feststehend anzunehmen ( $\xi_2 = 0$ ).

Die Kamerasimulation ist mit einer Auflösung von  
 1024 px  $\times$  768 px und einem horizontalen Kameraöffnungswin-  
 kel von 120° parametrisiert. Um die Robustheit der Verfahren zu  
 testen, wird ein künstliches Sensorrauschen definiert, das durch  
 künstliche Fehler in der Erkennung der Marker im Bild modelliert  
 wird. Das Rauschlevel  $\eta_1 \in \mathbb{R}^+$  definiert die Stärke eines additiven  
 Rauschens, das auf die Positionen der Markereckpunkte angewandt  
 wird. Das additive Rauschen simuliert sowohl Vibrationen des  
 Gespanns als auch unpräzise Detektionen der Marker. Für jeden Zeit-  
 schritt bzw. jede Iteration in der Gespanszustandsschätzung wird  
 ein Rauschvektor  $\boldsymbol{\eta} \in \mathbb{R}^2$  mit einer gleichverteilten beschränkten  
 Zufallsfunktion generiert:

$$\boldsymbol{\eta} = (\eta_x, \eta_y)^T \quad \eta_x, \eta_y \in [-\eta_l, +\eta_l] \quad .$$

Das Rauschen  $\boldsymbol{\eta}$  wird direkt auf die Bildkoordinaten der Markereckpunkte angewandt und beeinflusst damit die gesamte Schätzung des Gespannzustands, da die Markereckpunkte als Eingabe für die verschiedenen Verfahren dienen. Für das vorgestellte Szenario werden die folgenden Rauschlevel verwendet:

$$\eta_l \in \{0 \text{ px}, 5 \text{ px}, 10 \text{ px}, 15 \text{ px}, 20 \text{ px}, 25 \text{ px}, 30 \text{ px}\} \quad .$$

Um eine Aussage über die Qualität der unter dem Einfluss von Rauschen erreichten Trajektorie treffen zu können, muss eine Grundwahrheit herangezogen werden. Dies wird erreicht, indem initial das Fahrscenario mit perfektem Gespannzustand als Resultat der Schätzkomponente berechnet wird. Dazu wird eine direkte Weiterreichung (engl. *pass-through*) der Gespannkonfiguration realisiert, in der auch kein künstliches Rauschen enthalten ist.

Für jede Kombination aus Gespannstatusschätzung und Rauschlevel werden zehn Durchläufe berechnet, um eventuelle Ausreißer durch ungünstig aufeinanderfolgende bzw. extrem unterschiedliche Rauschvektoren zu eliminieren. In jedem Durchlauf wird zu jedem Zeitschritt die aktuelle Position (und damit die Trajektorie) und der Lenkwinkel  $\psi$  aufgezeichnet. Diese Datenmodalität bildet die Grundlage für die folgenden Auswertungen.

Die Trajektorien für das Rauschlevel  $\eta_l = 30 \text{ px}$  sind in Abbildung 4.26 zu sehen. Die metrische Distanz zwischen der Referenz-Trajektorie und der „gefahrenen“ Trajektorie wird durch die Kalman-basierte 3-D-Variante am besten minimiert.

Die metrische Distanz trifft eine wichtige Aussage über automatisierte Bewegungen (z. B. in automatisch-geführten bzw. autonomen Fahrzeugen), da sie die Distanz zwischen der Bewegungsplanung und der Bewegungsausführung quantisiert.

Das Bewegungsmodell des 3-D-Kalman-Ansatzes reduziert projektive Ungewissheiten durch das Ausnutzen temporaler Kohärenz in den Bildfolgen. Im Vergleich der simulierten Trajektorien zeigt sich dieses Verfahren als das robusteste unter den getesteten Methoden.

Die für den Vergleich mit der Grundwahrheit genutzte Distanzmetrik basiert auf der *Earth Mover's Distance (EMD)* und nutzt die diskret abgetasteten Trajektorien für die Berechnung (siehe Kapitel 4.2.3).

Die Bewertung der Trajektorien mit der *EMD* sind in Abbildung 4.27 zu sehen. Das Verfahren mit dem 3-D-Kalman zeigt die besten Eigenschaften hinsichtlich der *EMD*.

Ein weiterer wichtiger Aspekt für die Qualität der Trajektorien ist die Signifikanz der notwendigen Lenkkorrekturen, die benötigt werden, um dem geplanten Weg zu folgen. Abbildung 4.28 veranschaulicht die durch das Fahrermodell gesetzten Lenkwinkel während einer Fahrt bei einem Rauschlevel von  $\eta_l = \pm 15 \text{ px}$ . Die dargestellten

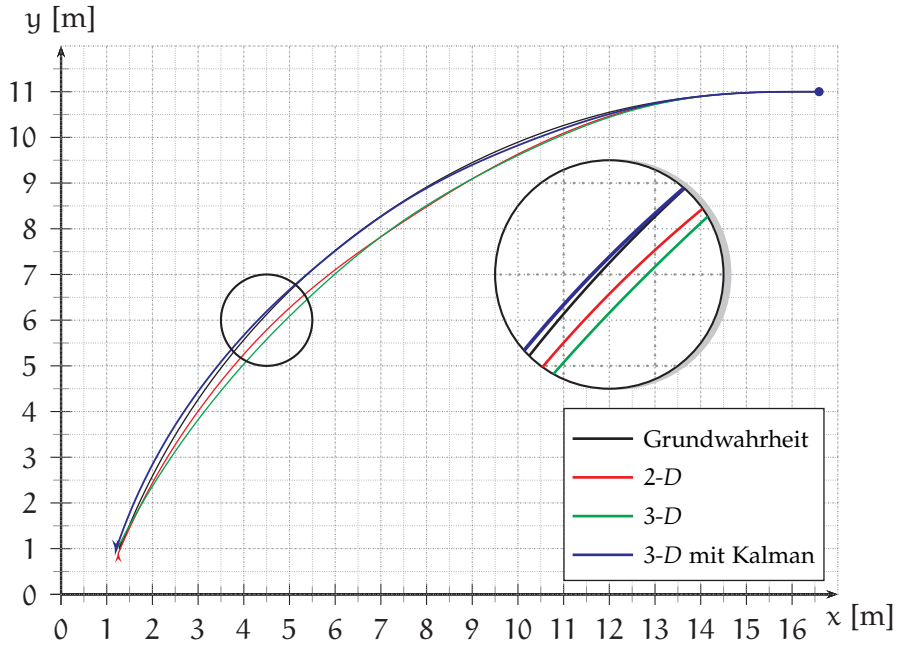


Abbildung 4.26: Trajektorien des Fahrzeugespanns für das Rauschlevel  $\eta_l = 30$  px. Der Kreis markiert eine zweifache Vergrößerung.

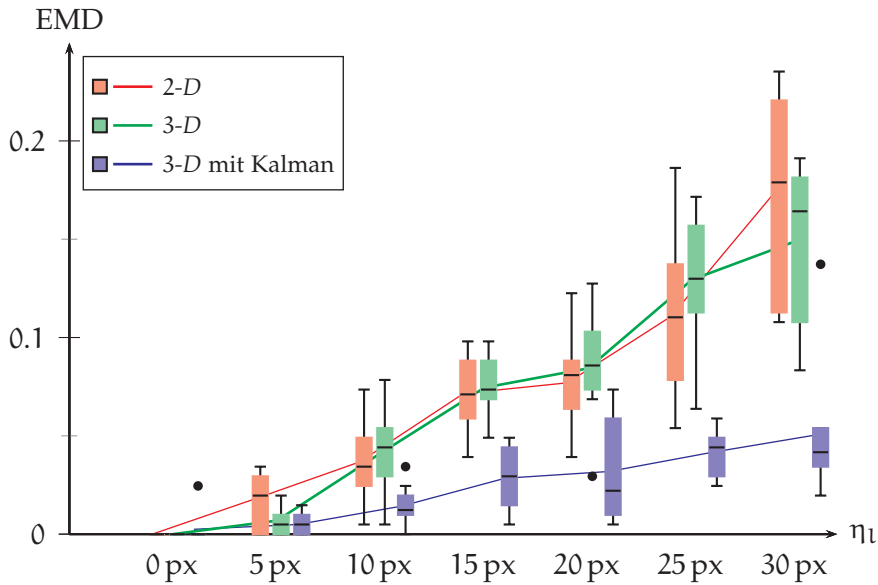


Abbildung 4.27: Vergleich der Trajektorienqualität mit der Grundwahrheit unter Verwendung der *EMD*. Für jede Parameterkombination wurden die Verfahren zehnmals simuliert. Die Ergebnisse sind mit Kastenplots dargestellt. Dabei kennzeichnen Punkte Ausreißer der jeweiligen Messreihe in den gemessenen Daten.

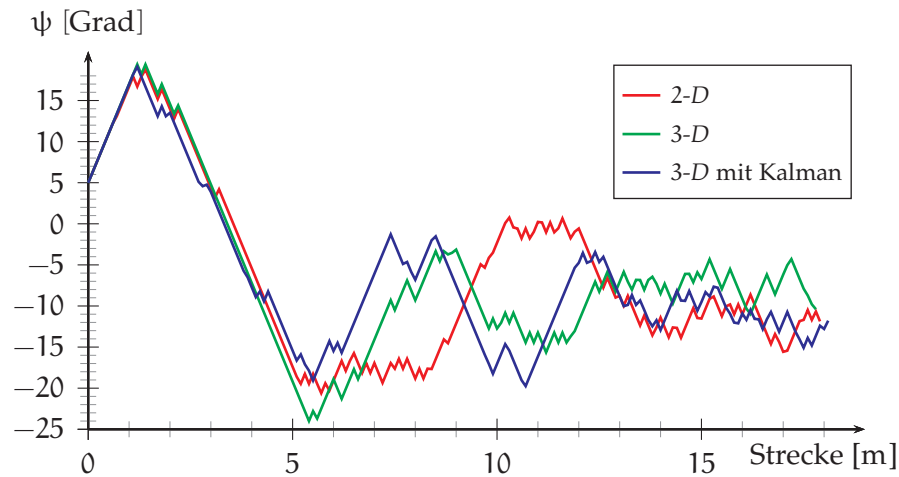
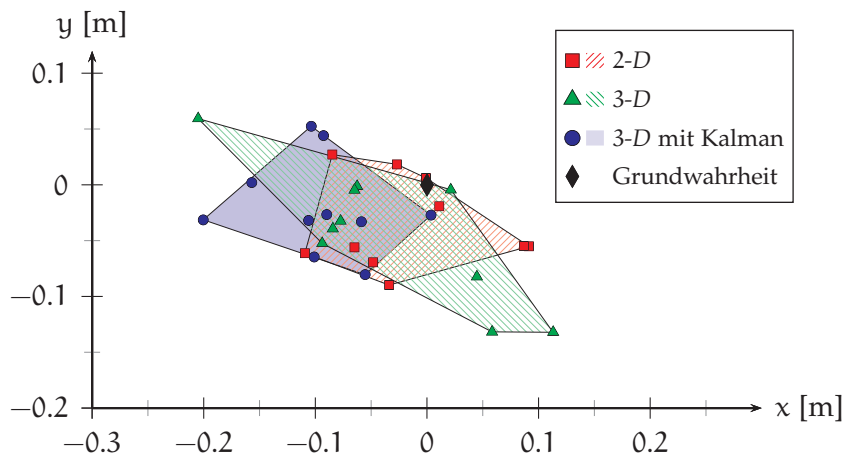


Abbildung 4.28: Lenkwinkel für ein Rauschlevel von  $\eta_l = 15 \text{ px}$  in Abhängigkeit zur zurückgelegten Strecke

Lenkwinkel sind repräsentativ zu den Ergebnissen anderer Durchläufe, auch mit anderen Rauschleveln.

Das für eine Fahrt angestrebte Verhalten hinsichtlich der Lenkbewegung unter dem Aspekt des sicheren Navigierens ist, dass eine minimale Korrekturrate notwendig ist. Dies ist insbesondere kurz vor dem angestrebten Zielpunkt wichtig. Die Visualisierung zeigt, dass unter Verwendung des Verfahrens mit dem 3-D-Kalman die benötigten Korrekturen minimal sind. Damit zeigt der 3-D-Kalman-Ansatz die wünschenswertesten Eigenschaften hinsichtlich der benötigten Lenkwinkelkorrektur.

Die Genauigkeit beim Erreichen des geplanten Endpunkts der Trajektorie ist ein weiteres Qualitätskriterium für das Fahrmanöver. Abbildung 4.29 zeigt eine metrische Darstellung der Endpunktverteilungen bei einem Rauschlevel von  $\eta_l = 30 \text{ px}$ . Das 2-D- und das Kalman-Filter-Verfahren zeigen vergleichbare Ergebnisse während der 3-D-Ansatz eine deutlich höhere Spreizung der Endpunkte aufweist. Die Visualisierung in Abbildung 4.29 zeigt, dass die Verteilung der Endpunkte nicht um den Endpunkt der Grundwahrheitstrajektorie verteilt ist wie evtl. zu erwarten wäre. Dieser Effekt ist durch die Implementierung der Steuerung zu erklären, die versucht, den finalen Soll-Endzustand so gut wie möglich zu erreichen. Ist die Ausrichtung des Anhängers am Endpunkt nicht hinreichend genau, erfolgt eine kleine Korrektur, bei der das Gespann ein wenig weiterbewegt wird, um den Gesamtzustand weiter zu optimieren.

Abbildung 4.29: Endpunktverteilung für das Rauschlevel  $\eta_l = 30\text{px}$





### 5.1 EINLEITUNG

Insbesondere bei großen Nutzfahrzeugen und Gespannen ist eine Unterstützung des Fahrers bzw. Bedieners bei der Umfeldbeobachtung ein wichtiges Thema, um die Betriebssicherheit von Großfahrzeugen zu erhöhen. Eine *korrekte* Darstellung der direkten Fahrzeugumgebung mit der Visualisierung von Hindernissen in der direkten Fahrzeugumgebung kann einen wichtigen Beitrag dazu leisten.

Im Folgenden wird ein System vorgestellt, das eine verbesserte Umgebungsvisualisierung und eine Hinderniserkennung aus Kameradaten fusioniert, um eine *perspektivisch korrekte* Ansicht aus der Vogelperspektive zu ermöglichen.

### 5.2 STAND DER TECHNIK

Die folgenden Unterkapitel beschäftigen sich mit für die Durchführung der Arbeit relevanten Aspekten und Bereichen.

#### 5.2.1 *Stereokameras und Stereoverarbeitung*

Die monokulare Datenaufnahme mit nur einer Kamera ermöglicht nur eine eingeschränkte Interpretation von räumlichen Informationen im Bild. Die Bestimmung von Größenverhältnissen und Tiefeninformationen aus einem einzelnen Bild sind nur im Rahmen von Referenzobjekten möglich. Durch die Hinzunahme einer zweiten Kamera wird es möglich, Tiefeninformationen zu den im Bild sichtbaren Objekten zu berechnen und somit weitergehende Geometrieinformationen zu erhalten. Ähnlich dem menschlichen Sehen werden zwei Bilder aus (leicht) unterschiedlichen Blickwinkeln genutzt, um eine verbesserte Interpretation der betrachteten Szene zu verwirklichen.

Gegeben zwei Kameras, die über ein ausreichend weit überlappendes Sichtfeld und eine rigide Positionierung zueinander verfügen, können die beiden Ansichten genutzt werden, um Disparitäten innerhalb des Bildpaares zu schätzen. Die dabei verwendete Epipolareometrie beschreibt die Zusammenhänge in der Ansicht zweier Kameras. Viele Publikationen beschäftigen sich mit dem Thema. Eine gute Übersicht geben unter anderen Zhang [Zha98] sowie Hartley und Zisserman [HZ03]. Die folgenden Absätze geben eine kurze Einführung über die wesentlichen Zusammenhänge. Die intrinsische Kalibrierung von einzelnen Kameras (siehe Kapitel 3.2) wird dabei

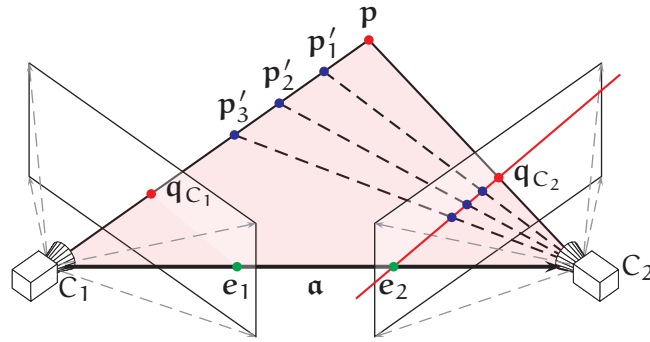


Abbildung 5.1: Grundlagen der Epipolargeometrie

vorausgesetzt bzw. miteinbezogen.

Das Grundprinzip ist in Abbildung 5.1 dargestellt: Kamera  $C_1$  betrachtet einen Punkt  $\mathbf{p}$ , der auf den Punkt  $\mathbf{q}_{C_1}$  in der Bildebene abgebildet wird. Die Punkte  $P = \{\mathbf{p}'_1, \mathbf{p}'_2, \dots, \mathbf{p}'_n\}$  liegen auf demselben Sichtstrahl von  $C_1$ . Dadurch werden sie ebenfalls auf den Punkt  $\mathbf{q}_{C_1}$  abgebildet. Es wird deutlich, dass die Tiefen- bzw. Abstandsinformation verloren geht.

Da Kamera  $C_2$  Punkt  $\mathbf{p}$  aus einer anderen Pose beobachtet, liegen die Punkte in  $P$  nicht auf demselben Sichtstrahl. Sie werden folglich auf andere Punkte auf der Bildebene abgebildet, die nicht der Projektion  $\mathbf{q}_{C_2}$  von  $\mathbf{p}$  entsprechen. Die Punkte in  $P$  liegen jedoch auf einer Geraden, die durch den Punkt  $\mathbf{p}$  und den *Epipol*  $\mathbf{e}_2$  verläuft. Diese Gerade wird als *Epipollinie* bezeichnet.

Die Kamerazentren von  $C_1$  und  $C_2$  sowie der Punkt  $\mathbf{p}$  spannen die sogenannte *Epipolare Ebene* auf, in der auch die Epipollinie liegt. Die Transformation von Kamera  $C_1$  zu Kamera  $C_2$  wird durch die Pose  $\mathbf{a} = \langle \mathbf{p}_{\mathbf{a}}, \mathbf{r}_{\mathbf{a}} \rangle$  beschrieben. In verschiedener Literatur wird die Länge der Translationskomponente  $\|\mathbf{p}_{\mathbf{a}}\|$  als *Baseline* bezeichnet. Wird nun die Projektion  $\mathbf{q}^{P_1}$  mit der Projektion  $\mathbf{q}^{P_2}$  bei bekannter Transformation  $\mathbf{a}$  in Verbindung gebracht, lässt sich die 3-D-Position des Punktes  $\mathbf{p}$  bestimmen.

Beschreibe  $\mathbf{p}^{C_1}$  die Richtung des Sichtstrahls zum Schnittpunkt im Koordinatensystem von  $C_1$  und  $\mathbf{p}^{C_2}$  den Sichtstrahl zu Punkt  $\mathbf{p}$  im Koordinatensystem von  $C_2$ . Die Vektoren  $\mathbf{p}^{C_1}$  und  $\mathbf{p}^{C_2}$  sind jeweils normiert. Da die Transformation von  $C_1$  in  $C_2$  durch die Pose  $\mathbf{a} = \langle \mathbf{p}_{\mathbf{a}}, \mathbf{r}_{\mathbf{a}} \rangle$  gegeben ist, gilt:

$$\begin{aligned} \mathbf{p}^{C_2} &= \mathbf{T}(\mathbf{a}) \cdot \mathbf{p}^{C_1} = \text{rot}(\mathbf{r}_{\mathbf{a}}) \cdot \mathbf{p}^{C_1} + \mathbf{p}_{\mathbf{a}} \\ \Leftrightarrow \text{rot}(\mathbf{r}_{\mathbf{a}})^T \cdot \mathbf{p}^{C_2} &= \mathbf{p}^{C_1} + \text{rot}(\mathbf{r}_{\mathbf{a}})^T \cdot \mathbf{p}_{\mathbf{a}} \end{aligned}$$

Durch die Epipolare Ebene ergibt sich der Zusammenhang:

$$\left( \mathbf{p}^{C_1} + \text{rot}(\mathbf{r}_{\mathbf{a}})^T \cdot \mathbf{p}_{\mathbf{a}} \right)^T \left( \left( \text{rot}(\mathbf{r}_{\mathbf{a}})^T \cdot \mathbf{p}_{\mathbf{a}} \right) \times \mathbf{p}^{C_1} \right) = 0 \quad .$$

Die Kombination dieser beiden Gleichungen ergibt die Bedingung:

$$\begin{aligned}
& \left( \text{rot}(\mathbf{r}_a)^T \cdot \mathbf{p}^{c_2} \right)^T \left( \left( \text{rot}(\mathbf{r}_a)^T \cdot \mathbf{p}_a \right) \times \mathbf{p}^{c_1} \right) = 0 \\
\Leftrightarrow & \mathbf{p}^{c_2 T} \cdot \text{rot}(\mathbf{r}_a) \left( \left( \text{rot}(\mathbf{r}_a)^T \cdot \mathbf{p}_a \right) \times \mathbf{p}^{c_1} \right) = 0 \\
\Leftrightarrow & \mathbf{p}^{c_2 T} \cdot \text{rot}(\mathbf{r}_a) \left[ \text{rot}(\mathbf{r}_a)^T \cdot \mathbf{p}_a \right]_{\times} \cdot \mathbf{p}^{c_1} = 0 \\
\Leftrightarrow & \mathbf{p}^{c_2 T} \cdot \underbrace{[\mathbf{p}_a]_{\times} \cdot \text{rot}(\mathbf{r}_a)}_{\mathcal{E}} \cdot \mathbf{p}^{c_1} = 0 \quad .
\end{aligned}$$

Die Matrix  $\mathcal{E}$  wird als *Essentielle Matrix* bezeichnet. Der Term  $\mathcal{E} \cdot \mathbf{p}^{c_1}$  in der Gleichung beschreibt dabei die Epipollinie für  $\mathbf{p}^{c_1}$  für die Kamera  $C_2$ .

Mit einer (einfachen) perspektivischen Projektionsmatrix und der Annahme das  $F = 1$  (d.h. die Projektionsmatrix ist eine Einheitsmatrix) ergibt sich mit  $\tilde{\mathbf{q}}^{i_1} = \mathcal{P}(\mathbf{p}^{c_1})$  und  $\tilde{\mathbf{q}}^{i_2} = \mathcal{P}(\mathbf{p}^{c_2})$  die *Epipolare Bedingung* nach Longuet-Higgins [LH81] für normalisierte Koordinaten:

$$(\tilde{\mathbf{q}}^{i_2})^T \cdot \mathcal{E} \cdot \tilde{\mathbf{q}}^{i_1} = 0 \quad .$$

Die Projektion beruht hierbei auf idealen Bildkoordinaten. Die Intrinsik einer echten Kamera mit Diskretisierung durch das Pixelraster wird mit den Kameramatrizen  $\mathbf{K}$  beschrieben:

$$\begin{aligned}
\tilde{\mathbf{q}}^{p_1} &= \mathbf{K}_{C_1} \tilde{\mathbf{q}}^{i_1} & \tilde{\mathbf{q}}^{p_2} &= \mathbf{K}_{C_2} \tilde{\mathbf{q}}^{i_2} \\
\Leftrightarrow \tilde{\mathbf{q}}^{i_1} &= \mathbf{K}_{C_1}^{-1} \tilde{\mathbf{q}}^{p_1} & \Leftrightarrow \tilde{\mathbf{q}}^{i_2} &= \mathbf{K}_{C_2}^{-1} \tilde{\mathbf{q}}^{p_2}
\end{aligned}$$

Zusammen ergibt dies:

$$(\tilde{\mathbf{q}}^{p_2})^T \cdot \underbrace{\left( \left( \mathbf{K}_{C_2}^{-1} \right)^T \cdot \mathcal{E} \cdot \mathbf{K}_{C_1}^{-1} \right)}_{\mathcal{F}} \cdot \tilde{\mathbf{q}}^{p_1} = 0 \quad .$$

Die *Fundamentalmatrix*  $\mathcal{F} \in \mathbb{P}^{2 \times 2}$  beschreibt die Beziehung zwischen den beiden Projektionspunkten  $\mathbf{q}^{p_1}$  und  $\mathbf{q}^{p_2}$ . Sie fusioniert die Intrinsik-Informationen der jeweiligen Kamera mit dem Wissen über die Relativpose  $\mathbf{a}$  und die Epipolare Ebene.

Die Pose  $\mathbf{a} = \langle \mathbf{p}_a, \mathbf{r}_a \rangle$  zwischen den Kameras  $C_1$  und  $C_2$  als Parameter wird durch Kalibrierverfahren bestimmt. Ähnlich der intrinsischen Kalibrierung (siehe Kapitel 3.2) werden vorzugsweise Kalibrierobjekte genutzt, die leicht im Bild zu detektieren sind. Ist das Kalibrierobjekt in beiden Kamerabildern identifiziert, werden Punkt-korrespondenzen genutzt, um die Überführung zwischen den Kamerabildern zu berechnen. Ein prominentes Verfahren zur Ermittlung der Fundamentalen Matrix  $\mathcal{F}$  bzw. der Essentiellen Matrix  $\mathcal{E}$  ist der 8-Punkte-Algorithmus in der Variante nach Hartley [Har97].

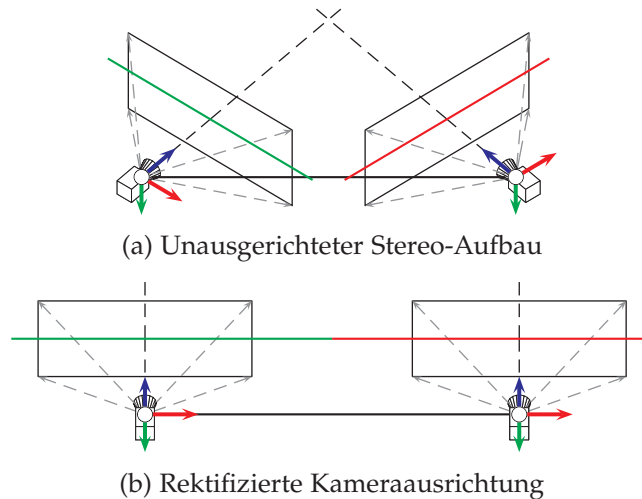


Abbildung 5.2: Ausrichtung der Kamerabilder durch Rektifizierung

### Rektifizierung

Ist die Kalibrierung der Stereokameras bekannt, werden die Bildpaare üblicherweise zunächst in Vorbereitung auf die kommenden Verarbeitungsschritte transformiert. Bei der Rektifizierung ist es das Ziel, die Bildebenen der beiden Kameras auf eine Ebene zu transformieren, so dass die optischen Achsen der beiden Kameraansichten parallel sind und die Epipollinien parallel verlaufen, d. h. einer Richtung des Pixelrasters entsprechen. Dadurch wird die Suche nach Korrespondenzen eines Bildpaares erheblich vereinfacht, da die notwendige Suchrichtung auf eine Achsenrichtung normiert wird.

Die Rektifizierung der Bilder kann mittels Homographien (siehe Kapitel 3.3) beschrieben werden. Voraussetzung für die Bestimmung der Homographien sind bekannte intrinsische Kamerakalibrierungen und bekannte Transformationen zwischen den Koordinatensystemen der Kameras (Fundamentale/Essentielle Matrix bzw. Pose zwischen den Kameras).

Das Prinzip der Vorverarbeitung für den Stereo-Matching-Prozess wird in Abbildung 5.3 dargestellt. Nach der Bildaufnahme werden die einzelnen Kamerabilder zunächst entzerrt (siehe Kapitel 3.2). Danach folgt die Rektifizierung, in deren Ergebnis die Korrespondenzen entlang einer Achsenrichtung liegen (in der Abbildung die x-Achse). In Abbildung 5.2 wird das Ausgangs- und Zielszenario für die Ausrichtung der Kamerabilder visualisiert.

Das Vorgehen zur Ermittlung der Homographien für die Rektifizierung wird u. a. von Loop und Zhang [LZ99] beschrieben: Ziel ist es, die Epipole  $e_1$  und  $e_2$  auf einen vorbestimmten Punkt abzubilden. Da die Sichtachsen nach der Rektifizierung parallel sein sollen, liegt dieser Punkt im Unendlichen. Mittels homogener Koordinaten lässt sich dieser Punkt darstellen als  $\tilde{e}_\infty = (1,0,0)^T$ . Aus der epipolaren Be-

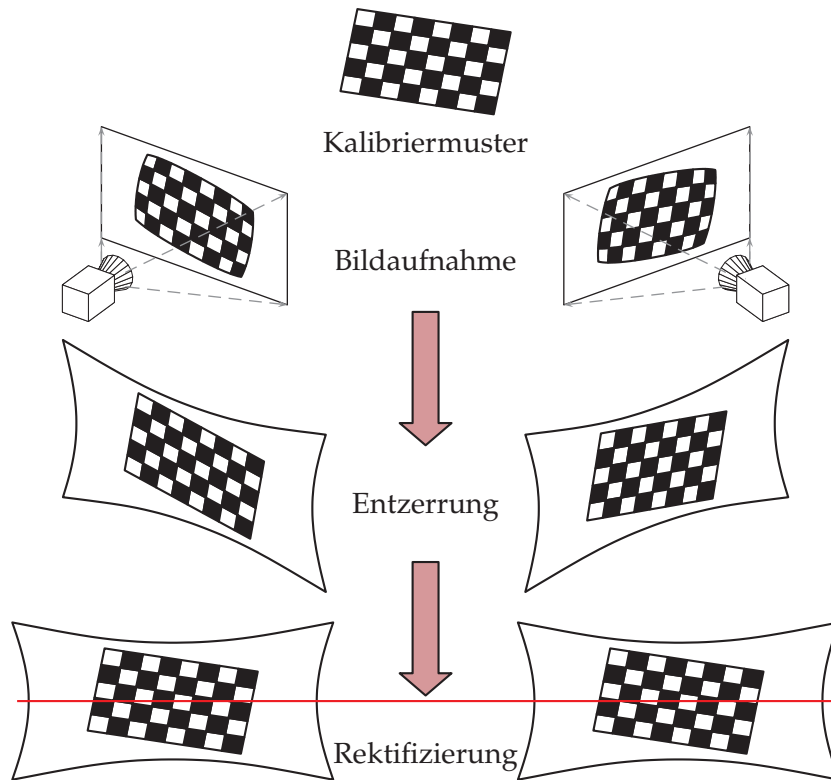


Abbildung 5.3: Schematische Darstellung des Rektifizierungsprozesses

dingung ergibt sich für die Fundamentalmatrix  $\mathcal{F}'$  der rektifizierten Bilder:

$$\mathcal{F}' = [\tilde{\mathbf{e}}_\infty]_\times = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix} .$$

Seien nun  $\mathbf{H}_{C_1}$  und  $\mathbf{H}_{C_2}$  die Homographie-Matrizen für die Transformation der jeweiligen Kamerabilder und  $\mathbf{q}_1$  und  $\mathbf{q}_2$  zwei korrespondierende Punkte zwischen den Kamerabildern. Die Punkte  $\mathbf{q}'_1$  und  $\mathbf{q}'_2$  seien die beiden rektifizierten Korrespondenzen:

$$\tilde{\mathbf{q}}'_1 = \mathbf{H}_{C_1} \cdot \tilde{\mathbf{q}}_1 \quad \tilde{\mathbf{q}}'_2 = \mathbf{H}_{C_2} \cdot \tilde{\mathbf{q}}_2 .$$

Es folgt für die transformierten Punktkorrespondenzen:

$$\begin{aligned} (\tilde{\mathbf{q}}'_2)^T \cdot \mathcal{F}' \cdot \tilde{\mathbf{q}}'_1 &= 0 \\ \Leftrightarrow (\tilde{\mathbf{q}}_2)^T \cdot \underbrace{\mathbf{H}_{C_2}^T \cdot \mathcal{F}' \cdot \mathbf{H}_{C_1}}_{\mathcal{F}} \cdot \tilde{\mathbf{q}}_1 &= 0 . \end{aligned}$$

Es ergibt sich als Faktorisierung der Fundamentalmatrix  $\mathcal{F}$ :

$$\mathcal{F} = \mathbf{H}_{C_2}^T \cdot [\tilde{\mathbf{e}}_\infty]_\times \cdot \mathbf{H}_{C_1} .$$

Diese Homographien sind nicht eindeutig bestimmbar. Mit der Lösung eines Optimierungsproblems, das in der Publikation von

Loop und Zhang [LZ99] ausführlich beschrieben ist, lassen sich die Homographie-Matrizen so bestimmen, dass die auftretende Verzerrung minimal wird.

Ist die mittels Stereokalibrierung ermittelte Pose  $\mathbf{a} = \langle \mathbf{p}_\mathbf{a}, \mathbf{r}_\mathbf{a} \rangle$  zwischen den Kameras gegeben, kann die Rektifizierung einfacher berechnet werden. Das Ziel, die Epipole ins Unendliche zu transformieren, bleibt bestehen. Beschreibe der Vektor  $\mathbf{a}_1 \in \mathbb{R}^3$  einen Einheitsvektor zwischen den beiden Kameras. Dieser kann durch die Translation  $\mathbf{p}_\mathbf{a}$  beschrieben werden:

$$\mathbf{a}_1 = \frac{\mathbf{p}_\mathbf{a}}{\|\mathbf{p}_\mathbf{a}\|} .$$

Das Kreuzprodukt des Vektors  $\mathbf{a}_1$  mit der Kamerablickrichtung (definiert in z-Richtung) ergibt einen zu  $\mathbf{a}_1$  orthogonalen Vektor, bezeichnet als  $\mathbf{a}_2 \in \mathbb{R}^3$ :

$$\mathbf{a}_2 = \mathbf{a}_1 \times (0, 0, 1)^T .$$

Ein dritter Vektor  $\mathbf{a}_3 \in \mathbb{R}^3$ , der orthogonal zu  $\mathbf{a}_1$  und  $\mathbf{a}_2$  ist, kann trivial mit dem Kreuzprodukt bestimmt werden:

$$\mathbf{a}_3 = \mathbf{a}_1 \times \mathbf{a}_2 .$$

Diese drei Hilfsvektoren können zu einer Rotationsmatrix  $\mathbf{R}_{\text{rect}}$  kombiniert werden, die die für die Rektifizierung notwendige Transformation beschreibt:

$$\mathbf{R}_{\text{rect}} = \begin{pmatrix} \mathbf{a}_1^T \\ \mathbf{a}_2^T \\ \mathbf{a}_3^T \end{pmatrix} .$$

Die Rotation  $\mathbf{R}_{\text{rect}}$  wird genutzt, um die beiden Kamerabilder zu rektifizieren. Für einen Bildpunkt  $\mathbf{q}^P = (q_x, q_y)^T$  gilt dann für die Berechnung des rektifizierten Bildpunktes  $\mathbf{q}_{\text{rect}}^P$ :

$$\begin{aligned} \mathbf{q}^i &= \mathbf{q}^P - (c_x, c_y)^T \\ \tilde{\mathbf{q}}' &= \mathbf{A} \cdot \mathbf{R}_{\text{rect}} \cdot \tilde{\mathbf{q}}^i \\ \mathbf{A} &= \begin{cases} \mathbf{Id} & \text{für die Stereo-Hauptkamera} \\ \text{rot}(\mathbf{r}_\mathbf{a}) & \text{für die zweite Kamera} \end{cases} \\ \mathbf{q}_{\text{rect}}^P &= (F_x \cdot q'_x, F_y \cdot q'_y)^T + (c_x, c_y)^T . \end{aligned}$$

### *Synchronisierte Bilder*

Bei statischen Szenarien ist es zunächst denkbar, dass eine Szene mit derselben Kamera sequentiell aus zwei Blickwinkeln aufgenommen wird, um darin Stereo-Korrespondenzen zu berechnen. Für die Aufnahme von dynamischen Szenen ist es wichtig, dass die Bildpaare zu

synchronisierten Zeitpunkten aufgenommen werden. Durch zentrale Auslöse-Mechanismen (engl. *trigger*) ist es möglich, ein Bildpaar mit beiden Kameras zu exakt gleicher Zeit aufzunehmen. Technisch gibt es bei Kameras hierfür spezielle Kabelbelegungen, die bei entsprechender Schaltung die Synchronisation vornehmen. Bei den in dieser Arbeit genutzten Bildaufnahmen handelt es sich durchweg um entsprechend synchronisierte Daten.

### *Stereorekonstruktion*

Um Tiefeninformationen aus Bildpaaren zu rekonstruieren, ist es erforderlich, korrespondierende Bildelemente zu identifizieren, die dann mit den vorgestellten Methoden in Tiefeninformationen überführt werden können. Dabei wird zwischen zwei prinzipiellen Vorgehensweisen unterschieden: Merkmals-basierte und Block-basierte Stereorekonstruktion. Das Problem des Findens von korrekten Korrespondenzen wird in der Literatur oft als *Stereo-Matching*- oder *Matching*-Problem bezeichnet.

### *Merkmals-basierte Stereorekonstruktion*

Eine etablierte Methode zum Wiederfinden von Bildpunkten in anderen Ansichten ist die Verwendung von Merkmalsdeskriptoren. Ziel ist es, zunächst markante Punkte in den beiden Bildern zu berechnen und dann die Merkmalsdeskriptoren abzugleichen.

Verfahren wie „Speeded up Robust Features“ (*SURF*) nach Bay, Tuytelaars und Gool [BTGo6], „Scale-Invariant Feature Transform“ (*SIFT*) nach Lowe [Low04], „Oriented FAST and Rotated Brief“ (*ORB*) nach Rublee et al. [RRKB11] oder „Binary Robust Invariant Scalable Keypoints“ (*BRISK*) nach Leutenegger, Chli und Siegwart [LCS11] seien an dieser Stelle als Beispiele für geeignete Merkmalsdeskriptoren genannt. Prinzipiell kann jeder Merkmalsdeskriptor eingesetzt werden, der für das Wiederfinden in einem anderen Bild bzw. einer anderen Ansicht geeignet ist.

Gefundene Paare werden verwendet, um eine Schätzung der Tiefe zu berechnen. Der Suchraum für korrespondierende Merkmale kann dabei entsprechend der Epipolaren Bedingung eingeschränkt werden.

Grimson [Gri85] verwendet Bildmerkmale, um Stereo-Korrespondenzen zu ermitteln. Horaud und Skordas [HS89] gruppieren die Merkmale zunächst um darauf aufbauend Stereo-Korrespondenzen zu finden. Ansätze, die auf einzelnen Merkmalskorrespondenzen beruhen, überzeugen durch die Präzision der geschätzten Tiefeninformationen. Nachteil dieser Verfahren sind durchweg nur dünne (engl. *sparse*) 3-D-Daten.

*Block-basierte Stereorekonstruktion*

Die zweite große Gruppe von Verfahren für den Stereoabgleich sind die Block-basierten Verfahren (engl. *blockmatching*). Dabei werden Makro-Pixelblöcke aus dem einen entlang der korrespondierenden Epipollinie im anderen Bild verglichen. Über entsprechende Abstandsmetriken werden Kandidaten gefunden. Eine einfache und effiziente Abstandsmetrik ist beispielsweise die Summe der absoluten Differenzen im Fall von Grauwertbildern. Ist ein hinreichend und maximal ähnlicher Block gefunden, so wird dieser Bereich als Korrespondenz interpretiert. Unter anderem gehören Blockgröße und minimale und maximale Abstände für die Suche zu den Parametern der Block-basierten Verfahren.

Hirschmüller [Hiro8] realisiert ein semi-globales Matching und nutzt dabei ein Pixelweises Matching von Transinformationen bei gleichzeitiger Approximierung eines globalen Glättekrateriums. Das 2-D-Glättekraterium (engl. *smoothness constraint*) wird dabei durch die Kombination vieler 1-D-Kriterien gebildet.

Der Algorithmus nach Hirschmüller wird in vielen Szenarien eingesetzt und wurde bereits in populäre Open-Source-Bibliotheken integriert [KB16]. Der Algorithmus wurde in einigen Publikationen erweitert oder für spezielle Szenarien angepasst: Pantilie und Nedevski [PN12] stellen eine optimierte, echtzeitfähige Variante des semi-globalen Matchings vor. Hermann und Klette [HK13] setzten das semi-globale Matching iterativ ein, um robuste Fahrerassistenzsysteme zu ermöglichen. Einecke und Eggert [EE14] nutzen einen lokalen Korrespondenzansatz, um die Ausführungszeit des semi-globalen Blockmatchings bei gleichbleibender Qualität signifikant zu reduzieren.

Vorteil der Block-basierten Rekonstruktion ist eine dichte Tiefenkarte, die Tiefendaten zu nahezu allen Bildpixeln ermöglicht. Bei wechselseitigen Verdeckungen ist selbstverständlich keine Disparitätenschätzung möglich. Die Präzision der Merkmals-basierten Ansätze (siehe Kapitel 5.2.1) ist jedoch überlegen, da Merkmalspunkte präziser lokalisiert werden können. Folglich sind auch die geschätzten Tiefeninformationen bei Merkmals-basierten Ansätzen gewöhnlich von höherer Präzision.

Die Optimierungen der einzelnen Verfahren werden im Folgenden nicht näher erläutert und können den jeweiligen Veröffentlichungen entnommen werden. Im Rahmen dieser Arbeit wird der Algorithmus nach Hirschmüller [Hiro8] eingesetzt.

*Disparitäten*

Die Stereo-Matching-Algorithmen erzeugen Disparitätenkarten in Referenz zur Stereo-Hauptkamera. In den Disparitätenkarten wird quantifiziert, wie groß der örtliche Versatz eines Punktes in Verhält-



nis zu dem im anderen Bild als Korrespondenz identifizierten Punkt ist. Bei Verwendung von rektifizierten Bildern entspricht die Disparität dabei dem Versatz eines Merkmals in Pixeln entlang der Achsen-Suchrichtung (im in Abbildung 5.3 abgebildeten Fall in horizontaler  $x$ -Richtung). Die Disparitätenwerte sind umgekehrt proportional zum damit korrespondierenden Abstand zur Kamera.

Mit Kenntnis der intrinsischen und extrinsischen Kalibrierung des Stereo-Systems und den Rektifizierungsdaten können die Disparitäten in Tiefenwerte und damit in 3- $D$ -Punkte umgerechnet werden. Die dazu notwendige Transformation kann aus diesen Werten abgeleitet werden. Die dazu notwendige Matrix  $\mathbf{Q} \in \mathbb{P}^{3 \times 3}$  transformiert einen Punkt  $\mathbf{q}^p$  aus dem Bild der Stereo-Hauptkamera zusammen mit der an der entsprechenden Bildposition ermittelten Disparität  $\lambda \in \mathbb{N}$ :

$$\mathbf{Q} = \begin{pmatrix} 1 & 0 & 0 & -c_x \\ 0 & 1 & 0 & -c_y \\ 0 & 0 & 0 & F \\ 0 & 0 & -\frac{1}{p_x} & \frac{c_x - c'_x}{p_x} \end{pmatrix}$$

$$\tilde{\mathbf{p}}^c = \mathbf{Q} \cdot (q_x^p, q_y^p, \lambda, 1)^T$$

$$= \left( q_x^p - c_x, q_y^p - c_y, F, -\frac{\lambda}{p_x} + \frac{c_x - c'_x}{p_x} \right)^T .$$

Der Term  $c_x - c'_x$  beschreibt dabei die Differenz der  $x$ -Komponente des Kamerahauptpunkts zwischen den beiden Kameras. Dieser korrigiert die dadurch bedingte zusätzliche Translation. In vielen Fällen ist diese jedoch 0.

### *Tiefeninformationen mittels Maschinellern*

Techniken des Maschinellen Lernens finden auch im Bereich des Stereo-Matchings bzw. dem Generieren von Tiefendaten aus Stereo-Bildern ihren Einsatz.

Žbontar und LeCun [ZL15] stellen ein Faltendes Neuronales Netzwerk (engl. *convolutional neural network*) vor, das als Kostenfunktion in das semi-globale Stereo-Matching [Hiro8] integriert wird. Das Verfahren zeigt verbesserte Ergebnisse, obwohl die dafür benötigte Rechenzeit (ca. 100s je Stereo-Bildpaar) extrem hoch ist. Zhang und Wah [ZW18] argumentieren, dass geeignete Merkmale über Konsistenz und Unterscheidungskraft verfügen müssen – sowohl bei Anwendungen im Stereo-Matching als auch beim optischen Fluss. Auf Basis dieser beiden Maxime schlagen Zhang und Wah ein Neuronales Netz vor, das ebenfalls im Bereich der Korrespondenzsuche ansetzt, jedoch ebenfalls hohe Laufzeiten (ca. 71 s) aufweist.

Ein komplett durch Maschinelles Lernen realisiertes Verfahren wird von Mayer et al. [MIH<sup>+</sup>16] vorgeschlagen. Das darin enthaltene

Neuronale Netz ist mit einem großen synthetischen Datensatz trainiert und erreicht kurze Verarbeitungszeiten auf aktuellen Grafikkarten (angegeben mit  $< 0.1$  s).

Das von Kendall et al. [KMD<sup>+</sup>17] entwickelte „GC-Net“ nutzt 19 verschiedene 3-D-Faltungsschichten (engl. *3-D convolutional layer*) um ein vollständiges Stereo-Matching mit Disparitätsermittlung auf Basis von rektifizierten Stereo-Bildern zu realisieren. Die Autoren geben an, damit Subpixel-Genauigkeit erreichen zu können. Chang und Chen [CC18] bauen auf dem Verfahren von Kendall et al. [KMD<sup>+</sup>17] auf und ermöglichen die Erweiterung auf mehr 3-D-Faltungsschichten, indem sie die Daten geschickt skalieren. Die angegebene Bearbeitungszeit liegt bei 0.9 s auf aktueller Grafikkarte.

Zhang et al. [ZPYT19] schlagen ein neuartiges Verfahren vor, in dem eine differenzierbare Annäherung des semi-globalen Matchings [Hiro8] mit gleichzeitiger Verfolgung unterschiedlicher Suchrichtungen in einer Schicht eines Neuronalen Netzes verwirklicht wird. Dies wird durch eine weitere Schicht ergänzt, die auf feine Bildstrukturen spezialisiert ist, die im Rahmen von Skalierungen potentiell verloren gehen können.

Verfahren, die auf Techniken des Maschinellen Lernens aufbauen, zeigen interessante Eigenschaften hinsichtlich der Qualität der erzeugten Ergebnisse. Wie jedoch in Kapitel 1 dargestellt, kommen diese in der vorliegenden Arbeit nicht zur Anwendung.

### *Stereo-Datensätze*

Eine Vielzahl von Datensätzen mit Stereo-Aufnahmen wurde bereits veröffentlicht. Ein in aktuellen Publikationen viel genutzter Datensatz geht auf Scharstein und Szeliski [SS02] zurück. Sie stellen hochauflösende Datensätze mit Disparitäten als Grundwahrheiten bereit. Die Datensätze sind als „Middlebury Stereo Datasets“ bekannt. Die Sammlung wurde kontinuierlich erweitert [SS03, SP07, HSo7], u. a. von Scharstein et al. [SHK<sup>+</sup>14] mit dem Datensatz aus dem Jahr 2014. Dieser Datensatz enthält unterschiedliche Szenen, die innerhalb von Gebäuden aufgenommen wurden und größtenteils über feine Details verfügen.

Der von Scharwächter et al. [SEFR13, SERF14] veröffentlichte Stereo-Datensatz beinhaltet Bildpaare aus der Fahrerperspektive eines sich im Straßenverkehr bewegendes PKW. Die Bilder sind in Graustufen aufgenommen und werden zusammen mit Disparitätenkarten und semantischen Annotationen für Umfeldklassifikationen angeboten. Auch von diesem Datensatz existieren mehrere Versionen. Der in dieser Arbeit verwendete Datensatz ist „Daimler Urban Segmentation Dataset 2014“ [SERF14].

Das Karlsruher Institut für Technologie (KIT) hat in Zusammenarbeit mit dem Toyota Technological Institute im Bereich der intelligenten Fahrzeuge sehr prominente Datensätze erstellt. Die unter dem

Namen „KITTI Vision Benchmark Suite“ (kurz: KITTI) veröffentlichten Datensätze beinhalten mehrere Sensormodalitäten (u. a. Stereokameras und 3-D-Laser) und adressieren mit unterschiedlichen Zusammenstellungen verschiedene Problemfelder des autonomen bzw. assistierten Fahrens. Die Datensätze enthalten genaue Informationen zu den verwendeten Sensoren sowie deren intrinsische und extrinsische Kalibrierung. Zu Vergleichszwecken sind zudem Klassifizierungsergebnisse als Grundwahrheit enthalten. Die initiale Version der Datensatzsammlung wurde von Geiger, Lenz und Urtasun [GLU12] publiziert und inzwischen durch weitere Datensätze erweitert.

### 5.2.2 Perspektivische 2-D Transformationen mit Homographien

Im Bereich der perspektivischen Geometrie finden sich Ansätze und Transformationen, um die Ansicht einer Kamera so zu transformieren, als sei sie von einem anderen Standpunkt aufgenommen worden. Die Idee hinter den dabei verwendeten Transformationen wird von Hartley und Zisserman [HZ03] und von Vincent und Laganierre [VLo1] beschrieben. Das prinzipielle Vorgehen ist:

Gegeben zwei Kameras  $C_1$  und  $C_2$ . Das von Kamera  $C_1$  aufgenommene Bild soll so transformiert werden, dass es der von Kamera  $C_2$  erwarteten Ansicht entspricht. Es wird die Annahme aufgestellt, dass sich alle Objekte, die in Kamera  $C_1$  zu sehen sind, auf einer Ebene  $E$  im 3-D-Raum befinden.

Die Ebene  $E$  kann im Bildraum der Kamera  $C_1$  mit mindestens vier Punkten beschrieben werden:

$$Q_{C_1} = \{q_i \in \mathbb{R}^2 \mid i \in \mathbb{N}^+\} \quad \text{mit} \quad |Q_{C_1}| \geq 4$$

Für die weiteren Schritte wird die Bildebene der Kameras  $C_1$  und  $C_2$  als projektive Ebene interpretiert. Seien nun die zu  $Q_{C_1}$  korrespondierenden Bildpunkte im Koordinatensystem der Bildebene von Kamera  $C_2$  gegeben:

$$Q_{C_2} = \{q_j \in \mathbb{R}^2 \mid j \in \mathbb{N}^+\} \quad \text{mit} \quad |Q_{C_2}| = |Q_{C_1}|$$

Zur Beschreibung der Transformation zwischen der Ansicht von Kamera  $C_1$  zur Ansicht von  $C_2$  kann nun eine Homographie-Matrix  $H_{C_1 \rightarrow C_2} \in \mathbb{P}^{2 \times 2}$  bestimmt werden. Die Grundlagen bzgl. Homographien und deren Berechnung sind in Kapitel 3.3 zu finden.

Die Matrix  $H_{C_1 \rightarrow C_2}$  kann unter der Voraussetzung von gleichbleibenden Optiken und unveränderter Pose zwischen den Kameras  $C_1$  und  $C_2$  als konstant angenommen werden.

### 5.2.3 Berechnung der Vogelperspektive

Die Berechnung der Vogelperspektive für Anwendungen in Fahrzeugen ist ein in mehreren Publikationen diskutiertes Thema. Homo-

graphien (siehe Kapitel 5.2.2) stellen dabei die vorrangig verwendete Technik dar.

Liu, Kin und Chen [LLCo8] nutzen am Fahrzeug verteilte Kameras und Homographien, um die Ansichten zu transformieren. Abschließend setzen sie die einzelnen Bilder zu einem Gesamtbild (sogenanntes *stitching*) zusammen. Um ein natürlicheres Aussehen zu erreichen, verzerren sie die Bilder in einer Fischaugenansicht. Dadurch werden die unter falschen Annahmen verzerrten Objekte fern der Homographie-Ebene etwas natürlicher dargestellt.

Luo et al. [LKP<sup>+</sup>09] schlagen ein Verfahren vor, das mit günstigen und leistungsschwachen Hardware-Komponenten auskommt. Auch sie nutzen Homographien in Kombination mit Stitching. Sato et al. [SMS<sup>+</sup>13] nutzen Fischaugen-Kameras, Homographien und räumliche und zeitliche Daten, um eine Vogelperspektive zu erzeugen. Li und Hai [LH11] fokussieren sich in ihrer Arbeit auf die Kalibrierung der für die Berechnung notwendigen Kamerasysteme. Eine kosteneffiziente Lösung wird von Thomas et al. [TCPC11] vorgeschlagen. Auch in dieser Publikation werden Homographien für die flächige Transformation verwendet.

Alle diese Ansätze weisen starke Verzerrungen auf, sobald Objekte, die nicht der Ebenenannahme entsprechen, abgebildet werden. Nichtsdestotrotz stellen Homographie-basierte Systeme den aktuellen Stand der Technik dar und werden in den aktuell auf dem Markt erwerbbaaren Systemen eingesetzt.

Bisher sind keine dem Autor bekannten Veröffentlichungen erschienen, die sich mit der perspektivisch korrekten Generierung von Vogelperspektivenansichten beschäftigen.

#### 5.2.4 Bild-basiertes Rendering

Die Berechnung von Ansichten virtueller Kameras wird auch in Publikationen aus dem Themenbereich des Bild-basierten Renderings (engl. *image-based rendering*) diskutiert:

Shum und Kang [SKoo] fassen verschiedene Ansätze für die Interpolation von Ansichten zusammen. Sie kategorisieren existierende Techniken anhand der Komplexität der vorgenommenen geometrischen Modellierung. In Fahrzeugapplikationen können vorab berechnete bzw. vorab modellierte Geometrien nicht zur Anwendung kommen, da sie sich in dynamischen Umgebungen bewegen und auch eine unverzögerte bzw. direkte Verarbeitung möglich sein muss. Für die vorliegende Thematik kommen daher lediglich Ansätze in Frage, die mit impliziter Modellierung arbeiten.

Laveau und Faugeras [LF94] schlagen eine Vorhersage-Methode vor, die die Fundamentalmatrix (siehe Kapitel 5.2.1) und zwei aufgenommene Bilder nutzt. Zinger, Do und De With [ZDD10] diskutieren die Berechnung von freien Blickpunkten auf Basis

von Tiefendaten bzw. Disparitätskarten für 3-D-TV-Anwendungen. Vogt et al. [VKS<sup>+</sup>04] nutzen Bild-basiertes Rendering in Kombination mit Lichtfeldern, um die Bildqualität in Bildfolgen zu verbessern.

Die vorgestellten Ansätze fokussieren sich auf die Berechnung virtueller Ansichten, die zwischen bzw. nahe an real aufgenommenen Ansichten liegen. Dabei werden lediglich kleine Rotationen bzw. Positionsänderungen vorgenommen, wie beispielsweise bei der Berechnung von 3-D-Filmen. Für die Vogelperspektivenansicht in Fahrzeugen müssen jedoch virtuelle Kameraansichten berechnet werden, deren Pose sich stark von der realen Kamera unterscheidet. Die Ansätze aus dem Bild-basierten Rendering lassen sich daher nicht auf diesen Fall anwenden, wenngleich sie grundlegende Informationen zu den voraussetzenden Themen enthalten.

### 5.2.5 Stereo-Hindernisdetektion

Für die Ansicht aus der Vogelperspektive ist es wünschenswert, detektierte Hindernisse aus den Kamerabildern auszuwerten und diese in die Ansicht integrieren zu können. Dabei soll das Verfahren generell Hindernisse bzw. potentielle Gefahren erkennen und korrekt verorten können. An dieser Stelle ist es weniger das Ziel, Hindernisse zu *klassifizieren* (z. B. „Auto“, „Fußgänger“, etc.), sondern vielmehr sind Verfahren von Interesse, die mögliche Kollisionsbereiche identifizieren und den dazugehörigen Abstand präzise ermitteln. Folglich ist die *Detektion* von Hindernissen im Fokus.

Eine Zusammenfassung der aktuellen Verfahren wurde von Bernini et al. [BBC<sup>+</sup>14] veröffentlicht.

Bertozzi und Broggi [BB98] schlagen ein Verfahren vor, dass die Projektion von 3-D-Daten auf eine planare Bodenebene und Fehlräume von 3-D-Punkten nutzt, um mittels statistischer Auswertung der Fehlbereiche auf Objekte zu schließen. Das Verfahren kann geometrisch signifikante und geometrisch sich klar vom Hintergrund abhebbende Objekte, wie z. B. planare Fahrzeugrückseiten umgeben von „freier“ Straße, erkennen. Voraussetzung für das korrekte Finden von Objekten ist zudem, dass das Objekt komplett im Bild zu sehen sind.

Der Ansatz von Oniga und Nedevschi [ON10] nutzt Erhebungskarten, um die Umgebung in ein Raster einzuteilen und eine Analyse auf diesem durchzuführen. Basierend auf der Dichte der 3-D-Punkte je Rastereinheit wird eine Hindernisklassifikation vollzogen. Die Autoren verwenden dabei einen quadratischen Ansatz, um die Bodenoberfläche zu approximieren. Mit einer auf ein Straßenszenario angepassten Filterung wird ferner ein implizites Modell von Umgebungsklassen verwendet, um die Robustheit zu steigern.

Labayrade, Aubert und Tarel [LAT02] erzeugen sogenannte V-Disparitäten-Bilder (engl. *V-disparity*), mit denen sie die Straßengeometrie abbilden. Das Verfahren geht von einem primären Vorkom-

men von Straßen im Bild aus und nutzt die dabei erwartete Geometrie für die Analyse. Eine ebene Fahrbahn ist auch in diesem Verfahren die Annahme. Labayrade und Aubert [LA03] haben ihr Verfahren erweitert, um robuster hinsichtlich Roll- und Nickbewegungen des Fahrzeugs zu werden. Nichtsdestotrotz bleibt die Annahme des ebenen Bodens.

Das von Badino, Franke und Pfeiffer [BFP09] entwickelte Verfahren „Stixel World“ nutzt Stereo-Disparitäten, um den Freiraum vor einem Fahrzeug und damit einhergehend Hindernisse zu detektieren. Das Verfahren nutzt dabei die Annahme, dass die Kamera in Richtung des Horizonts blickt. Dieses Verfahren stellt die Grundlage für die im Folgenden vorgeschlagene Erweiterung dar. Die genaue Darstellung des Verfahrens und die entwickelten Erweiterungen sind in Kapitel 5.8 zu finden.

Auch im Bereich des Maschinellen Lernens gibt es Ansätze, die Objekte bzw. Hindernisse in Stereo-Daten detektieren. Obwohl sie, wie in Kapitel 1 erläutert, in der vorliegenden Arbeit keine Anwendung finden können, sind die Verfahren im Rahmen der Darstellung des Stands der Technik interessant:

Wang et al. [WCG<sup>+</sup>19] generieren Pseudo-LIDAR-Daten aus Stereo-Bildern, um sie als Eingabe für einen auf Laserdaten trainierten Klassifikator zu verwenden. Diese Veröffentlichung veranschaulicht die Wiederverwendung von Klassifikationsmechanismen durch die Imitation von Sensormodalitäten. Li, Chen und Shen [LCS19] detektieren Objekte in Stereo-Daten, indem sie das 3-D-Lokalisierungsproblem als Lernproblem definieren und trainieren. Levi, Garnett und Fetaya [LGF15] stellen einen Ansatz vor, der die Idee hinter der „Stixel World“ aufgreift und die spaltenweise Klassifikation bzgl. des Hindernisses, das der Kamera am nächsten ist, durch Einsatz eines Neuronales Netzes lösen. Dabei wird eine spaltenweise Regression auf den Bilddaten angewendet.

### 5.2.6 Umgebungsrepräsentation mit Rasterkarten

Für Fahrzeuge und Roboter ist es wichtig, die über die Umgebung akquirierten Sensordaten in einer effizienten und vielseitigen Datenstruktur zu repräsentieren. Rasterkarten sind dabei ein weit verbreiteter Ansatz. Martin und Moravec [MM96] fassen verschiedene Entwicklungsstufen dieser Datenstruktur zusammen und dokumentieren die dahinterstehende Evolution.

Rasterkarten werden für die Umgebungsrepräsentation im erweiterten Ansatz für die perspektivisch korrekte Ansicht aus der Vogelperspektive eingesetzt (siehe Kapitel 5.6).

## 5.3 MODELLBILDUNG

Eine Stereokamera ist ein Verbund von (mindestens) zwei Kameras, deren Relativausrichtung zueinander als konstant angenommen wird. Oftmals sind die Bildsensoren einer Stereokamera rigide miteinander verbunden bzw. direkt auf einer Platine verbaut, sodass möglichst wenig mechanisches Spiel auftritt. Effekte, wie beispielsweise thermische Einflüsse auf die Geometrie, sollen so verhindert werden. Die Transformationen zwischen den Kameras werden als konstant angenommen. Die Annahme ist, dass sich die Kameras untereinander nicht verschieben.

Im Folgenden wird die Modellierung eines Sensorverbundes aus mehreren Kameras definiert. Dabei wird die Formulierung generisch für Stereokameras mit einer beliebigen Anzahl von Kameras ( $\geq 2$ ) beschrieben.

Um einen Stereokameraverbund als abgekapselte Einheit zu definieren wird eine Hauptkamera festgelegt, von der aus die Transformationen zu den anderen Kameras im Verbund bekannt sein müssen. Die Transformation zwischen der Hauptkamera und jeweils einer anderen Kamera wird durch eine Relativpose beschrieben, die die Koordinatensysteme ineinander überführt. Die Pose beschreibt die notwendigen Parameter für die weiteren Berechnungen (siehe Kapitel 5.2.1 für Erläuterungen zu Stereo-Paaren). Die genaue Information über diese Pose ist Voraussetzung für die präzise Rekonstruktion von Tiefendaten aus den Stereo-Bildern.

Zusätzlich zur extrinsischen Anordnung der Kameras sind die intrinsischen Kalibrierungen Bestandteil des Stereokameramodells. Eine intrinsische Kalibrierung (siehe Kapitel 3.2) wird durch die Angabe einer Kameramatrix  $\mathbf{K}$ , die die Kamerakonstanten  $k_x, k_y \in \mathbb{R}$  und den Kamerahauptpunkt  $(c_x, c_y)^T \in \mathbb{R}^2$  enthält, sowie eine Menge von Verzerrungskoeffizienten als Vektor  $\boldsymbol{\kappa}$ , die je nach dem eingesetzten Kalibrierverfahren definiert sind, und der Auflösung der Kamera  $w \times h$  mit  $w, h \in \mathbb{N}$  in Pixeln definiert. Die Intrinsik einer Kamera wird in einem Tupel zusammengefasst:

$$\mathcal{J} = (\mathbf{K}, \boldsymbol{\kappa}, w, h) \quad .$$

Ein Stereoverbund ist definiert als ein Tupel, bestehend aus der Konfiguration der Hauptkamera und der Menge der Konfigurationen der anderen Kameras des Stereoverbunds zusammen mit deren Relativpose zur Hauptkamera. Sei  $C_0$  die Hauptkamera und seien  $C_i$  mit  $i \geq 1$  die weiteren Kameras mit jeweiliger Relativpose  $\mathbf{a}_i$ . Der Stereoverbund  $\mathcal{S}$  wird beschrieben durch:

$$\mathcal{S} = (\mathcal{J}_0, \{(\mathcal{J}_1, \mathbf{a}_1), (\mathcal{J}_2, \mathbf{a}_2), \dots, (\mathcal{J}_n, \mathbf{a}_n)\}) \quad .$$

Mit dieser Modellierung ist es möglich, den Stereoverbund als Einheit zu kalibrieren um ihn dann auf einen Sensorträger, z. B. ein Nutzfahrzeug, zu montieren bzw. ihn zu repositionieren. Dies macht im

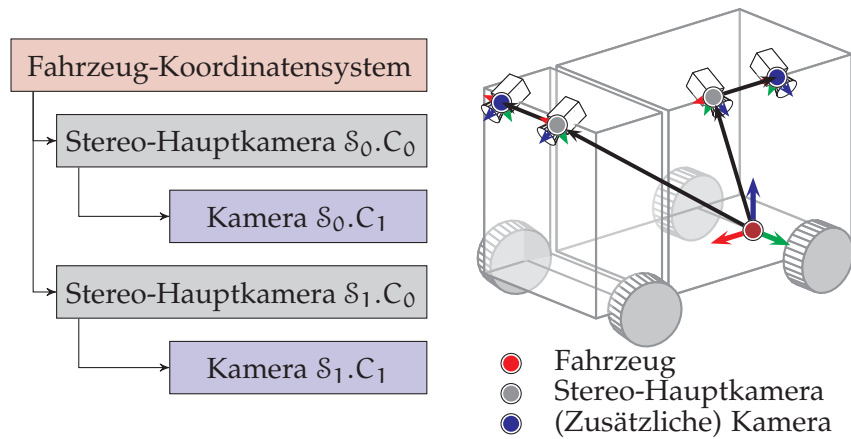


Abbildung 5.4: Beispiel für eine Sensorhierarchie in einem Fahrzeug: Im Fahrzeug-Koordinatensystem sind zwei Stereokameras vorhanden. In jedem Paar hängt die zweite Kamera per Relativpose an der Hauptkamera.

Hinblick auf ein Fahrzeugmodell mit verbauter Sensorik eine Positionierung innerhalb des Fahrzeugs notwendig. Analog zu Kapitel 4.3.1 wird der Ursprung des Fahrzeug-Koordinatensystems auf den Mittelpunkt der Hinterachse gelegt. Die Lage eines Stereoverbundes  $\mathcal{S}_j$  innerhalb des Fahrzeugs wird mit einer Pose  $\mathbf{b}_j$  beschrieben. Ein Fahrzeug mit mehreren verbauten Stereoverbänden wird mit folgender Menge beschrieben:

$$\{(\mathcal{S}_0, \mathbf{b}_0), (\mathcal{S}_1, \mathbf{b}_1), \dots\} \quad .$$

Es ergibt sich eine geometrische Hierarchie, in der die Sensoren angeordnet sind. Eine Stereokamera als Sonderfall des Stereoverbundes ( $n = 1$ ) stellt die im Folgenden verwendete Sensorik dar. Ein einfaches Beispiel einer solchen Geometrie hierarchie mit zwei Stereokamera-Paaren ist in Abbildung 5.4 zu sehen.



## 5.4 HOMOGRAPHIE-VERSCHATTUNG

Wie in Kapitel 5.2.3 beschrieben, stellen Homographie-basierte Verfahren den aktuellen Stand der Technik – sowohl wissenschaftlich als auch in kommerziellen Anwendungen – dar.

Für Fahrzeuganwendungen scheint die bei Homographien enthaltene Grundannahme bezüglich einer Homographie-Ebene auf den ersten Blick passend im Hinblick auf die Bodenebene (Straße). Jegliche Objekte, die sich *nicht* auf dieser Ebene befinden, führen jedoch zu einer Verletzung dieser Annahme. Dies betrifft alle Umgebungsobjekte, die beim Navigieren relevant sind, wie beispielsweise andere Fahrzeuge, Hindernisse und Fußgänger.

Prinzipiell lässt sich festhalten, dass alle Objekte, die für die Fahrzeugnavigation von besonderer Bedeutung sind, nicht adäquat durch eine Homographie transformiert werden können. Bewegen sich Fahrzeuge nicht auf ebenem Grund, wie Straßen, sondern auf unebenem Terrain, wie in agrar- oder forsttechnischen Szenarien oder auf Baustellen, wird auch der eigentliche Fahruntergrund verzerrt dargestellt. Beispiele solch verzerrter Ansichten in einer PKW-Anwendung sind in Abbildung 5.5 zu sehen. Der Boden in diesen Szenarien ist grundlegend gerade (Ebenenannahme). Sobald diese Annahme verletzt wird, treten deutlich Verzerrungen auf, z. B. bei der Darstellung der anderen Fahrzeuge/Felgen.

Der Effekt, der zu den unnatürlichen Darstellungen bei der Verletzung der Ebenenannahme der Homographie führt, wird als **Homographie-Verschattungseffekt** [8, 9] bezeichnet und im Folgenden erläutert:

Gegeben eine Kamera  $C$ , die eine Szene beobachtet und eine *virtuelle* Kamera  $V$ , die die gewünschte Ansicht der Szene darstellt, in die das durch Kamera  $C$  aufgenommene Bild transformiert werden soll. Sei nun  $H_{C \rightarrow V} \in \mathbb{P}^{2 \times 2}$  die Homographie-Matrix, die mittels Korrespondenzen auf der Homographie-Ebene  $E$  zwischen dem Bild von Kamera  $C$  und der gewünschten Ansicht von Kamera  $V$  berechnet bzw. optimiert wurde. Der Aufbau der Transformation ist in Abbildung 5.6 dargestellt.

Ein Objekt, das die Ebenenannahme verletzt (in Abbildung 5.6 der weiße Quader), wird nicht anhand seiner eigentlichen Geometrie transformiert, sondern in der Ansicht so verzerrt, als wäre das Objekt auf die Ebene  $E$  projiziert. Die Form des Objekts entspricht seinem Schattenwurf auf die Ebene  $E$ , wenn eine Punktlichtquelle an der Position von Kamera  $C$  angenommen wird. Das Bild der virtuellen Kamera  $V$  entspricht folglich dem beschriebenen Schattenbild. Ein Beispiel für eine konkrete Verzerrung ist in Abbildung 5.7 zu sehen. Dort wird das zunächst entzerrte Bild aus einer Kamera mittels Homographie in eine virtuelle Vogelansicht transformiert. Die Personen, die aufgrund ihrer Form die Bodenannahme verletzen, werden in ih-



(a) Fahrsituation in engem Raum (Bildquelle: Daimler AG [@1])



(b) Parksituation (Bildquelle: Daimler AG [@2])

Abbildung 5.5: Beispiele eines kommerziellen Systems für eine Vogelperspektivenansicht (engl. *topview*) in Kombination mit einer Rückfahrkamera

ren Homographie-Schatten transformiert und dadurch deutlich und unnatürlich verzerrt. Das sich auf dem Boden befindliche Schachbrettmuster, anhand dessen auch die Ebene für die dargestellte Homographie definiert ist, wird erwartungskonform korrekt transformiert.

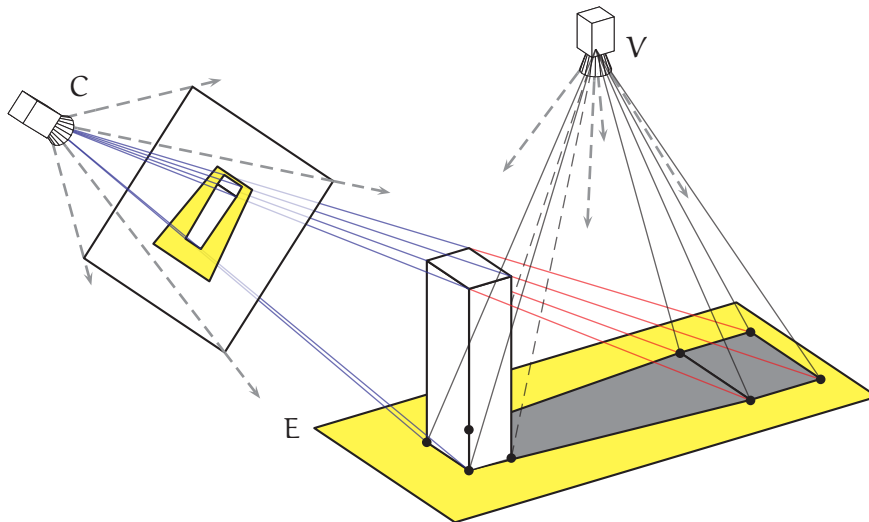
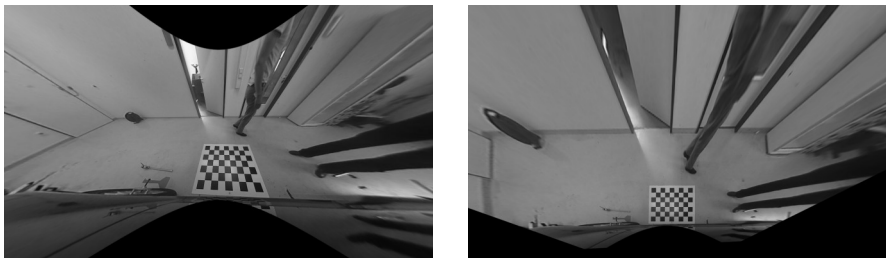


Abbildung 5.6: Homographie-Verschattungseffekt



(a) Entzerrtes Kamerabild

(b) Virtuelle Vogelpersicht

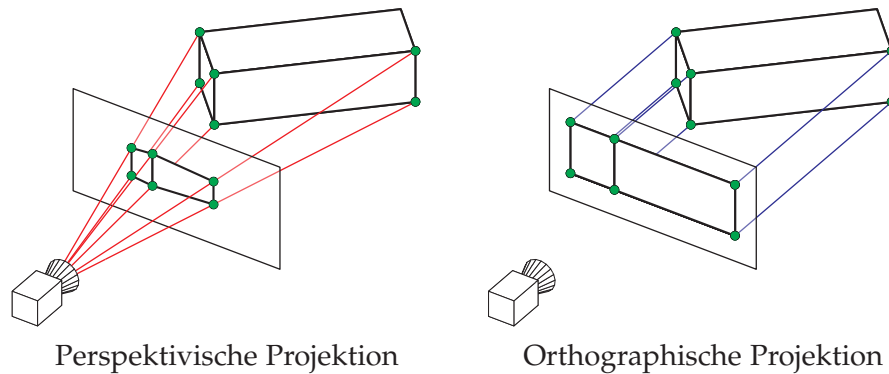
Abbildung 5.7: Beispiel für den Homographie-Verschattungseffekt. Die Personen werden unnatürlich verzerrt.

## 5.5 TRANSFORMATION MIT TIEFENDATEN

Der in diesem Kapitel vorgestellte Ansatz zu Berechnung virtueller Kameraansichten ist bereits publiziert. In einer ersten Veröffentlichung wird der Ansatz generisch beschrieben [9] und anhand von Stereo-Datensätzen getestet und später für Vogelperspektivenansichten spezialisiert [8].

Um eine perspektivisch korrekte Transformation zu ermöglichen, muss die Geometrie der Szene erfasst und dieses Wissen in die Berechnung einbezogen werden. Der Schlüssel hierzu ist die Integration von Tiefendaten. Zur Aufnahme bzw. Extraktion von Tiefendaten bieten sich zunächst mehrere Möglichkeiten. Drei zum aktuellen Zeitpunkt prominente Vertreter sind:

- *Mono-Kamera und Struktur aus Bewegung,*  
Durch die mit sich bewegenden monokularen Aufbauten aufgenommenen Bildsequenzen lässt sich eine Tiefenschätzung berechnen. Neu auftauchende Objekte können erst nach mehreren Frames (d. h. Bildern in Zeitfolge) rekonstruiert werden. Da die



Perspektivische Projektion

Orthographische Projektion

Abbildung 5.8: Beispiele für perspektivische und orthographische Projektion

Tiefendaten möglichst sofort und für jeden Zeitschritt berechnet werden sollten, ist diese Methode weniger geeignet.

- *Stereokameras*,  
Mit synchron aufgenommen Bildern können bei bekannter Kamerageometrie und entsprechenden Matching-Technologien Tiefendaten geschätzt werden (siehe Kapitel 5.2.1).
- *3-D-LIDAR-Sensoren*.  
Die mit (Infrarot-)Lasern arbeitenden Sensoren liefern präzise Tiefendaten, die durch Fusion mit Kamerabildern „eingefärbt“ werden können. Diese Technik liefert präzise Punktwolken, stellt jedoch eine zweite, von der Kamera verschiedene Sensormodalität dar. Zudem sind die Sensoren zum aktuellen Zeitpunkt sehr hochpreisig.

Für die vorgestellten Techniken werden Stereokameras als primäre Quelle für Tiefendaten verwendet. Bei gleichzeitiger Verwendung als Sensor für die Ansichten ergibt sich zudem eine Mehrfachverwendung derselben Hardware. Sicherlich steigt jedoch der Kamerabedarf im Vergleich zu den klassischen Systemen zur Berechnung von Vogelperspektivenansichten.

### 5.5.1 Geometrie virtueller Kameras

Als Voraussetzung für die Berechnung einer virtuellen Kameraansicht muss zunächst eine Definition für die Beschreibung der Abbildungseigenschaften getroffen werden. Dabei werden zunächst *orthographische* und *perspektivische* Projektionen unterschieden.

Die orthographische Projektion nutzt parallel verlaufende Strahlen um die Abbildung auf der Bildebene zu erzeugen. Die abzubildenden Objekte werden sozusagen „eingeebnet“. Die perspektivische Projektion entspricht dem Lochkameramodell (siehe Kapitel 3.1). Die unterschiedlichen Projektionsarten sind in Abbildung 5.8 zu sehen. Eine

Adaption der in Kapitel 3.2 definierten Kameramatrix  $\mathbf{K} \in \mathbb{P}^{2 \times 2}$  ist naheliegend und dient als Grundlage für den Konfigurationsraum einer virtuellen Kamera.

Für die Modellierung einer virtuellen perspektivischen Kamera wird initial die Geometrie des sog. Kamera-Frustums betrachtet (siehe Abbildung 5.9), das den bei der Abbildung verwendeten Kegelstumpf beschreibt. Generell kann diese durch die Blickrichtung der Kamera und die vom Kameramittelpunkt ausgehenden Öffnungswinkel in horizontaler bzw. vertikaler Richtung beschrieben werden. Die Blickrichtung der Kamera ist dabei durch die Ausrichtung der Bildebene an der  $xy$ -Ebene des Kamera-Koordinatensystems als Einheitsvektor in  $z$ -Richtung gegeben. Als Vereinfachung kann eine symmetrische Abbildung im Hinblick auf die Blickrichtung der Kamera angenommen werden. Es gilt daher für Abbildung 5.9:

$$\alpha_l = \alpha_r \quad \alpha_u = \alpha_b$$

mit

- $\alpha_l \in \mathbb{R}$  Breite vom Bildmittelpunkt zum linken Rand,
- $\alpha_r \in \mathbb{R}$  Breite vom Bildmittelpunkt zum rechten Rand,
- $\alpha_u \in \mathbb{R}$  Höhe vom Bildmittelpunkt zum oberen Rand,
- $\alpha_b \in \mathbb{R}$  Höhe vom Bildmittelpunkt zum unteren Rand.

Diese Symmetrieannahme ist ebenfalls in den meisten Kalibrierverfahren (siehe Kapitel 3.2) enthalten. Die trigonometrischen Beziehungen für die Öffnungswinkel  $\alpha_V$  (horizontal) und  $\beta_V$  (vertikal) der Kamera sind:

$$\alpha_V = 2 \cdot \operatorname{atan}(\alpha_r \cdot d^{-1}) \quad \beta_V = 2 \cdot \operatorname{atan}(\alpha_u \cdot d^{-1}) \quad .$$

Die Pixelauflösung der modellierten Kamera ist durch  $w_V \times h_V$  mit  $w_V, h_V \in \mathbb{N}$  beschrieben. Da bereits diese Werte das Seitenverhältnis des Sensorausschnitts auf der Bildebene beschreiben, kann die Kamera mit nur einem Öffnungswinkel beschrieben werden. Dabei wird angenommen, dass der Bildsensor mit quadratischen Pixeln (Seitenverhältnis 1 : 1) arbeitet. Der Konfigurationsraum beinhaltet:

- $w \in \mathbb{N}$  Bildbreite der Kamera in Pixeln,
- $h \in \mathbb{N}$  Bildhöhe der Kamera in Pixeln,
- $\alpha \in \mathbb{R}$  Horizontaler Öffnungswinkel der Kamera.

Die Parameter werden in einem Vektor zusammengefasst. Es ergibt sich der Konfigurationsvektor  $\operatorname{conf}_p$  für eine virtuelle perspektivische Kamera:

$$\operatorname{conf}_p = (w_V, h_V, \alpha_V)^T \quad .$$

Mit dieser Kameradefinition kann die zugehörige perspektivische Projektionsmatrix  $\mathbf{P}(\operatorname{conf}_p) \in \mathbb{P}^{2 \times 2}$  erzeugt werden:

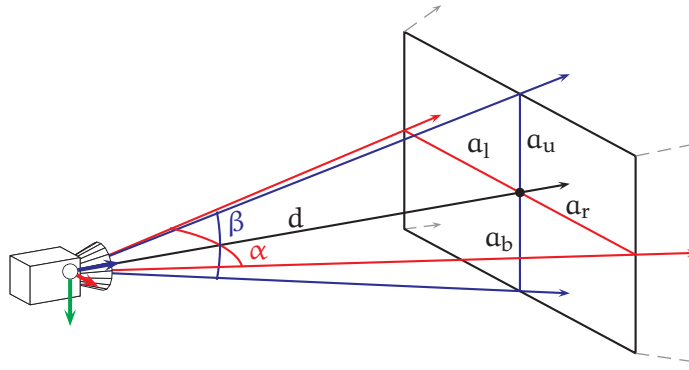


Abbildung 5.9: Parameter für die geometrische Beschreibung eines Kamera-Frustums einer perspektivischen Kamera

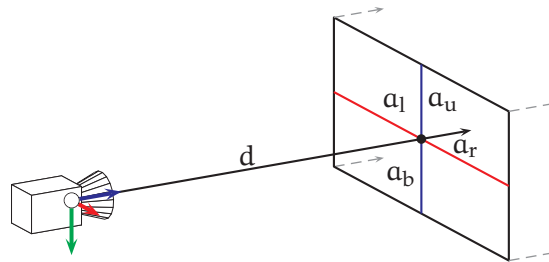


Abbildung 5.10: Parameter für die geometrische Beschreibung einer orthographischen Kamera

$$P_p(\text{conf}_p) = \begin{pmatrix} (\tan(0.5 \cdot \alpha_V))^{-1} & 0 & 0 \\ 0 & w_V(h_V \cdot \tan(0.5 \cdot \alpha))^{-1} & 0 \\ 0 & 0 & 1 \end{pmatrix} .$$

Für die orthographische Kamera gestaltet sich der notwendige Konfigurationsraum einfacher. Die Projektion ist lediglich von den Dimensionen des Bildausschnitts der Bildebene abhängig (siehe Abbildung 5.10), da orthogonal auf diesen abgebildet wird. Es werden dieselben Symmetrieannahmen wie bei der perspektivischen Modellierung angewandt. Unter Berücksichtigung der Bildauflösung der Kamera muss ein Faktor  $s \in \mathbb{R}$  hinzugezogen werden, der die Seitenlänge eines Pixels skaliert. Der Konfigurationsraum besteht aus:

- $w \in \mathbb{N}$     Bildbreite der Kamera in Pixeln,
- $h \in \mathbb{N}$     Bildhöhe der Kamera in Pixeln,
- $s \in \mathbb{R}$     Skalierungsfaktor.

Es ergibt sich der Konfigurationsvektor für eine virtuelle orthographische Kamera:

$$\text{conf}_o = (w_V, h_V, s)^T .$$

Die orthographische Projektionsmatrix unter Berücksichtigung des Kamera-Koordinatensystems ist definiert als:

$$\mathbf{P}_o(\text{conf}_o) = \begin{pmatrix} 2 \cdot (s \cdot w_V)^{-1} & 0 & 0 \\ 0 & 2 \cdot (s \cdot h_V)^{-1} & 0 \\ 0 & 0 & 1 \end{pmatrix} .$$

Zu Vereinfachung der Schreibweise im Folgenden werden die Funktionen  $\mathbf{P}_p$  und  $\mathbf{P}_o$  zur Funktion  $\mathbf{P}$  zusammengefasst:

$$\mathbf{P}(\text{conf}) = \begin{cases} \mathbf{P}_p(\text{conf}) & \text{für conf perspektivisch,} \\ \mathbf{P}_o(\text{conf}) & \text{für conf orthographisch.} \end{cases}$$

### 5.5.2 Punktwolken und virtuelle Kameraansichten

Sei eine Stereokamera  $\mathcal{S}$  mit bekannter Kalibrierung gegeben. Mittels der in Kapitel 5.2.1 vorgestellten Methoden wird eine Disparitätenkarte für die Hauptkamera  $C_0$  (mit der Methode nach Hirschmüller [Hiro8]) berechnet. Jedem Pixel mit gefundener Disparität wird nun ein Tiefenwert zugeordnet. Dieser Tiefenwert kann mit den mittels Kalibrierung und Rektifizierung ermittelten Werten zu einem 3- $D$ -Punkt umgerechnet werden.

Ziel ist es nun, die Ansicht einer virtuellen Kamera  $V$  mit gegebener Konfiguration  $\text{conf}$  zu erzeugen. Die Transformation zwischen den Kamera-Koordinatensystemen von  $C_0$  und  $V$  kann mit einer Pose  $\mathfrak{w}$  beschrieben werden. Der Gesamtaufbau ist in Abbildung 5.11 ersichtlich.

Gegeben die Menge der 3- $D$ -Punkte aus Sicht von Kamera  $C_0$ :

$$P_{C_0} = \{\mathbf{p}_{C_0} \mid \mathbf{p}_{C_0} \in \mathbb{R}^3\} .$$

Die Transformation in das Koordinatensystem der Kamera  $V$  erfolgt mittels der Relativpose  $\mathfrak{w}$  mit der Transformationsmatrix  $\mathbf{T}(\mathfrak{w}) = \mathbf{T}_{C_0 \rightarrow V}$  (siehe Kapitel 2.2):

$$P_V = \Upsilon_V = \{\mathbf{T}_{C_0 \rightarrow V}(\mathbf{p}) \mid \mathbf{p} \in P_{C_0}\} .$$

Die Projektion der Punkte  $P_V$  auf die Bildebene von  $V$  wird durch die Funktion  $\Gamma : \{\mathbb{R}^3\} \rightarrow \{\mathbb{R}^2\}$  beschrieben. Es gilt:

$$Q_V = \Gamma_V(P_V) \quad \Gamma_V = \{\gamma_V(\mathbf{p}) \mid \mathbf{p} \in P_V\}$$

mit  $\gamma_V : \mathbb{R}^3 \rightarrow \mathbb{R}^2$  der Projektionsfunktion für einen einzelnen Punkt und  $Q_V = \{\mathbf{q}_V \mid \mathbf{q}_V \in \mathbb{R}^2\}$  die Menge der auf die Bildebene von  $V$  projizierten Punkte:

$$\gamma_V(\mathbf{p}) = \mathbf{I}_V \cdot \mathbf{P}(\text{conf}) \cdot \mathbf{p}$$

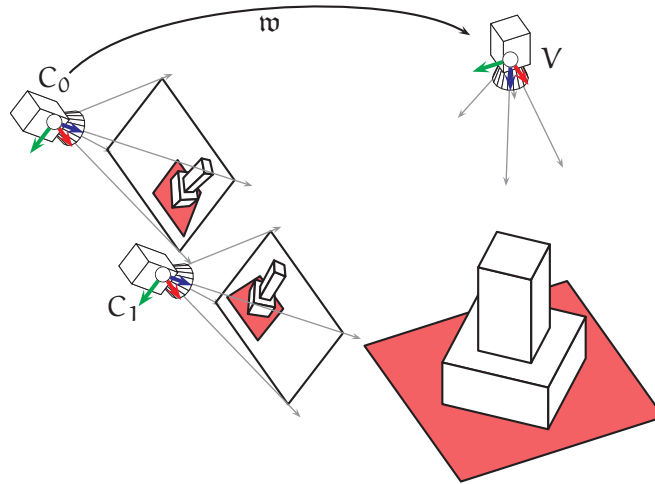


Abbildung 5.11: Koordinatensysteme für die Berechnung einer virtuellen Kamera. Die Stereokamera  $S$  bestehend aus den Kameras  $C_0$  und  $C_1$  wird genutzt, um zu  $C_0$  korrespondierende Disparitäten zu berechnen. Die Pose  $w$  transformiert zwischen den Kamera-Koordinatensystemen von  $C_0$  und dem der virtuellen Kamera  $V$ .

mit  $I_V$  als die Transformationsmatrix in Pixelkoordinaten im virtuellen Kamerabild:

$$I_V = \begin{pmatrix} 0.5 \cdot w_V & 0 & 0.5 \cdot w_V \\ 0 & 0.5 \cdot h_V & 0.5 \cdot h_V \\ 0 & 0 & 1 \end{pmatrix} .$$

Für eine perspektivische Projektion gilt im Hinblick auf die intrinsische Kameramatrix der virtuellen Kamera  $V$ :

$$K_V = I_V \cdot P(\text{conf}_p) .$$

Die resultierende Gesamttransformation der 3-D-Punkte aus Sicht von Kamera  $C_0$  in das Bild-Koordinatensystem der virtuellen Kamera  $V$  wird beschrieben durch:

$$Q_V = \Gamma_V \circ \Upsilon_V(P_{C_0}) .$$

Das dargestellte Vorgehen beschreibt die Transformation aus *einer* Quellkamera mit Tiefendaten. Das Verfahren lässt sich auf beliebig viele ( $n$ ) Quellkameras erweitern:

$$Q_V = \Gamma_V \left( \bigcup_{i=1}^n \Upsilon_{C_i}(P_{C_i}) \right) .$$

Da im Zielbild der virtuellen Kamera eine Rasterisierung in Pixel vorgenommen wird, können potentiell mehrere Punkte auf ein Pixel



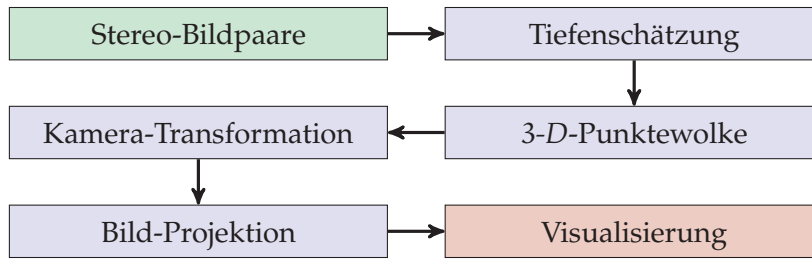


Abbildung 5.12: Schema für die Überführung aus Stereo-Bildpaaren in eine perspektivisch korrekte Ansicht einer virtuellen Kameraperspektive

abgebildet werden. Die Pixelkoordinate  $\mathbf{q}'$ , auf die abgebildet wird, ist wie folgt definiert:

$$\mathbf{q}'_V = \lfloor \mathbf{q}_V \rfloor = (\lfloor q_x \rfloor, \lfloor q_y \rfloor)^T \quad \mathbf{q}'_V \in \mathbb{Z}^2 \quad .$$

Durch die Einschränkung des Wertebereichs für die Pixelkoordinate auf  $[0; w_V[$  für die  $x$ -Komponenten und  $[0; h_V[$  für die  $y$ -Komponenten wird sichergestellt, dass sich der abgebildete Punkt auf dem Bildausschnitt der Bildebene befindet. Verletzt die berechnete Pixelkoordinate die Einschränkung, wird der Punkt verworfen, da er außerhalb des sichtbaren Bereichs liegt. Dadurch kann der Wertebereich für  $\mathbf{q}'_V$  auf  $\mathbb{N}^2$  mit den jeweiligen Intervallen in  $x$ - bzw.  $y$ -Richtung weiter eingeschränkt werden.

Das vorgestellte Verfahren beschreibt die geometrische Konvertierung der Stereo-Punktwolke in das Bild-Koordinatensystem der virtuellen Kamera  $V$ . Die dazu notwendigen Schritte sind in Abbildung 5.12 zusammengefasst.

Selbstverständlich ist auch die zu dem ursprünglichen Bildpixel gehörende Farbinformation relevant für die Erzeugung der virtuellen Ansicht. Bei einer Stereokamera ergibt sich die zu einem 3- $D$ -Punkt zugeordnete Farbe aus dem ursprünglichen Kamerabild. Ein Punkt  $\mathbf{p}_{C_0}$  wird anhand der Disparitätenkarte berechnet. Ihm wird der Farbwert an der entsprechenden Stelle im Bild zugewiesen. Um die Formeln lesbar zu halten, wurde dieser Aspekt bisher vernachlässigt. Zur Vereinfachung wird daher eine Funktion `colorMap` definiert, die zu einem gegebenen Punkt (3- $D$  und 2- $D$ ) die Farbzuordnung (beispielsweise  $RGB$ -Tupel oder Grauwert) im Ursprungsbild vornimmt.

Gegeben nun die Menge  $P_{V;x,y} \subseteq P_V$ , die für ein Pixel an der Position  $\mathbf{a} = (a_x, a_y)^T$  des Kamerabildes von  $V$  die möglichen Ursprungs-3- $D$ -Punkte  $\mathbf{p}_V$  enthält:

$$P_{V;\mathbf{a}} = \left\{ \mathbf{p}_V \mid \mathbf{p}_V \in P_V, \lfloor \gamma_V(\mathbf{p}_V) \rfloor = (a_x, a_y)^T \right\} \quad .$$

Für den Fall, dass kein Punkt auf das Pixel projiziert, liegt ein Fehlpixel vor. Wird genau ein Punkt auf das Pixel abgebildet, so wird die

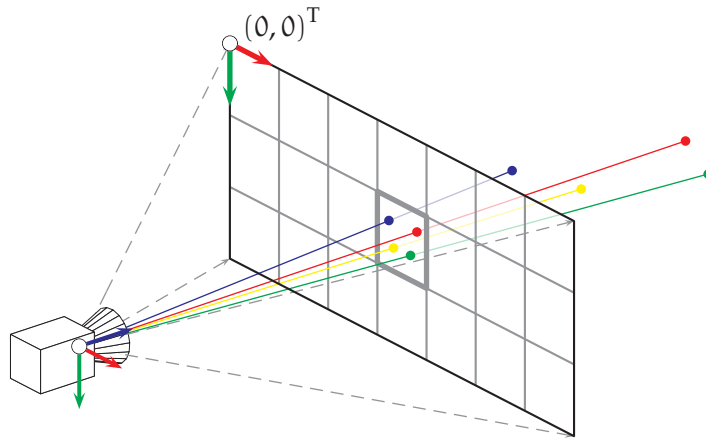


Abbildung 5.13: Problem bei der Rasterisierung: Durch die diskrete Abtastung des Bildsensors in Pixel können mehrere 3-D-Weltpunkte auf einen Pixel abgebildet werden. Da ein Pixel als kleinste Einheit eines digitalen Bildes lediglich eine Farbinformation abbilden kann, muss entschieden werden, wie die Farbe ausgewählt wird.

Farbinformation des Punktes übernommen. Für den Fall, dass mehrere Punkte auf einen Bildpunkt abgebildet werden, muss entschieden werden, wie das Pixel eingefärbt wird ( $\text{colorMap}(\mathbf{p}'_V)$ ).

Zur Färbung des Zielpixels ist es sinnvoll, die Farbinformation des Punktes zu übernehmen, der am nächsten zur Kamera liegt. Dadurch lassen sich Fehler durch gegenseitige Überdeckung ausschließen. Der Sachverhalt wird in Abbildung 5.13 dargestellt. Der Punkt wird wie folgt ermittelt:

$$\mathbf{p}'_V = \underset{\mathbf{p}_V \in \mathcal{P}_{V,a}}{\text{argmin}} \|\mathbf{p}_V\| \quad \Rightarrow \quad \text{colorMap}(\mathbf{p}'_V) \quad .$$

### 5.5.3 Ergebnisse mit generischen Datensätzen

Um das prinzipielle Vorgehen für die Berechnung virtueller Kameraansichten zu prüfen [9], werden hochauflösende Stereo-Datensätze verwendet. Dabei kommen Datensätze aus den „Middlebury Stereo Datasets“ aus dem Jahr 2014 zum Einsatz [SHK<sup>+</sup>14] (siehe Kapitel 5.2.1).

Die Beurteilung der erreichten Bildqualität kann jedoch mangels adäquater Referenz nur anhand manueller geometrischer Interpretation der Ergebnisbilder erfolgen. Die folgenden Berechnungsergebnisse zeigen die rohen Ausgaben des entwickelten Verfahrens. Fragestellungen wie beispielsweise das Füllen vorhandener Fehlpixel bzw. -bereiche und weitere Bildverbesserungsmechanismen werden an dieser Stelle nicht weiter beleuchtet, da die perspektivisch korrekte Transformation der Bilddaten vordergründig dargestellt wird.

In einem ersten Test wird lediglich die Projektionsart der Abbildung geändert, um perspektivische Einflüsse (Fluchtpunkte etc.) in der Bilddarstellung zu eliminieren. Die Relativpose  $\mathfrak{w}$  der virtuellen Kamera entspricht folglich der Null- bzw. Identitätspose ( $\mathfrak{w} = \mathfrak{o}$ ).

Abbildung 5.14 zeigt die Ergebnisse für den Datensatz „playtable“. Das Originalbild (5.14a) ist mit einer Kamera aufgenommen, die auf dem Lochkameraprinzip basiert. Entsprechend ist eine perspektivische Projektion auf die Bildebene zu sehen. Dies wird im Vergleich zum Ergebnis für die virtuelle orthographische Kamera mit gleicher Kamerapose (5.14b) deutlich. Gut zu sehen ist dies bei den nun parallel verlaufenden Kanten der Tischplatte, den Tischbeinen und dem vorderen Stuhl.

Die Ansicht aus der Vogelperspektive ist an der Tischplatte als „Bodenebene“ ausgerichtet. Das Verfahren mit klassischer Homographie-Technik (5.14c) zeigt starke Verzerrungen für alles, was nicht auf dieser Ebene liegt. Dies ist unter anderem an den Stühlen und den Tischbeinen gut zu erkennen. Die mit der vorgeschlagenen Methode berechnete Ansicht (5.14d) weist geometriebedingt zwar Flächen ohne Bildinformationen auf, jedoch sind die sichtbaren Objekte korrekt und ohne Verzerrung dargestellt. Dies ist beispielsweise an der Tischplatte, die in der Ansicht zu einem Rechteck wird, ersichtlich. Ein zunächst evtl. merkwürdig erscheinender Effekt ist, dass Teile des Teppichbodens durch die Tischplatte hindurch zu sehen sind. Allerdings liegt auch hier eine korrekte Projektion vor. In der Seitenansicht der ursprünglichen Aufnahme ist auch der Bereich unter dem Tisch sichtbar. Da der entsprechende Bereich der Tischplatte durch die Objekte auf dem Tisch verschattet wird, ist dort daher der unterliegende Bereich zu sehen.

Der in Abbildung 5.15 verwendete Datensatz „pipes“ zeigt eine verschachtelte Rohrstruktur mit Armaturen, die sich selbst verdecken (5.15a). Die orthographische Projektion (5.15b) zeigt deutlich den Einfluss der Perspektive auf die Projektion von Geometrien mit stark unterschiedlichen Tiefendaten. Wo im Originalbild der Eindruck einer fast komplett mit Rohren verdeckten Wand entsteht, zeigt die orthographische Projektion klarer die Strukturen der Geometrie.

Die orthographische Vogelansicht (5.15c) zeigt lediglich die im Kamerabild sichtbaren vorderen Teile der Rohre. Die Struktur der rechtwinklig abknickenden Rohrleitungen ist trotz der großen Flächen ohne Bildinformation jedoch eindeutig zu erkennen.

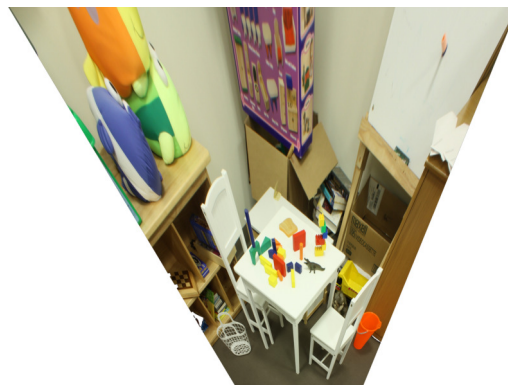
Der Datensatz „recycle“ (siehe Abbildung 5.16) stellt einen für die Transformation in eine virtuelle Kameraansicht interessanten Aufbau dar. Die primär von der Mülltonne sichtbaren Bereiche (5.16a) sind solche, die aus einer Vogelperspektive nicht zu sehen sind. Die orthographische Projektion (5.16b) lässt erkennen, dass die Mülltonne eine nach unten verjüngte Form hat, die sich in der perspektivischen Ansicht schlechter erkennen lässt.



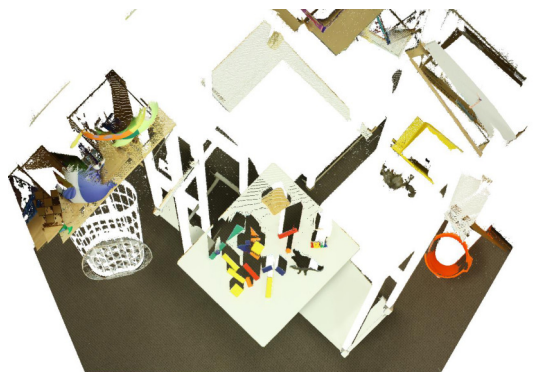
(a) Originalbild [SERF14]



(b) Orthographische virtuelle Kamera

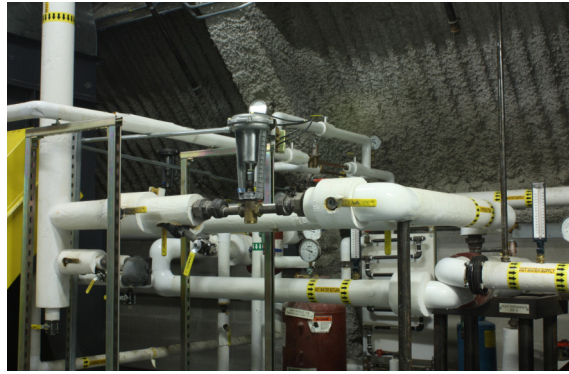


(c) Vogelperspektive mittels Homographie (siehe Kapitel 5.2.2)

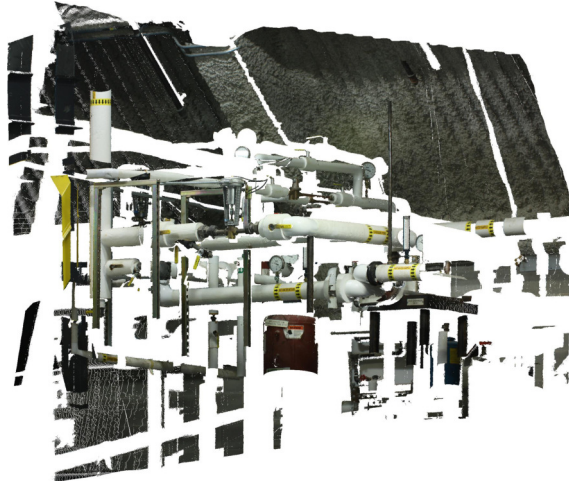


(d) Orthographische virtuelle Vogelperspektive

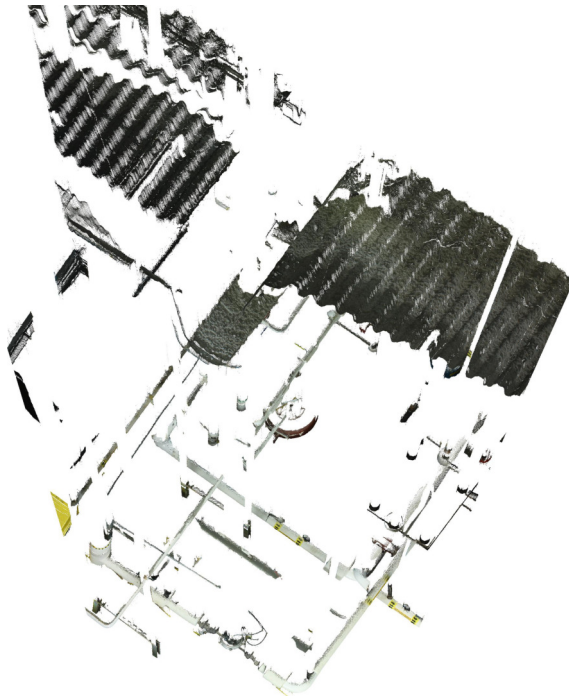
Abbildung 5.14: Testergebnisse für den Datensatz „playtable“



(a) Originalbild [SERF14]



(b) Orthographische virtuelle Kamera



(c) Orthographische virtuelle Vogelperspektive

Abbildung 5.15: Testergebnisse für den Datensatz „pipes“

Der Umriss der Tonne ist in der orthographischen Vogelperspektive gut zu erkennen. Jedoch bleiben große Bereiche ohne Bilddaten. Dies ist dadurch zu erklären, dass viele Punkte aufgrund der geometrischen Struktur „aufeinander“ projiziert wurden.

Im Fall des Datensatzes „piano“ wird deutlich (siehe Abbildung 5.17), dass die Flucht der Klaviatur (5.17a) durch die orthographische Projektion (5.17b) entfernt wird. Die perspektivische Projektion aus einer anderen Kamerapose (5.17c) erzeugt einen perspektivischen Eindruck aus einem anderen Blickpunkt.

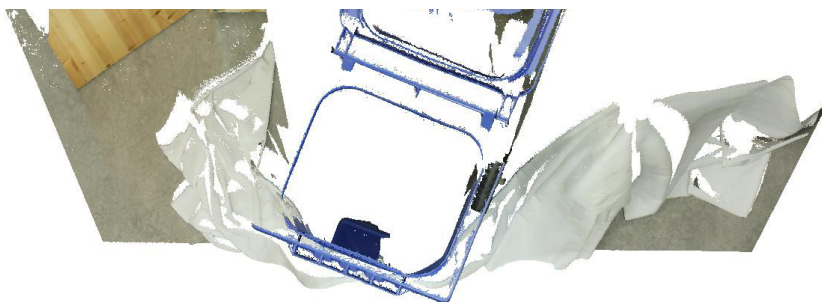
Der Datensatz „shelves“ (siehe Abbildung 5.18) zeigt parallel verlaufende Regale, die sich durch die ausgewählte Perspektive leicht zu einem Fluchtpunkt auf der linken Seite verzerren (5.18a). Bei der Tasche ist nicht klar, ob sie in der Tiefe des Regalbretts aufgehängt ist. Die orthographische Projektion (5.18b) mit parallelen Regalen lässt aufgrund ihrer Fehlflächen rund um die Tasche bereits vermuten, dass sie sich näher an der Kamera befindet. Die seitliche perspektivische Projektion (5.18c) zeigt deutlich, dass sich die Tasche deutlich vor dem Regal befindet und auch dass der Gehstock schräg nach vorne hängt. Die fluchtenden Geraden der Regale sind hier aufgrund der gewählten Pose für die virtuelle Kamera ausgeprägter zu sehen als im Originalbild.



(a) Originalbild [SERF14]



(b) Orthographische virtuelle Kamera

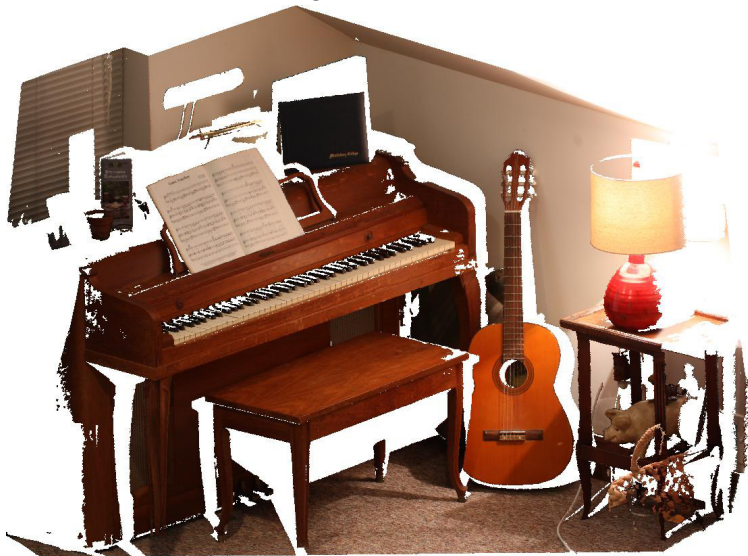


(c) Orthographische virtuelle Vogelperspektive

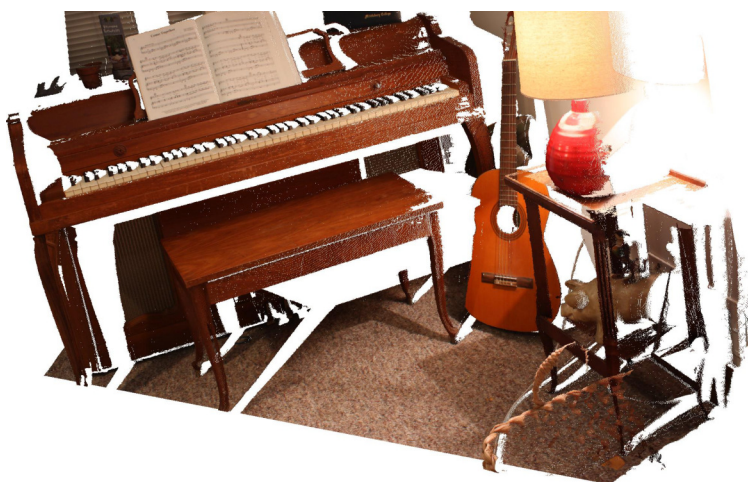
Abbildung 5.16: Testergebnisse für den Datensatz „recycle“



(a) Originalbild [SERF14]



(b) Orthographische virtuelle Kamera



(c) Perspektivische virtuelle Kamera

Abbildung 5.17: Testergebnisse für den Datensatz „piano“





(a) Originalbild [SERF14]



(b) Orthographische virtuelle Kamera



(c) Perpektivische virtuelle Kamera

Abbildung 5.18: Testergebnisse für den Datensatz „shelves“

#### 5.5.4 Ergebnisse mit Fahrzeug-Stereo-Datensätzen

Die entwickelte, punkt-basierte Methode zur Berechnung virtueller Kameraperspektiven zeigt vielversprechende Ergebnisse, deren Korrektheit anhand der dargestellten Beispieldatensätze verdeutlicht wird. Die Eingabedaten aus den *Middlebury*-Stereo-Datensätzen [SHK<sup>+</sup>14] sind qualitativ sehr hochwertig und wurden ursprünglich zur Evaluation von Stereo-Matching-Algorithmen (siehe Kapitel 5.2.1) aufgenommen. Im Folgenden wird nun geprüft, ob sich die Ergebnisse des entwickelten Verfahrens auch mit qualitativ schlechteren Daten, die in Bruchteilen von Sekunden in Fahrzeugen berechnet werden, übertragen lassen. Die in diesem Kapitel vorgestellten Ergebnisse wurden im Rahmen einer internationalen Konferenz publiziert und präsentiert [8].

Der gewünschte geometrische Aufbau für eine Rundumsicht beschreibt mehrere Stereokamera-Paare, die die direkte Fahrzeugumgebung betrachten (siehe Kapitel 5.3). Ein speziell für diesen Fall aufgenommener Datensatz bzw. eine Grundwahrheit für diesen Fall ist nach aktuellem Stand nicht (öffentlich) verfügbar. Dies lässt damit keine quantitative Evaluation der Ergebnisse zu. Ein direkter Vergleich mit klassischen Methoden (Homographie-basiert) ist jedoch möglich. Es wird ein öffentlich verfügbarer Stereo-Datensatz verwendet, um die Reproduzierbarkeit der Ergebnisse zu gewährleisten.

Um die Anwendbarkeit des vorgestellten Verfahrens zu testen, wird der *Daimler Urban Segmentation Dataset* verwendet. Der Datensatz enthält Stereo-Aufnahmen, die mit einem PKW aus Fahrerperspektive aufgenommen sind. Die abgefahrenen Strecken beinhalten verschiedene Szenarien. Konkret findet die Version aus dem Jahr 2014 Anwendung [SERF14].

Für die Berechnung der Ergebnisbilder werden die im Datensatz enthaltenen Kalibrierinformationen und Disparitäten genutzt. Da lediglich Grauwertbilder mit der verwendeten Stereokamera enthalten sind [SERF14], können keine klassischen Farbbilder als Ergebnisse berechnet werden. Um die geometrischen Dimensionen besser zu verdeutlichen, werden die Ausgabebilder farblich kodiert. Die Kodierung erfolgt je RGB-Kanal und hat folgende Semantik:

- Rot:* Objekte/Hindernisse über dem Boden (via Schwellwert),
- Grün:* Grauwert-Intensität aus dem Originalbild,
- Blau:* Nähe zum Fahrzeug.

Abbildung 5.19 vergleicht eine klassische Homographie-Vogelperspektive und zeigt eine innerstädtische Straßenszene an einem Fußgängerüberweg. Die kreuzenden Passanten und ein Fahrradfahrer werden bei der klassischen Berechnungsmethode stark verzerrt (Homographie-Verschattungseffekt). Bereits die Unebenheit der Fahrbahn führt zu einer Verzerrung der Straßenmarkierung

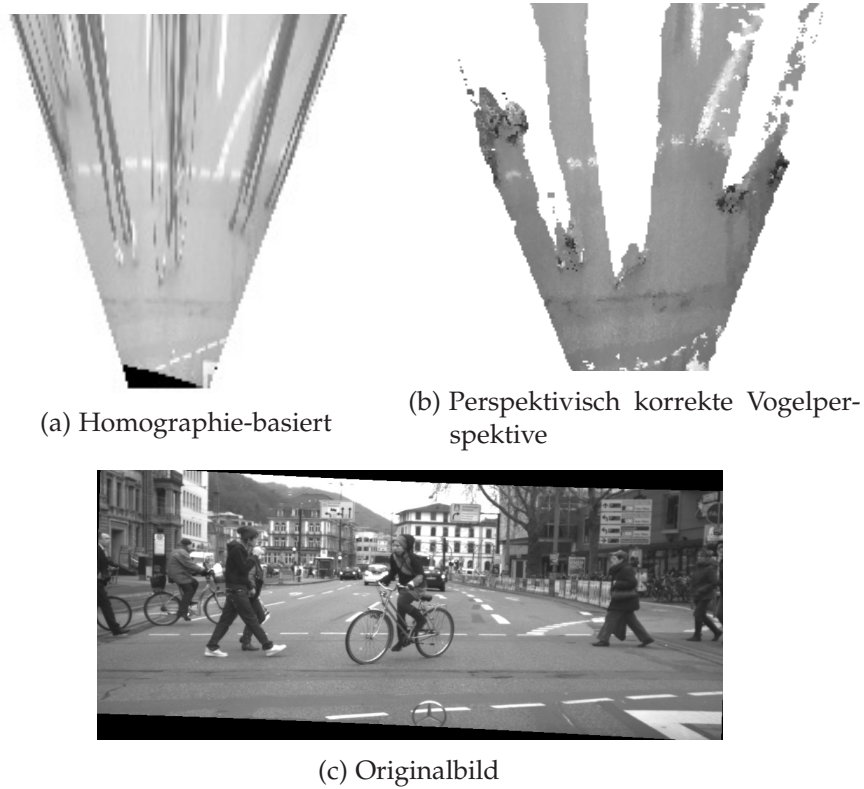


Abbildung 5.19: Homographie-basierte Vogelperspektive (5.19a) und perspektivisch korrekte Vogelperspektive (5.19b). Bild #656 (5.19c) des Datensatzes „test\_2“ [SERF14].

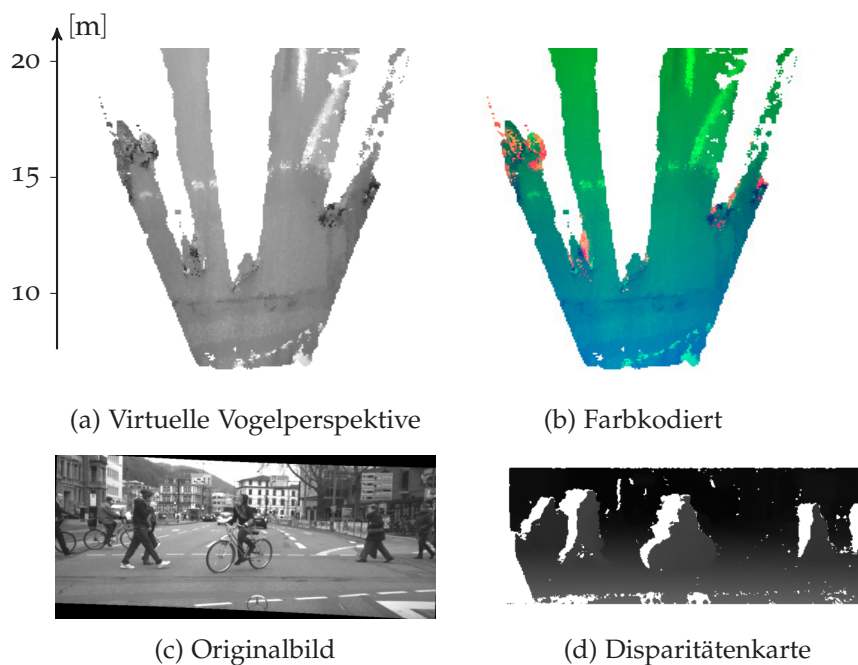


Abbildung 5.20: Virtuelle Vogelperspektive zu Datensatz „test\_2“, Bild #656

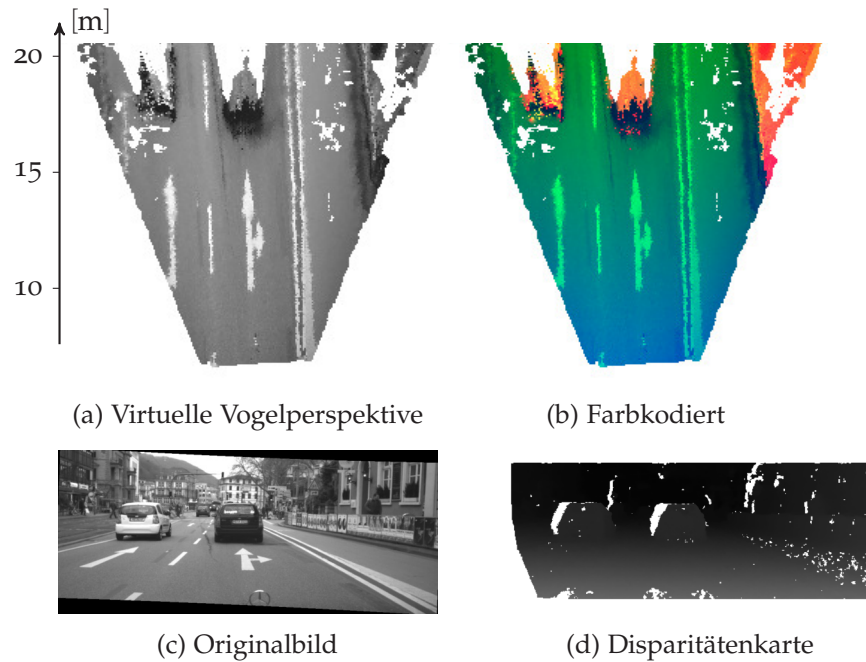


Abbildung 5.21: Virtuelle Vogelperspektive zu Datensatz „test\_2“, Bild #394

in der Ansicht (5.19a). Die perspektivisch korrekte Ansicht zeigt hingegen keine Verzerrungen, jedoch auch große Fehlbereiche hinter den Passanten bzw. hinter dem Fahrradfahrer (5.19b). Die leichte Sprengelung der Fahrbahnmarkierung ist durch weniger präzise Tiefendaten im Vergleich zu den Daten aus dem Middlebury-Datensatz (siehe Kapitel 5.5.2) zu erklären.

Die Abbildungen 5.20, 5.21 und 5.22 zeigen weitere Berechnungsbeispiele mit der vorgeschlagenen Methode. Die in Abbildung 5.20 dargestellte Situation entspricht der in Abbildung 5.19 und ergänzt die farbliche Visualisierung. Abbildung 5.21 zeigt eine Straßenszene in vorwärtiger Fahrt. Die Bodenmarkierungen in ca. 11 bis 15 m Entfernung vor der Kamera sind klar erkennbar, ebenso die vorausfahrenden Fahrzeuge. Durch das Fehlen lediglich einzelner kleinerer Bereiche in der Disparitätenkarte sind nur wenige Fehlbereiche ersichtlich. Abbildung 5.22 zeigt eine Fahrzeugposition unmittelbar vor einer Kreuzung mit querenden Passanten. Die Geometrie wirkt korrekt, jedoch ist im Bereich der Fahrbahnmarkierungen bzw. der Halteinie eine Sprengelung der eigentlich vollständig gefärbten Linien zu erkennen. Dies ist wiederum auf Ungenauigkeiten in der Disparitätenschätzung zurückzuführen.

In der anhand von Fahrsequenzen berechneten Folge von einzelnen Vogelperspektiven lässt sich erkennen, dass die vorgeschlagene Methode einen guten Ausgleich der durch das Fahrwerk bedingten Dynamik erreicht: Fährt das Fahrzeug über unebenen Grund wie beispielsweise Bodenwellen, Straßenbahnschienen oder Teerfli-

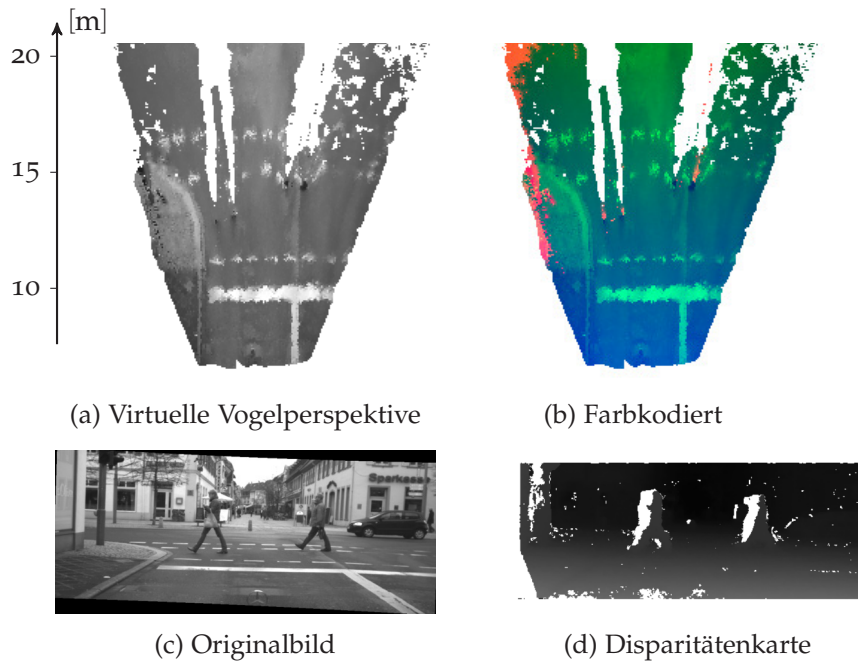


Abbildung 5.22: Virtuelle Vogelperspektive zu Datensatz „train\_1“, Bild #951

cken, ergibt sich durch das Federverhalten des Fahrwerks ein „Wipp-Effekt“ in der klassischen Homographie-Vogelansicht. Dies ist durch die Verletzung der Bodenannahme *für den Boden selbst* und damit auch für alle anderen Objekte in Kombination mit dem Homographie-Verschattungseffekt zu erklären. Dieser Effekt wird in steigender Entfernung zum Fahrzeug entsprechend größer. Das vorgestellte Verfahren kann diesen Effekt aufgrund der Rückführung auf die 3-D-Geometrie ausgleichen und führt zu wesentlich „ruhigeren“ Bildfolgen. Dieser Effekt wird insbesondere beim Betrachten von Bildfolgen (als Video) verdeutlicht und kann daher nicht sinnvoll in einer Abbildung dargestellt werden.

Im Gegensatz zu den in Kapitel 5.5.3 gezeigten Ergebnissen anhand sehr hochauflösender Datensätze verfügen die Fahrzeugdatensätze über eine deutlich geringere Auflösung mit weniger dichten Disparitätenkarten. Daraus resultiert unweigerlich eine vergleichsweise niedrige Auflösung für die perspektivisch korrekte Ansicht aus der Vogelperspektive. Fehlbereiche in den Tiefendaten resultieren zudem in Fehlbereichen in der berechneten Vogelperspektive. Wenngleich die Anwendbarkeit des vorgestellten Verfahrens und seine Korrektheit in der Projektion gezeigt wird, liefern die Fahrzeug-bezogenen Datensätzen mit Daten repräsentativer Sensorik keine zufriedenstellende Qualität in der resultierenden Ansicht aus der Vogelperspektive. Als Ziel an dieser Stelle ist ein Verfahren wünschenswert, das eine dichtere Ansicht mit höherer Auflösung liefert und gleichzeitig eine geometrische Korrektheit aufweist.

## 5.6 INTEGRATION UND GESCHLOSSENES OBERFLÄCHENMODELL

Um die Auflösung und Darstellungsqualität der perspektivisch korrekten Ansicht aus der Vogelperspektive (siehe Kapitel 5.5) zu verbessern, wird eine Modellierung der Umgebung anhand der Stereotiefeninformationen vorgenommen. Diese Modellierung dient als Approximation der Umgebungsgeometrie und bildet die Grundlage für die angestrebte Vogelperspektivenansicht. Die dazu notwendigen Schritte werden in den folgenden Abschnitten vorgestellt. Die Ergebnisse sind im Rahmen einer internationalen Konferenz veröffentlicht [10]. Die Übertragbarkeit der Modellierung auf ein anderes Anwendungsszenario (Integration dünner multispektraler Daten in eine hochaufgelöste Textur) ist zudem in einer weiteren Publikation dokumentiert [13].

### 5.6.1 Gleitende lokale Rasterkarte

Zur Modellierung des Umgebungsmodells muss eine geeignete Datenstruktur genutzt werden. Anforderungen an diese Datenstruktur sind unter anderem:

- Eignung zur Integration von Punktwolken,
- Skalierbarkeit im Hinblick auf die Fahrzeugbewegung und
- Geometrische Interpretierbarkeit.

Rasterkarten (engl. *grid map*) haben sich für die strukturierte Organisation von Sensordaten aus der Umgebung etabliert und können die genannten Anforderungen erfüllen. Eine angepasste Datenstruktur, die auf das vorliegende Problem zugeschnitten ist, wird im Folgenden beschrieben:

Als grundlegende Datenstruktur wird eine Rasterkarte verwendet. Eine Rasterkarte teilt den Raum durch eine Hyperebene (im 3-*D*-Fall eine Ebene). Auf dieser Hyperebenen werden Zellen definiert, die entsprechende Teilausschnitte des Raums beschreiben. Dies stellt eine Analogie zur Bildraasterisierung einer Kamera dar. Die Zellen der Rasterkarte werden als äquidistant definiert, die Basis der Zellen wird als orthogonal festgelegt. Dadurch entsteht ein gleichmäßiges, rechtwinkliges Raster auf der Ebene.

Die Ausrichtung der Rasterkarte ist dabei an der Ebene unter dem Fahrzeug orientiert und entspricht im Wesentlichen der Ausrichtung der Homographie-Ebene bei den klassischen Verfahren (idealisierte Bodenebene). Als Referenz-Koordinatensystem wird das Welt-Koordinatensystem festgelegt.

Abbildung 5.23 zeigt die Rasterkarte und ihr Referenz-Koordinatensystem: Der Ursprung des Welt-Koordinatensystems

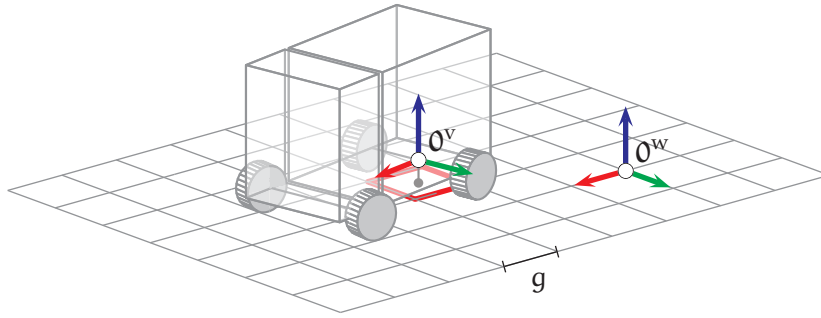


Abbildung 5.23: Rasterkarte unter einem Fahrzeug mit Welt-Koordinatensystem  $w$  und Fahrzeug-Koordinatensystem  $v$

$0^w$  definiert den Ursprung der Rasterkarte. Die Karte ist an den Orthonormalbasen der  $xy$ -Ebene des Weltkoordinatensystems ausgerichtet und verfügt selbst auch über orthonormale Achsen. Die Zellengröße wird als quadratisch definiert. Der Parameter  $g \in \mathbb{R}$  beschreibt die Seitenlänge einer Zelle. Die Positionen der einzelnen Zellen sind und bleiben konstant im Hinblick auf den Ursprung des Welt-Koordinatensystems und verändern sich während der Verwendung nicht.

Dadurch wird die diskrete Abtastung der Hyperebene erreicht. Diese ermöglicht eine indizierte Adressierung einer einzelnen Zelle  $c$  mit einer 2- $D$ -Koordinate  $\mathbf{q}_c \in \mathbb{Z}^2$ .

Für die Modellierung der Rasterkarte werden deren Grenzen als unendlich angenommen. Die Position und Ausrichtung des Fahrzeugs relativ zum Welt-Koordinatensystem wird mit der Pose  $\mathbf{v} \in \mathbb{R}^3 \times \text{SO}(3)$  mit  $\mathbf{v} = \langle \mathbf{p}, \mathbf{r} \rangle$  beschrieben. Zunächst ist nur die Positionskomponente  $\mathbf{p} \in \mathbb{R}^3$  der Pose  $\mathbf{v}$  relevant. Das Fahrzeug bewegt sich auf dem Boden durch die Welt. Jede 3- $D$ -Position des Fahrzeugs kann auf die entsprechende Position auf der Rasterkarte projiziert werden. Die 2- $D$ -Position auf der Rasterkarte wird durch die Funktion  $\rho_G : \mathbb{R}^3 \rightarrow \mathbb{R}^2$  berechnet:

$$\rho_G(\mathbf{p}) = \rho_G \left( (p_x, p_y, p_z)^T \right) = (p_x, p_y)^T .$$

Die projizierte Position kann einer Zelle zugeordnet werden. Die Funktion  $\Phi : \mathbb{R}^2 \rightarrow \mathbb{Z}^2$  bildet die 2- $D$ -Fahrzeugposition auf der Rasterkarte auf die Koordinate der entsprechenden Zelle ab:

$$\Phi(\mathbf{q}) = \left( \left\lfloor \frac{q_x}{g} \right\rfloor, \left\lfloor \frac{q_y}{g} \right\rfloor \right)^T .$$

Es ergibt sich die für einen Punkt  $\mathbf{p}$  die Koordinate  $\mathbf{c}_p$  der zuständigen Zelle:

$$\mathbf{c}_p = \Phi(\rho_G(\mathbf{p})) .$$

Da es Ziel ist, die direkte Umgebung des Fahrzeugs zu modellieren, ist die Verwendung einer infinit definierten Rasterkarte lediglich zu

Modellierungszwecken sinnvoll. Eine Einschränkung auf eine *lokale* Rasterkarte, die um das Fahrzeug herum aufgebaut wird, ist notwendig. Dabei ist es Ziel, dass sich das Fahrzeug jeweils in der Zentrumszelle, d. h. in der Mitte der Rasterkarte befindet. Da sich das Fahrzeug natürlich relativ zum Welt-Koordinatensystem bewegt, wird eine entsprechend spezialisierte Variante der Rasterkarte definiert.

Die Grundüberlegungen hinsichtlich der Welt-Referenz und der Ausrichtung der Rasterkarte bleiben bestehen. Die Zentrumszelle der Rasterkarte muss entsprechend der Fahrzeugposition gewählt werden. Diese muss durch die Rasterkarte adaptiert werden.

Die lokale Rasterkarte um das Fahrzeug wird an der Zentrumszelle ausgerichtet. Die Ausbreitung der Rasterkarte wird durch den Parameter  $e \in \mathbb{N}$  definiert, der die Anzahl Zellen in jede Richtung um die Zentrumszelle beschreibt. Die lokale Rasterkarte enthält folglich  $(2 \cdot e + 1)^2$  Zellen, mit einer *lokalen* Zellindizierung im Bereich  $([-e; e], [-e; e]) \in \mathbb{Z}^2$ . Dies bedeutet, dass die lokale Rasterkarte einen quadratischen Ausschnitt aus der *globalen infiniten* Rasterkarte beschreibt.

Diese Einschränkung ermöglicht die Beschreibung des direkten Fahrzeugumfelds, was der Anforderung der Umfeldvisualisierung genügt. Je nach konkret überwachtem Umfeld ist eine andere geometrische Positionierung der Zentrumszelle innerhalb des lokalen Ausschnitts sinnvoll.

Da sich das Fahrzeug bewegt, muss der lokale Ausschnitt auf die jeweilige Fahrzeugposition angepasst werden. Daher wird die lokale Rasterkarte zu einer *lokalen gleitenden Rasterkarte* (engl. *local shifting grid map*) erweitert. Sei  $\mathbf{q}_\tau \in \mathbb{R}^2$  die 2-D-Referenzposition von  $\mathbf{v}_\tau$  hinsichtlich der Rasterkarte zum Zeitschritt  $\tau$  und  $\mathbf{q}_{\tau+1} \in \mathbb{R}^2$  die Position zum Zeitschritt  $\tau + 1$ . Eine Aktualisierung des Kartenausschnitts der gleitenden lokalen Rasterkarte ist notwendig, wenn  $\Phi(\mathbf{q}_\tau) \neq \Phi(\mathbf{q}_{\tau+1})$ .

Sobald sich die der Fahrzeugposition zugeordnete Zellenkoordinate ändert, muss der lokale Ausschnitt der Karte entsprechend den Rasterdimensionen angepasst werden. Das Prinzip hinter der lokalen gleitenden Rasterkarte ist in Abbildung 5.24 dargestellt.

Die Rasterkarte dient als Datenstruktur zur Abbildung der Umgebungsinformationen. Die relativ zum Welt-Koordinatensystem konstanten Positionen der einzelnen Zellen ermöglichen eine zeitliche Integration der Umgebungsdaten in den Zellen für deren jeweiligen Raumausschnitt. Bei der Anpassungsoperation für eine neue Zentrumszelle müssen die Daten beibehalten werden (engl. *preserved*).

Als Eingabedaten für die geometrische Modellierung der Umgebung werden 3-D-Punktwolken verwendet. Es können beispielsweise Stereoverbünde (siehe Kapitel 5.2.1) oder 3-D-LIDAR-Daten verwendet werden. Die im Folgenden beschriebenen Verarbeitungsschritte hin zur geometrischen Modellierung des Bodens sind unabhängig



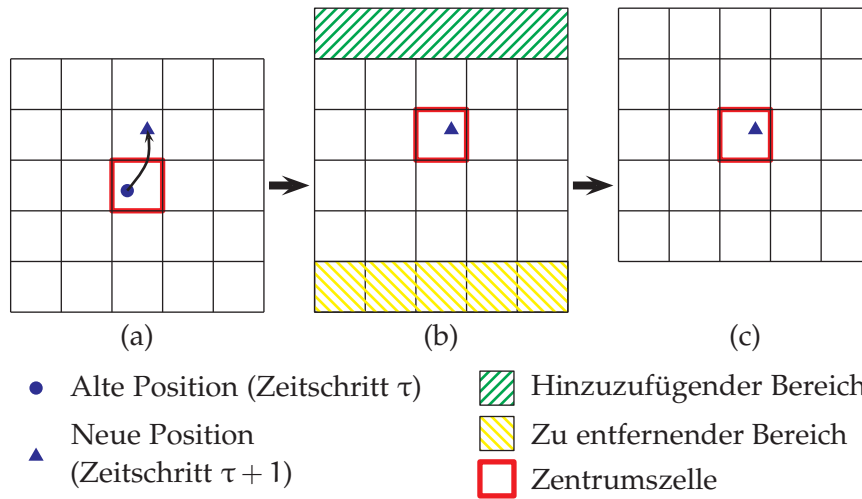


Abbildung 5.24: Anpassung einer lokalen gleitenden Rasterkarte bei neuer Zentrumszelle. Sobald das Fahrzeug die Zentrumszelle verlässt (a), wird die neue Zentrumszelle bestimmt und der zu verkleinernde bzw. zu vergrößernde Bereich ermittelt (b). Der Ausschnitt des Rasters wird so angepasst, dass es zur neuen Zentrumszelle passt (c).

von der gewählten Sensormodalität.

Um die Verständlichkeit der Ausführungen zu gewährleisten, wird im Folgenden die Verarbeitung nur eines Sensors beschrieben. Selbstverständlich ist jedoch die Verwendung mehrerer Sensoren möglich.

Für die Integration der 3-D-Punktwolken, in die bereits beschriebene lokale gleitende Rasterkarte werden zunächst die involvierten Koordinatensysteme beschrieben. Dabei wird die Modellierung der Koordinatensysteme für die Rasterkarte übernommen und mit der in Kapitel 5.3 vorgenommenen Modellbildung für die Vogelperspektivansicht verknüpft. Abbildung 5.25 zeigt das Welt-Koordinatensystem  $w$  und die lokale gleitende Rasterkarte mit der Zentrumszelle  $z$ , der das Fahrzeug  $v$  zugeordnet ist. Im Fahrzeug ist eine Stereokamera verbaut, die mit der Relativpose  $\mathbf{s} \in \mathbb{R}^3 \times \text{SO}(3)$  zum Fahrzeug-Koordinatensystem  $v$  beschrieben wird.

Die Punktmenge  $A_\tau^s$  zum Zeitschritt  $\tau$  beinhaltet die 3-D-Sensordatenpunkte im Sensor-Koordinatensystem  $s$ :

$$A_\tau^s = \{ \mathbf{p}^s \mid \mathbf{p}^s \in \mathbb{R}^3 \} \quad .$$

Um die Punkte mit den Zellen der Rasterkarte in Verbindung zu bringen, müssen sie transformiert werden. Abbildung 5.26 gibt einen Überblick über das vorliegende Szenario. Sei  $\mathbf{v} \in \mathbb{R}^3 \times \text{SO}(3)$  die Relativpose des Fahrzeugs zum Welt-Koordinatensystem. Der Positionvektor  $\mathbf{p}_v \in \mathbb{R}^3$  kann mit der Funktion  $\Phi$  auf eine Zelle in der Rasterkarte abgebildet werden. Dazu ist der Ursprung der jeweiligen

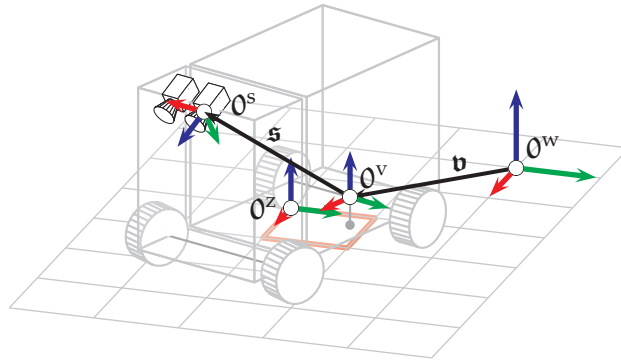


Abbildung 5.25: Welt-Koordinatensystem  $w$ , Fahrzeug-Koordinatensystem  $v$ , Zentrum-Zelle  $z$  und Sensor-Koordinatensystem  $s$ . Die Fahrzeugpose  $\mathbf{v}$  transformiert das Welt in das Fahrzeug-Koordinatensystem. Die Sensorpose  $\mathbf{s}$  definiert das Koordinatensystem für die Sensordaten innerhalb des Fahrzeugs.

Zelle notwendig. Sei  $c$  die korrespondierende Zelle. Der Ursprung  $0^z$  der Zelle  $c$  wird mit der Funktion  $\Omega : \mathbb{R}^3 \rightarrow \mathbb{R}^3$  berechnet:

$$\Omega(\mathbf{p}_v) = ([p_x \cdot g^{-1}] \cdot g, [p_y \cdot g^{-1}] \cdot g, 0)^T .$$

Da es zwischen dem Welt-Koordinatensystem und dem jeweiligen Zell-Koordinatensystem keine Rotation gibt (Ausrichtung entlang der Achsen des Welt-Koordinatensystems), bedingt das Rotationsquaternion der Pose  $\mathbf{v}$  keine weitere Transformation. Die Fahrzeugpose  $\mathbf{v}$  in Referenz zum Zell-Koordinatensystem  $z$  ergibt sich mit:

$$\mathbf{v}^z = \langle \mathbf{p}_v^w - \Omega(\mathbf{p}^w), \mathbf{r}^w \rangle .$$

Die Pose  $\mathbf{s}$  beschreibt die Transformation vom Fahrzeug-Koordinatensystem zum Ursprung des Koordinatensystems der Sensordaten. Für den in Abbildung 5.25 dargestellten Aufbau ist das Kamerazentrum der Stereo-Hauptkamera der Ursprung des Sensor-Koordinatensystems.

Die akkumulierte Pose  $\mathbf{a}$  für Registrierung einer Punktwolke  $A_\tau^s$  kann mit gegebener Zentrumszelle  $c$  der Rasterkarte  $G$  bestimmt werden:

$$\mathbf{a} = \mathbf{v}^z \oplus \mathbf{s} .$$

Die Transformation zu  $A_\tau^z$  wird mit Hilfe der Transformationsmatrix  $T_{s \rightarrow z} = \mathbf{T}(\mathbf{a})$  vollzogen:

$$A_\tau^z = \{T_{s \rightarrow z}(\tilde{\mathbf{p}}^s) \mid \mathbf{p}^s \in A_\tau^s\} .$$

Um die Punkte für die zeitliche Integration in den Zellen zu clustern, wird für jede Zelle eine Menge  $J_{\tau,u,v} \subseteq A_\tau^z$  mit  $(u,v)^T \in \mathbb{Z}^2$  der Zellkoordinaten definiert:

$$J_{\tau,u,v} = \left\{ \mathbf{p}^z \mid \mathbf{p}^z \in A_\tau^z \wedge \left( \Phi(\Omega(\mathbf{p}^w)) = (u,v)^T \right) \right\} .$$

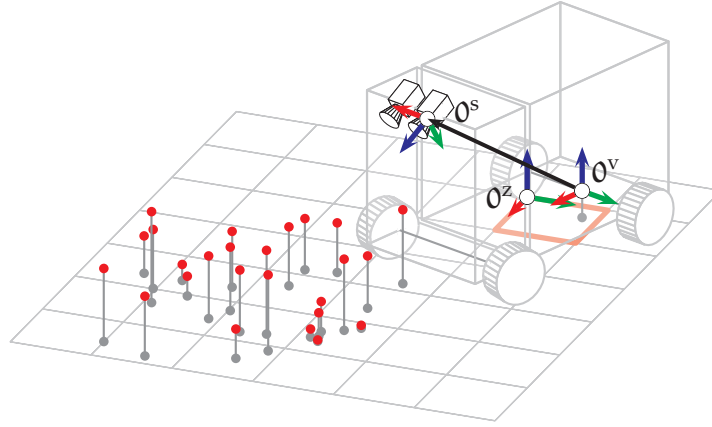


Abbildung 5.26: Abbilden der Punktwolken auf die Raster-Struktur. Jeder Punkt wird der Zelle zugeordnet, zu der die 2-D-Grundprojektion des Punktes gehört.

Die lokale gleitende Rasterkarte wird als Datenstruktur genutzt, in der die beobachteten 3-D-Punkte der entsprechenden Zelle zugeordnet werden. Da die Rasterkarte eine globale Referenzposition hat, ist die Datenstruktur auch für die temporale Integration der Punktwolken geeignet.

Im Folgenden werden nicht nur die 3-D-Positionswerte der aufgenommenen Punkte verwendet. Die Distanz  $d \in \mathbb{R}^+$  des Punktes bei Aufnahme zum Sensor wird als Qualitätskriterium integriert. Dies bedeutet, dass die Zell-Datenstruktur neben dem ursprünglichen Punkt auch die Entfernung mit abbilden muss. Daher wird die Menge der Punkte in einer Zelle auf ein Tupel  $(\mathbf{p}, d) \in \mathbb{R}^3 \times \mathbb{R}^+$  erweitert:

$$A_\tau^s = \{(\mathbf{p}^s, d) \mid \mathbf{p}^s \in \mathbb{R}^3 \wedge d = \|\mathbf{p}^s\|\} \quad .$$

Die nachfolgenden Schritte bleiben gleich, wobei der Distanzparameter  $d$  jeweils beibehalten wird. Die resultierende Punktmenge mit Distanz für eine einzelne Zelle an der Position  $(u, v)^T \in \mathbb{Z}^2$  zum Zeitschritt  $\tau$  ergibt:

$$J_{\tau;u,v} = \left\{ (\mathbf{p}^z, d) \mid (\mathbf{p}^z, d) \in A_\tau^z \wedge \Phi(\Omega(\mathbf{p}^w)) = (u, v)^T \right\} \quad .$$

Da die Sensoren kontinuierlich neue Punktwolken liefern, ist eine Aktualisierung und Verbesserung einer jeden Rasterzelle möglich. Ziel ist es, die Daten über die einzelnen Zeitschritte  $\tau$  zu akkumulieren. Die temporale Integration der Daten erfolgt durch:

$$J_{u,v} := J_{u,v} \cup J_{\tau;u,v} \quad .$$

Selbstverständlich ist es wenig sinnvoll, die Daten ohne Einschränkung über die komplette Laufzeit zu akkumulieren. Für den Fall, dass sich das Fahrzeug bewegt, werden alte Daten, d. h. solche aus

weiter vergangenen Zeitschritten, durch die gleitende lokale Rasterkarte implizit entfernt, da sie sich aus dem Erfassungsbereich der Karte bewegen.

Ein aktives Entfernen der Daten aus der Rasterkarte ist in zwei Szenarien sinnvoll: Für den Fall von dynamischen Hindernissen und für den Fall, dass sich das Fahrzeug für längere Zeit nicht bewegt. Der erste Fall wird durch eine Heuristik hinsichtlich der sich über die Zeit veränderten Varianz in der Höhe der Punkte einer Zelle erreicht. Der zweite Fall kann entweder durch eine obere Grenze der Punktezahlen je Zelle oder durch eine maximale Anzahl an akkumulierten Zeitschritten in der Karte erreicht werden.

### 5.6.2 Geschlossenes Oberflächenmodell

Die lokale gleitende Rasterkarte wird verwendet, um ein Oberflächenmodell für die Bodenebene zu erstellen. Ziel ist es, den Boden mit einer geschlossenen geometrischen Form zu modellieren. Für jede Zelle wird anhand der in ihr befindlichen Daten ein Höhenwert bestimmt. Die lokale gleitende Rasterkarte wird dadurch um eine Höhenkarte ergänzt. Dazu wird ein Ansatz nach Neuhaus et al. [NWW<sup>+</sup>16] adaptiert.

Für die Höhenschätzung werden jeweils die Punktdaten  $J_{u,v}$  einer einzelnen Zelle betrachtet. Pro 3-D-Punkt wird ein Gewicht berechnet, das eine große Entfernung zum Sensor bei der Aufnahme „bestraft“. Je näher ein Punkt zum Sensor bei der Aufnahme ist, umso zuverlässiger wird der gemessene Wert angenommen. Die monoton fallende Funktion  $\chi : \mathbb{R}^+ \rightarrow [0; 1]$  berechnet das Gewicht anhand dem Distanzwert  $d \in \mathbb{R}^+$ . Ein Beispiel einer solchen Gewichtungsfunktion ist in Abbildung 5.30 zu sehen.

Das iterative Verfahren für die Schätzung der Höhe für eine Zelle ist in Algorithmus 5.1 ersichtlich. Zellen mit einer hohen Höhenvarianz innerhalb der zugeordneten Punktwolke werden als *Hindernis* klassifiziert. Das folgende Vorgehen bezieht sich primär auf *Boden*-Zellen. Der Umgang mit *Hindernis*-Zellen wird zu einem späteren Zeitpunkt genauer beleuchtet (siehe Kapitel 5.7).

Beschreibe  $\omega : \mathbb{Z}^2 \rightarrow \mathbb{R}$  das Berechnungsergebnis der in Algorithmus 5.1 beschriebenen Funktion.

Mit den für die Zellen ermittelten Höhen wird das geschlossene Oberflächenmodell berechnet. Ein Oberflächenmodell ist dabei als die Beschreibung der Oberfläche des erfassbaren Umfelds mit geometrischen Primitiven zu verstehen. Die geometrischen Primitive werden im nächsten Schritt als Grundlage für die Erzeugung der perspektivisch korrekten, dichten Ansicht aus der Vogelperspektive genutzt.

Dazu wird die geometrische Anordnung der Zellen der Rasterkarte verwendet. Der Höhenwert wird zwischen vier benachbarten Zellen gemittelt. Dadurch wird eine in der Ebene äquidistant angeordnete

---

**Algorithmus 5.1** Höhengschätzung für eine Zelle (angelehnt an [NWW<sup>+</sup>16])

---

```

1:  $\sum_h \leftarrow 0$      $\sum_w \leftarrow 0$ 
2:  $\Gamma \leftarrow \{(z,d) \mid (\mathbf{p},d) \in J_{i,j}, z = p_z\}$ 
3:  $n \leftarrow |\Gamma|$      $\hat{h} \leftarrow \Gamma[1].z$ 
4: sort tuples in  $\Gamma$  ascending by  $z$ 
5: for  $i := 1$  to  $n$  do
6:    $h_i \leftarrow \Gamma[i].z$             $\triangleright$  Höhe relativ zur Rasterkarte
7:    $w_i \leftarrow \chi(\Gamma[i].d)$         $\triangleright$  Gewichtungsfaktor
8:   if  $i > 0.5 \cdot n$  and  $h - \hat{h} > k$  then    $\triangleright k$  ein Schwellwert
9:     break
10:   $\sum_h \leftarrow \sum_h + w_i \cdot h_i$ 
11:   $\sum_w \leftarrow \sum_w + w_i$ 
12:   $\hat{h} \leftarrow \sum_h \cdot (\sum_w)^{-1}$ 
return  $\hat{h}$ 

```

---

te Menge von Oberflächenpunkten erzeugt. Die Höhenwerte an den Eckpunkten der Zellen werden interpoliert, sodass ein 3-D-Punkt für die geschlossene Oberfläche resultiert. Der Oberflächenpunkt am Ursprung der Zelle mit der Rasterkoordinate  $(u,v)$  wird durch die Funktion  $\mathbf{y} : \mathbb{Z}^2 \rightarrow \mathbb{R}^3$  definiert:

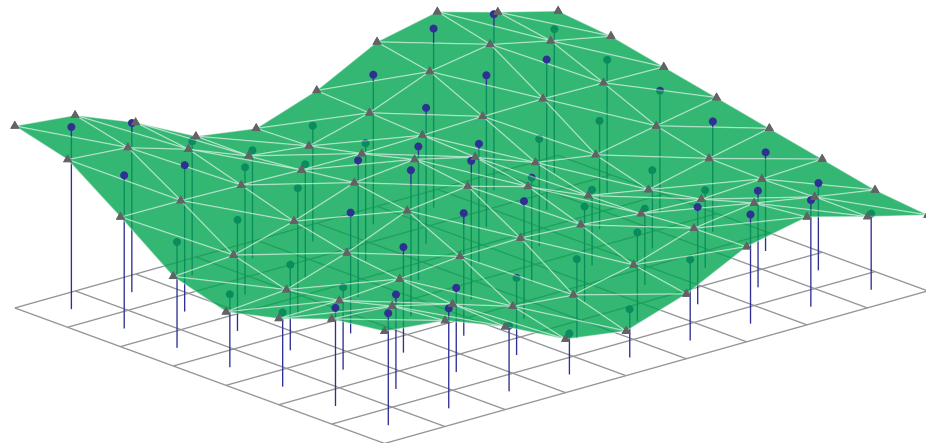
$$\mathbf{y}(u,v) = \left( g \cdot u, g \cdot v, \frac{1}{4} \sum_{i=-1}^0 \sum_{j=-1}^0 \omega(u+i, v+j) \right)^T .$$

Jede (quadratische) Zelle  $(u,v)$  verfügt über vier Eckpunkte, die nun mit geometrischen Primitiven approximiert werden. Um die Geschlossenheitsbedingung zu erfüllen, wird die Oberfläche jeder Zelle durch zwei Dreiecke modelliert:

$$\begin{aligned} \Delta_1(u,v) &= \{\mathbf{y}(u,v), \mathbf{y}(u+1,v), \mathbf{y}(u,v+1)\} \\ \Delta_2(u,v) &= \{\mathbf{y}(u+1,v+1), \mathbf{y}(u,v+1), \mathbf{y}(u+1,v)\} . \end{aligned}$$

Die finale geschlossene Oberfläche für die Daten in der lokalen gleitenden Rasterkarte bestehen aus der Menge aller nach dem vorgestellten Verfahren berechneten Dreiecke. Die Dreiecke überspannen die gesamte Rasterkarte bzw. die als *Boden* klassifizierten Zellen. Abbildung 5.27 visualisiert das Prinzip des entwickelten Ansatzes. Der Glättungseffekt durch die Mittelung der benachbarten Zellen ist dort deutlich zu erkennen.

Modellbasiert bedingt die Mittelung der Zellen eine Glättung der tatsächlich gemessenen Höhen. Dies führt unweigerlich zu einer leichten Abweichung in der Abbildung der Umgebung, ist aber eine Notwendigkeit im Hinblick auf die lokale Planarität. Der Effekt ist dabei abhängig von der Zellengröße  $g$ . Je kleiner der Parameter  $g$  gewählt wird, desto geringer ist dieser Effekt. Die geschlossene Oberfläche ist folglich eine Approximierung der Umgebungsgeometrie.



- Zellenhöhen (berechnet durch Algorithmus 5.1)
- ▲ Punkte der geschlossenen Oberfläche  $y$
- ▲ Planares Oberflächenelement ( $\Delta_1/\Delta_2$ )

Abbildung 5.27: Geschlossene Oberflächenkarte. Die Zellenhöhen werden für die Berechnung der geschlossenen Oberfläche verwendet. Die Oberfläche ist aufgrund der verwendeten Dreiecke partiell planar.

Durch die vorab vorgenommene Klassifizierung der einzelnen Zellen wird bereits eine Vorselektion im Hinblick auf die Berechnung des Oberflächenmodells vorgenommen. Dadurch, dass Zellen mit sehr hoher Varianz in der Höhe der Punkte als *Hindernis* aussortiert werden (siehe Algorithmus 5.1), können sie keine Deformierung der Oberfläche verursachen. Liegt ein Hindernis in einer Zelle, ist es unwahrscheinlich, dass es die komplette Zelle abdeckt. Dies bedeutet, dass ein potentieller Bodenbereich innerhalb der Zelle durch die Klassifizierung der *gesamten* Zelle dazu führen kann, dass dieser partielle Bodenbereich nicht mit in die Oberfläche einbezogen wird. Dieser Problemstellung kann durch eine gesonderte Betrachtung entsprechender Kandidatenzellen begegnet werden (siehe Kapitel 5.7).

### 5.6.3 Partielle Homographien

Ein Hauptproblem der Methoden für die Berechnung virtueller Vogelperspektiven nach dem Stand der Technik ist die unnatürliche Verzerrung in Bereichen, die die Bodenannahme verletzen (siehe Kapitel 5.4). Nichtsdestotrotz liefern diese Verfahren gute Ergebnisse für Bereiche, in denen die Bodenannahme erfüllt ist. Mit dem Ansatz zur Berechnung der geschlossenen Oberfläche wird die vorgefundene Geometrie in partiell planare Teilbereiche unterteilt. Zudem wird die Geometrie fortlaufend durch neue 3-D-Daten erweitert, verfeinert und aktualisiert, während sich das Fahrzeug fortbewegt (bzw. steht).

Da die Oberflächenmodellierung alleine noch keine Visualisierung der Umgebung ermöglicht, werden im nächsten Schritt Kamerabilder integriert. Die Kamerabilder können dabei aus verschiedenen Quellen stammen. Beispielsweise können dieselben Stereokameras als Bildquelle verwendet werden, die bereits für die Disparitätenschätzung und damit für die Schätzung der 3-D-Punktwolke genutzt werden. Andere Kameras können ebenso als Bildquelle verwendet werden. Generell ist es möglich, eine beliebige Anzahl an Kameras einzusetzen.

Sei eine Kamera mit bekannter Intrinsic  $\mathcal{J}$  und bekannter Pose  $\mathbf{c} \in \mathbb{R}^3 \times \text{SO}(3)$  relativ zum Fahrzeug gegeben. Für die im Folgenden betrachteten Schritte werden die zur Beschreibung der Kameraabbildungseigenschaften notwendigen Verzerrungsparameter ignoriert, um die Darstellung verständlicher zu machen. Selbstverständlich müssen diese aber bei der Umsetzung mit bekannten Verfahren (siehe Kapitel 3.2) berücksichtigt werden.

Ziel ist es, das vorab berechnete Oberflächenmodell mit Bilddaten perspektivisch korrekt zu bestücken. Dazu muss das Oberflächenmodell um eine Texturschicht  $\mathcal{T}$  erweitert werden, in der die Bildinformationen akkumuliert werden. Die Texturschicht  $\mathcal{T}$  speichert Bilddaten und entspricht im Wesentlichen einer digitalen Bilddatenstruktur. Die Dimensionen der Texturschicht werden auf die Dimensionen des Ausschnitts der lokalen gleitenden Rasterkarte angepasst, die der Oberflächenmodellierung zugrunde liegt (siehe Kapitel 5.6.1). Das Gleitverhalten der Texturschicht ist analog zu dem Verhalten der lokalen gleitenden Rasterkarte.

Mit dem Parameter  $r \in \mathbb{N}$  wird die Texturauflösung einer einzelnen Zelle  $\mathbf{c}$  definiert. Da die Zellen quadratisch sind, wird dadurch für eine Zelle ein  $r \times r$  großer Ausschnitt in der Texturschicht vorgesehen. In Kombination mit den Dimensionen der lokalen gleitenden Rasterkarte ergibt sich die Gesamtauflösung der Texturschicht:

$$r \cdot (2 \cdot e + 1) \text{ px} \times r \cdot (2 \cdot e + 1) \text{ px} \quad .$$

Durch die Verknüpfung der Texturschicht  $\mathcal{T}$  mit dem geschlossenen Oberflächenmodell ist es möglich, jeder Zelle einen eindeutigen Bereich in der Textur zuzuordnen. Das Befüllen der Texturschicht nutzt die geometrische Modellierung.

Gegeben die 3-D-Koordinaten eines Dreiecks  $\Delta$  des geschlossenen Oberflächenmodells. Die Projektion des Dreiecks in das Kamerabild der Kamera  $C$  wird mit deren Intrinsic-Matrix  $\mathbf{K}_C$  beschrieben. Um die Projektion zu berechnen, müssen die Punkte zunächst in das Koordinatensystem der Kamera  $C$  transformiert werden. Dies erfolgt mittels der akkumulierten Pose  $\mathbf{v}^z \oplus \mathbf{c}$ . Es ergibt sich ein Oberflächen-dreieck im Koordinatensystem der Kamera  $C$ :

$$\Delta^c = \{\mathbf{p}_1^c, \mathbf{p}_2^c, \mathbf{p}_3^c\} \quad .$$

Die korrespondierenden 2-*D*-Bildpunkte auf der Bildebene der Kamera *C* werden berechnet durch:

$$\Delta^P = \{q_i \mid q_i \in \mathbb{R}^2, i \in \{1, 2, 3\}\} = \{K_C \cdot p_i^c \mid p_i^c \in \Delta^c, i \in \{1, 2, 3\}\} \quad .$$

Die drei Bildpunkte in  $\Delta^P$  markieren den Bereich im Bild, der den für das Dreieck korrekten Bildbereich beinhaltet. Dieser Bildbereich beschreibt den Texturausschnitt, der durch das Kamerabild für das Oberflächenelement gesetzt bzw. integriert werden kann. Der Bereich ist jedoch durch die Projektion verzerrt und entspricht nicht dem für das Einsetzen in die Textur notwendigen Format. Es muss folglich eine Transformation erfolgen, die den Bildausschnitt korrekt auf den Texturbereich abbildet.

Da es sich in der 3-*D*-Modellierung um ein lokal planares Oberflächenelement handelt, kann an dieser Stelle eine Homographie eingesetzt werden, die genau diese Grundannahme beinhaltet. Der Homographie-Verschattungseffekt (siehe Kapitel 5.4) stellt an dieser Stelle kein Problem dar, da die Annahme der Planarität (hinreichend) erfüllt ist. Um das Punktkorrespondenzproblem zu formulieren, sind jedoch mindestens vier Punkte erforderlich. Der vierte Punkt wird durch einen virtuellen Punkt repräsentiert, der durch eine Linearkombination der 3-*D*-Dreieckspunkte beschrieben wird:

$$p_4^c = p_2^c + p_3^c - p_1^c \quad \Rightarrow \quad \square^c = \Delta^c \cup \{p_4^c\} \quad .$$

Die Linearkombination expandiert das Dreieck zu einem Quadrat im 3-*D*-Raum auf derselben Ebene. Die Projektion des Punktes  $p_4^c$  auf die Bildebene der Kamera *C* erfolgt analog:

$$\square^P = \Delta^P \cup \{K_C \cdot p_4^c\} \quad .$$

Die für die Homographie notwendigen Punktkorrespondenzen können direkt auf den zur Oberfläche verbundenen Texturkoordinaten formuliert werden. Die Texturfläche, die einer einzelnen Rasterzelle *c* zugeordnet wird, ist quadratisch ( $r \times r$ ). Dadurch lässt sich eine lokale Texturcoordinate innerhalb des Texturausschnitts definieren, die im Bereich  $[0; 1] \times [0; 1]$  liegt. Da die Dreiecke respektive die expandierten Quadrate auf jeweils ganze Texturausschnitte abgebildet werden, können den Punkten in  $\square^P$  die Eckkoordinaten des Texturausschnitts zugeordnet werden.

Diese Zuordnung lässt sich im Hinblick auf die Gesamttextur erweitern: Beschreibe die Rastercoordinate  $(u, v) \in \mathbb{Z}^2$  die Position der jeweiligen betrachteten Zelle. Der Ursprung des Bereichs, der der Zelle zugeordnet ist, befindet sich in der Textur  $\mathcal{T}$  an der Texturkoordinaten  $t_{u,v} \in \mathbb{N}^2$ :

$$t_{u,v} = r \cdot (u, v)^T + (e, e)^T \quad .$$



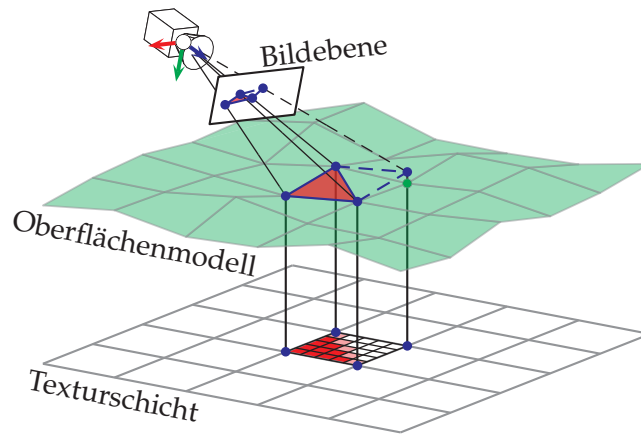


Abbildung 5.28: Prinzip der partiellen Homographie-Transformation anhand einer geschlossenen Oberflächenmodellierung. Die 3-D-Punkte werden in das Kamerabild transformiert, um die korrekte Region auszuwählen. Der Bildinhalt wird dann auf die Ebene der Texturschicht transformiert.

Die Homographie-Matrix  $\mathbf{H}_i \in \mathbb{P}^{2 \times 2}$  für die Transformation zwischen dem  $i$ -ten Oberflächenelement  $\square_i^p$  und den korrespondierenden Texturkoordinaten wird mit einem Verfahren wie dem in Kapitel 3.3.1 beschriebenen berechnet:

Die Homographie-Matrix wird durch Multiplikation auf die Bildpunkte des Kamerabildes angewandt. Um die korrekte 2-D-Koordinate zu erhalten, ist eine perspektivische Division notwendig. Die beiden Schritte werden in der Funktion  $\mathbf{hom}_{\mathbf{H}} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$  zusammengefasst:

$$\mathbf{hom}_{\mathbf{H}}(\mathbf{q}) = (a_x \cdot a_z^{-1}, a_y \cdot a_z^{-1})^T \text{ mit } \mathbf{a} = (a_x, a_y, a_z)^T = \mathbf{H} \cdot \tilde{\mathbf{q}} \quad .$$

Der berechnete quadratische Bildausschnitt ist aufgrund der Extrapolation des vierten Referenzpunktes  $\mathbf{p}_4^c$  zu groß. Für das Dreieck ist daher nur die Hälfte des Zielbereichs relevant, die zur ursprünglichen Dreiecksfläche korreliert. An der Diagonalen, an der sich die beiden Dreiecke treffen, ist eine Fusion der beiden Berechnungsergebnisse (Überblendung) notwendig.

Die Texturschicht  $\mathcal{T}$  wird für jedes Oberflächenelement jeweils partiell aktualisiert, sofern das Element im Kamerabild von  $C$  sichtbar ist. Da im vorgestellten neu entwickelten Verfahren jeweils lokale Homographien berechnet werden, ist es ein *partielles Homographie-Verfahren*.

Selbstverständlich muss eine Sichtbarkeitsbedingung bei der Abbildung in die Texturschicht berücksichtigt werden. Bei der Projektion in das Kamerabild muss geprüft werden, ob das jeweilige Dreieck aus der jeweiligen Kamera sichtbar ist um die Übernahme von falschen Bildausschnitten zu vermeiden. Dabei werden auch die als *Hindernis* klassifizierten Zellen berücksichtigt, damit keine falsche Textur

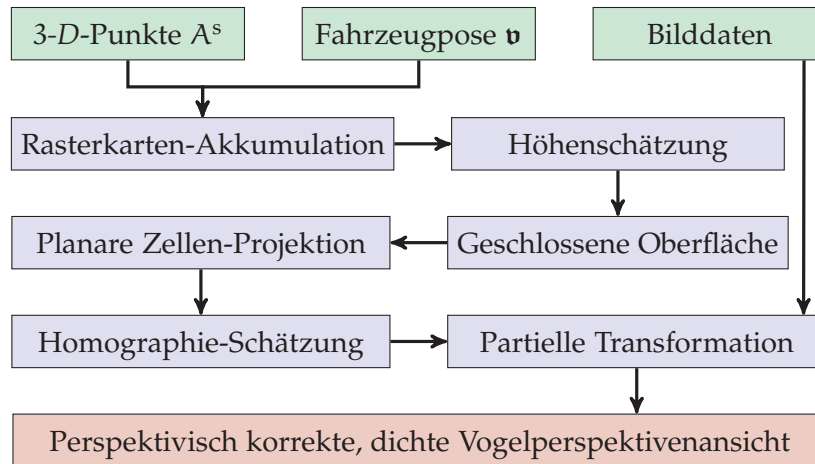


Abbildung 5.29: Überblick über die Schritte für die partielle Homographie-Transformation.

auf einen hinter einem Hindernis liegenden, aus Sicht der Kamera verdeckten Bereich übernommen wird. Um eine entsprechende Schnitttest-Geometrie zu erzeugen, werden die *Hindernis*-Zellen daher durch Quader in ihrer Ausdehnung beschrieben. Die Höhengrenzen werden dabei aus den Zell-Punktewolken heuristisch ermittelt.

Der gesamte Prozess wird in Abbildung 5.28 dargestellt. Für ein Dreieck in der 3-D-Modellierung wird ein vierter 3-D-Punkt extrapoliert. Das resultierende Quadrat wird auf die Bildebene projiziert, damit der für das Dreieck passende Bildausschnitt ausgewählt werden kann. Der ausgewählte Bereich wird mittels Homographie in den entsprechenden Bereich der Texturschicht transformiert.

Die Texturschicht als solche entspricht modellbedingt einer (approximierten) orthographischen Vogelperspektive. In Kombination mit der Oberflächenmodellierung können andere Kameraansichten (perspektivisch und orthographisch) berechnet werden, indem eine 3-D-Ansicht des texturierten Oberflächenmodells erzeugt wird.

Die Oberflächentextur enthält in den Bereichen der Rasterkarte, die durch das oben beschriebene Verfahren erfasst werden, entsprechende Texturinformationen. *Hindernis*-Zellen, die über keine Texturinformation verfügen, können entsprechend markiert werden.

Ein Überblick über die einzelnen Bausteine für die Berechnung der dichten, perspektivisch korrekten Ansicht aus der Vogelperspektive ist in Abbildung 5.29 gegeben.

#### 5.6.4 Ergebnisse

Zum Test des Verfahrens werden die *KITTI Odometry Datasets* [GLU12] verwendet. Dieser öffentlich verfügbare Datensatz enthält u. a. Aufnahmen eines Grauwert-Stereo-Paares, eines Farb-Stereo-Paares sowie eines Velodyne 3-D-LIDAR. Der Datensatz enthält (wie

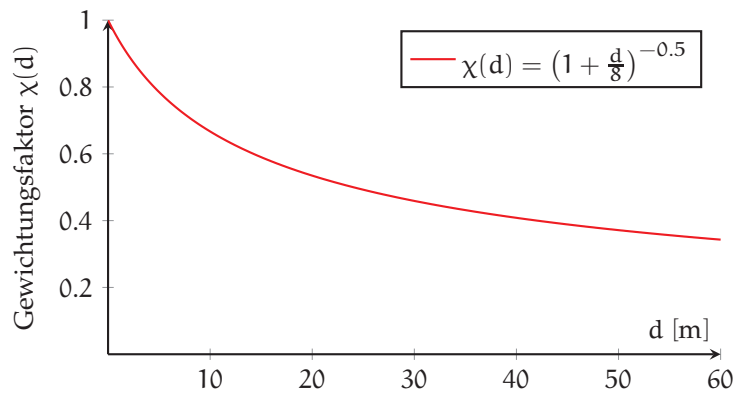


Abbildung 5.30: Gewichtungsfunktion für den Einfluss eines Punktes auf die Höenschätzung, die von der Entfernung des Punktes zum Sensor zum Zeitpunkt seiner Aufnahme abhängt

auch der *Daimler Urban Segmentation Dataset* [SERF14], Kapitel 5.5) lediglich Kamera-Ansichten, die in Fahrtrichtung blickend aufgenommen wurden. Nichtsdestotrotz ist der Datensatz gut geeignet, um das Prinzip hinter der vorgestellten Methodik zu zeigen.

Für die Ergebnisbilder wurden folgende Parameter gewählt:

- $g = 3,5^{-1} \text{ m}$   
Die Seitenlänge einer Zelle wird auf  $\approx 28,6 \text{ cm}$  festgelegt.
- $e = 30$   
Die lokale gleitende Rasterkarte besteht aus  $61 \times 61$  Zellen, d.h. sie deckt eine Fläche von ca.  $303,8 \text{ m}^2$  ab.
- $r = 16 \text{ px}$   
Eine Texturauflösung von  $16 \text{ px}$  bei gegebener Zellgröße entspricht einer Fläche ca.  $1,7 \text{ cm} \times 1,7 \text{ cm}$  pro Pixel ( $\approx 2,89 \text{ cm}^2$ ) in der Texturebene.
- $\chi(d) = (1 + \frac{1}{8} \cdot d)^{-0.5}$   
Der Werteverlauf der Gewichtungsfunktion ist in Abbildung 5.30 dargestellt.

Die Abbildungen 5.31, 5.32, 5.33 und 5.34 zeigen Ergebnisse unter Verwendung unterschiedlicher Datensätze. Die temporale Integration der 3-D-Daten ermöglicht eine präzise Modellierung der Fahrzeugumgebung und die Detektion von Hindernissen.

In Abbildung 5.31 ist die Korrektheit der Projektion klar ersichtlich wie z. B. an den Pflastersteinen. Hindernisse, wie Zäune, Wände oder andere Fahrzeuge, werden durch den heuristischen Ansatz erkannt und korrekt lokalisiert. Die Ausrichtung der Kameras in Fahrtrichtung (Kamera-Frustren in Abbildung 5.31b) in Kombination mit der Geradeausfahrt bedingt jedoch, dass nicht alle Bereiche aufgrund von

Verdeckungen mit Bilddaten gefüllt werden können. Dies ist in Abbildung 5.31a im unteren Bereich an den Pflastersteinen zu sehen, die links eine diagonale Begrenzung in der Sichtbarkeit aufweisen. Wenn weitere Kameras in diesen Aufbau integriert werden, können auch diese Bereiche mit Bilddaten gefüllt werden.

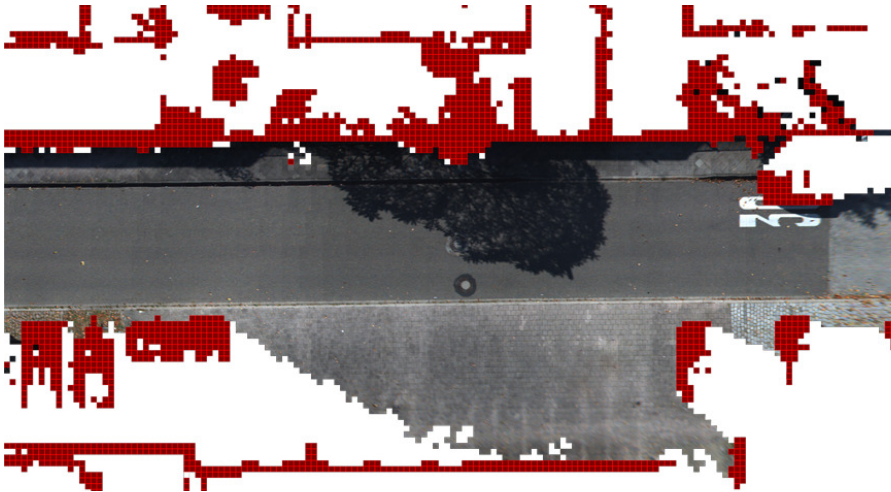
Abbildung 5.32 zeigt die Vogelperspektive auf eine Straßenkreuzung. Zur Interpretation des dargestellten Ergebnisses ist die Kenntnis des gefahrenen Wegs wichtig. In Referenz zu Abbildung 5.32a ist das Fahrzeug von rechts kommend an der Kreuzung nach rechts abgebogen. Die Ansichten entsprechen der in Abbildung 5.32c dargestellten Position. Bereits von der Position in der Mitte der Kreuzung wird der Bereich großflächig erfasst. Ein Effekt hinsichtlich der Texturqualität lässt sich in diesem Beispiel gut erkennen: Die Geschwindigkeitslimitierung „30“ ist rechts deutlich schärfer dargestellt als oben (5.32a). Während der Vorbeifahrt können qualitativ bessere Bilder mit höherer lokaler Auflösung von der rechten „30“ aufgenommen werden. Für die „30“ oben liegen aktuell nur Bilder aus größerer Entfernung vor. Sobald sich das Fahrzeug weiter nähert, werden diese Bildbereiche aktualisiert, sodass auch diese „30“ schärfer wird, sobald sich das Fahrzeug nähert.

In Abbildung 5.33 ist ein Szenario zu sehen, in dem sich ein Fahrzeug auf einem befestigten Weg durch unebenes Grasland bewegt. Die umliegende Vegetation wie Büsche oder Bäume bzw. Baumstämme wird korrekt als Hindernis klassifiziert sowie einige unebene Stellen in der Böschung. Die Wiese wird korrekt durch das Oberflächenmodell modelliert (5.33b) und texturiert. Fehlende kleine Bereiche/individuelle Zellen sind durch die Selbstverdeckung des Terrains aus Sicht der Kamera zu erklären.

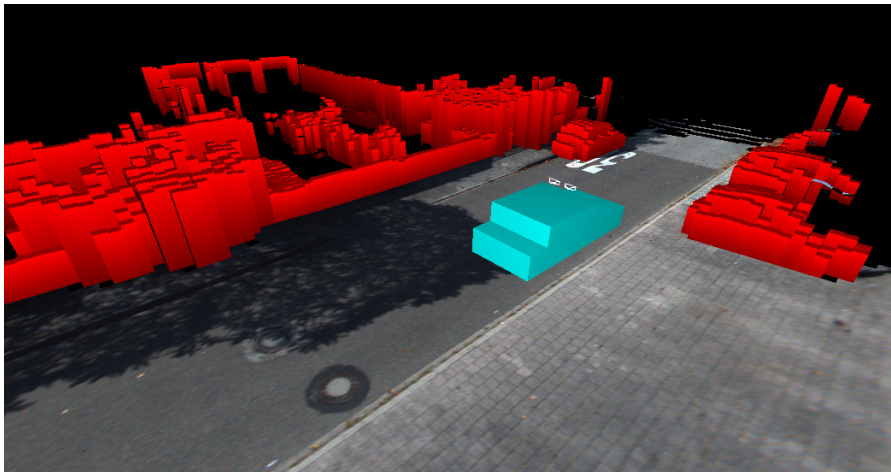
Ein hybrides Szenario ist in Abbildung 5.34 dargestellt. Das Fahrzeug fährt auf einer Straße, die links von Ackerland und rechts durch Bebauung begrenzt ist. Die Mauern bzw. die dichten Hecken auf der rechten Seite werden korrekt als Hindernisse klassifiziert. Das unebene Terrain auf der linken Seite wird entsprechend seiner Steigung durch das Oberflächenmodell abgebildet. Die Pflanzen auf dem Feld werden als Hindernis klassifiziert. Die erreichte Abbildungsqualität lässt sich anhand der Teernähte, des Gullideckels und der Geschwindigkeitsmarkierung verdeutlichen.

Der vorgestellte Ansatz zeigt eine dem in Kapitel 5.5 vorgestellten Verfahren überlegende Ergebnisqualität, wenngleich eine quantitative Evaluation aufgrund fehlender Referenzdaten nicht möglich ist.

Die Oberflächenmodellierung in Kombination mit der temporalen Integration ermöglicht eine dichte Bodendarstellung. Die erreichte partielle Planarität des Modells ermöglicht das Aufstellen von Homographien für die einzelnen Teilbereiche und vereint damit die Vorteile des klassischen Verfahrens mit einer Homographie pro Bild (siehe Kapitel 5.2.2) und perspektivischer Korrektheit.



(a) Texturschicht/Orthographische Vogelperspektive.  
Rote Bereiche markieren Hindernis-Zellen.

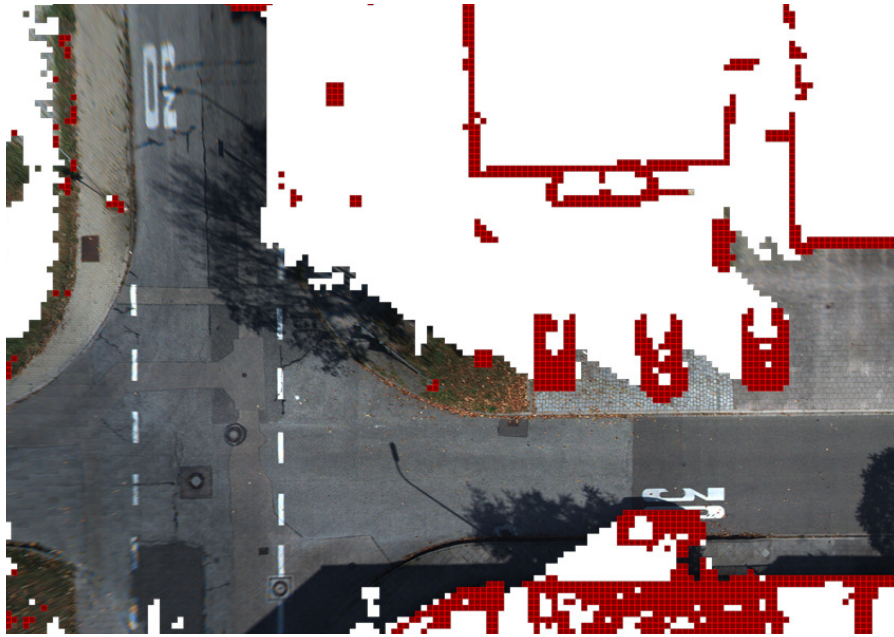


(b) 3-D-Ansicht mit Fahrzeugposition und -visualisierung.  
Die roten Säulen markieren Hindernis-Zellen.  
Die beiden Kamera-Frustren über dem Fahrzeug markieren die Kamera-  
Posen.

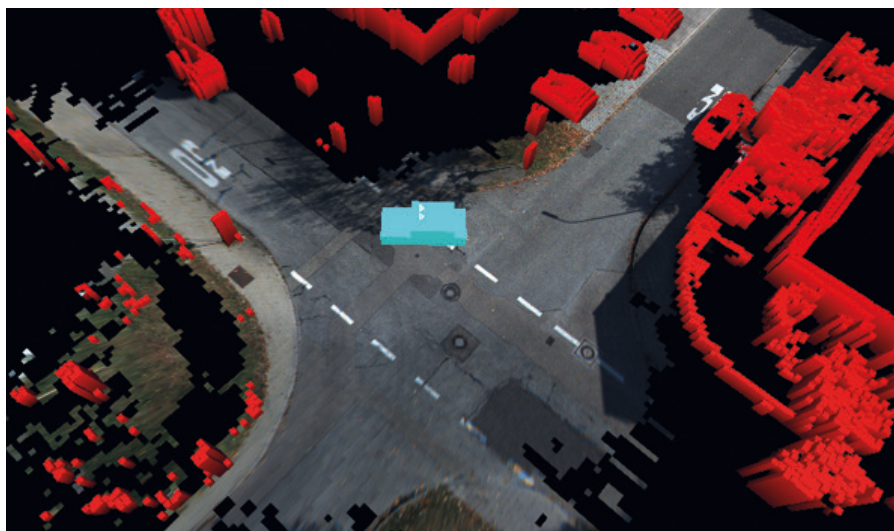


(c) Szenenbild (Datensatz #16, Bild 150 [GLU12]).

Abbildung 5.31: Perspektivisch korrekte dichte Vogelperspektive,  
KITTI Odometry Datensatz #16.



(a) Texturschicht/Orthographische Vogelperspektive.

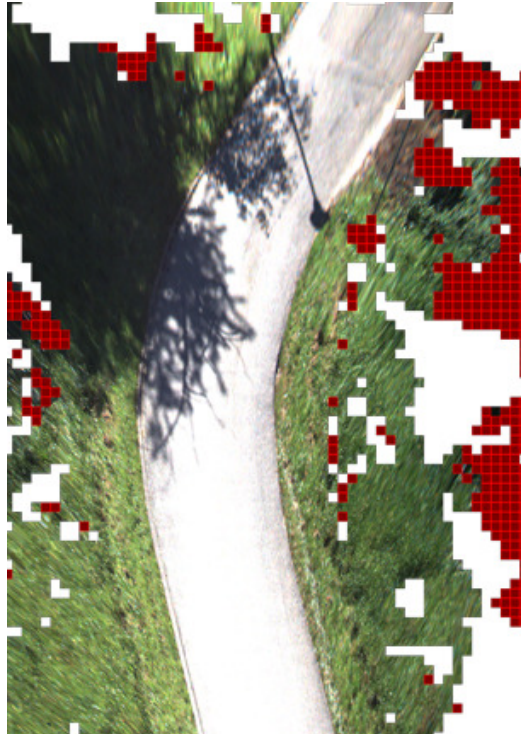


(b) 3-D-Ansicht mit Fahrzeugposition und -visualisierung.

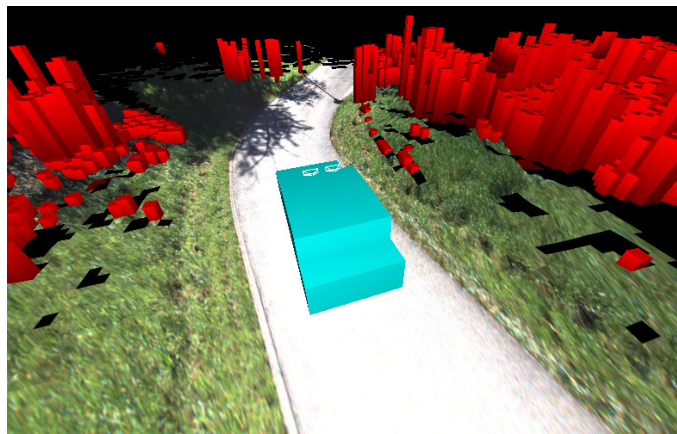


(c) Szenenbild (Datensatz #16, Bild 170 [GLU12])

Abbildung 5.32: Perspektivisch korrekte dichte Vogelperspektive, KITTI Odometry Dataset #16.



(a) Texturschicht/Orthographische Vogelperspektive.



(b) 3-D-Ansicht mit Fahrzeugposition und -visualisierung.

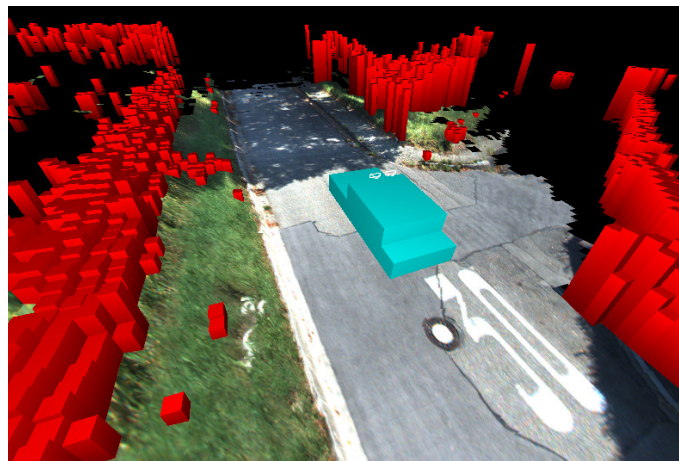


(c) Szenenbild (Datensatz #10, Bild 33 [GLU12])

Abbildung 5.33: Perspektivisch korrekte dichte Vogelperspektive, KITTI Odometry Dataset #10.



(a) Texturschicht/Orthographische Vogelperspektive.



(b) 3-D-Ansicht mit Fahrzeugposition und -visualisierung.



(c) Szenenbild (Datensatz #3, Bild 503 [GLU12])

Abbildung 5.34: Perspektivisch korrekte dichte Vogelperspektive, KITTI Odometry Dataset #3.



Um den darstellbaren Bereich zu maximieren ist eine erweiterte Betrachtung von geeigneten *Hindernis*-Zellen sinnvoll. Weiterhin kann eine genauere Detektion von Hindernissen und deren Darstellung in der Vogelperspektivenansicht einen weiteren Mehrwert generieren.

#### 5.7 VERBESSERTE BODENAPPROXIMATION

Das in Kapitel 5.6.3 vorgestellte Verfahren nähert die Bodengeometrie mit einer lokalen gleitenden Rasterkarte an. Dabei wird je Rasterzelle bestimmt, ob sie zum Boden gehört. Die für jede Zelle – mit hinreichender Anzahl zugeordneter Punkte (Schwellwert) – einzeln getroffene binäre Entscheidung wird im folgenden Abschnitt als *Hindernis* bzw. *Boden* bezeichnet, um die Begrifflichkeit einfach zu halten. Selbstverständlich beschreibt eine als *Boden* klassifizierte Zelle nicht notwendigerweise einen Bodenausschnitt, sondern vielmehr einen durch die Oberflächenmodellierung (geschlossene Oberfläche) beschreibbaren Ausschnitt, der mit dem vorgestellten Verfahren texturiert werden kann. Für Zellen, die mangels Daten nicht klassifiziert werden können, ist eine Rückweisungsklasse (*Unbekannt*) vorgesehen.

Je nach Entscheidung wird die Zelle dann für die Berechnung der virtuellen und perspektivisch korrekten Vogelansicht herangezogen. Die Zellen haben in der vorgenommenen Modellierung eine quadratische Größe und beschreiben einen entsprechenden Ausschnitt des Bodens. Wird eine solche Zelle nur partiell durch ein Hindernis bzw. ein Objekt verdeckt und als *Hindernis* klassifiziert, ist auch der Teil der Zelle betroffen, der noch zum Boden gehört. Im Bereich der Fahrzeugnavigation ist es insbesondere in Rangierszenarien oftmals wichtig, den zur Verfügung stehenden Platz optimal auszunutzen. Es ist daher erforderlich, die geometrische Präzision bei der Oberflächenmodellierung zu erhöhen.

Auf Basis der vorgestellten Bodenklassifikation wird daher eine Erweiterung realisiert, die als *Hindernis* klassifizierte Zellen der lokalen gleitenden Rasterkarte iterativ verfeinert, um den Bereich der Bodenabdeckung für die perspektivisch korrekte Vogelansicht zu maximieren. Ziel ist es, auch *Hindernis*-Zellen soweit als möglich korrekt mit einer Bodentextur zu füllen.

Sei *obstacle* ein Prädikat bzw. eine boolesche Funktion, die ausdrückt, ob eine Zelle *c* als *Hindernis* klassifiziert wird. Analog beschreibt die Funktion *ground*, ob eine Zelle *c* als *Boden* klassifiziert wird. Es gilt:

$$\text{obstacle}(c) = \begin{cases} 1 & c \text{ ist } \textit{Hindernis}, \\ 0 & \text{sonst.} \end{cases}$$

$$\text{ground}(c) = \begin{cases} 1 & c \text{ ist } \textit{Boden}, \\ 0 & \text{sonst.} \end{cases}$$

Da es das Ziel ist, den Bereich um die als Boden klassifizierten Zellen herum zu *erweitern*, ist es naheliegend, nur diejenigen Zellen zu betrachten, die an *Boden*-Zellen angrenzen. Dazu wird eine Nachbarschaftsrelation auf der lokalen gleitenden Rasterkarte definiert, die die Zellen beschreibt, die eine gemeinsame Kante mit der Referenzzelle haben. Im Fall des orthogonalen äquidistanten Rasters entspricht die verwendete Nachbarschaftsrelation  $\nu$  einer Vierernachbarschaft. Für die Zelle  $\mathbf{c}_{u,v}$  an der Position  $(u,v) \in \mathbb{Z}^2$  gilt:

$$\nu(u,v) = \{\mathbf{c}_{u-1,v}, \mathbf{c}_{u,v+1}, \mathbf{c}_{u+1,v}, \mathbf{c}_{u,v-1}\} .$$

Die Menge der *Hindernis*-Zellen, die potenziell für die Erweiterung der Bodenfläche herangezogen werden können, sei mit  $\Pi$  bezeichnet:

$$\Pi = \left\{ \mathbf{c}_{i,j} \mid \text{obstacle}(\mathbf{c}_{i,j}) \wedge \left( \bigvee_{\mathbf{a} \in \nu(i,j)} \text{ground}(\mathbf{a}) \right) \right\} .$$

Die zur Kandidatenzelle  $\mathbf{c}_{u,v}$  gehörende Zell-Punktewolke  $J_{u,v}$  wird in lokale Zellkoordinaten in Bezug auf die  $x$ - und  $y$ -Komponenten in den Bereich  $[0; 1]$  normalisiert. Bei diesem Schritt bleibt jedoch die Höhe des Punktes ( $z$ -Komponente) unverändert. Als normalisierte Zell-Punktewolke  $O_{u,v}$  ergibt sich:

$$O_{u,v} = \left\{ \left( \left( p_x^z \cdot g^{-1}, p_y^z \cdot g^{-1}, p_z^z \right)^T, d \right) \mid (\mathbf{p}, d) \in J_{u,v} \right\} .$$

Sei die Hindernis-Zelle  $\mathbf{c}_{u,v} \in \Pi$  ein Kandidat für die Erweiterung. Die korrespondierende normalisierte Zell-Punktewolke  $O_{u,v}$  wird iterativ unterteilt, um die in der Zelle potentiell enthaltenen Bodenanteile anzunähern. Dazu wird die Fläche der Zelle in vier gleichgroße Subzellen  $L_{i,j}$  mit  $i,j \in \{0, 1\}$  unterteilt, die in der Menge  $S_{u,v}$  zusammengefasst werden:

$$S_{u,v} = \{L_{0,0}, L_{1,0}, L_{0,1}, L_{1,1}\} .$$

Für die Berechnung der einzelnen Subzellen  $L_{i,j}$  in der ersten Stufe gilt:

$$\begin{aligned} L_{i,j}(O_{u,v}) = \{ & (\mathbf{A} \cdot \mathbf{p}, d) \mid (\mathbf{p}, d) \in O_{u,v}, \\ & p_x \in [i \cdot 0.5, i \cdot 0.5 + 0.5], \\ & p_y \in [j \cdot 0.5, j \cdot 0.5 + 0.5], \\ & p_z \in \mathbb{R} \} . \end{aligned}$$

Die Matrix  $\mathbf{A} \in \mathbb{R}^{3 \times 3}$  ist eine Skalierungsmatrix, die die Punkte  $\mathbf{p} \in \mathbb{R}^3$  analog normalisiert:

$$\mathbf{A} = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix} .$$

**Algorithmus 5.2** Rekursive Zellenerweiterung

---

```

1: procedure SUBDIVIDECCELL( $O, r$ )
2:   if  $r < 1$  or  $|O| < \vartheta$  then           ▷ Abbruchkriterien prüfen
3:     return {}
4:    $S \leftarrow \{\}$ 
5:   for  $i := 0$  to 1 do
6:     for  $j := 0$  to 1 do
7:        $A \leftarrow L_{i,j}(O)$            ▷ Subzelle berechnen
8:       if  $\text{ground}(A)$  then           ▷ Gehört die Subzelle zum Boden?
9:          $S \leftarrow S \cup \{A\}$ 
10:      else
11:         $S \leftarrow S \cup \text{SUBDIVIDECCELL}(A, 0.5 \cdot r)$ 
return  $S$ 

```

---

Die Unterteilung der Kandidatenzellen wird iterativ ausgeführt, wie in Algorithmus 5.2 dargestellt. Sollte eine Subzelle als *Boden* klassifiziert werden, kann sie genutzt werden, um das geschlossene Oberflächenmodell zu erweitern. Andernfalls wird der Unterteilungs-Algorithmus ausgeführt, bis eine *Boden*-Zelle gefunden wird bzw. ein Abbruchkriterium erreicht wird. Ein Abbruch findet statt, wenn zu wenige Punkte in einer Subzelle verbleiben oder wenn der potentielle Auflösungsgewinn für die Textur kleiner als ein Pixel wäre.

Sind Subzellen gefunden, die Erweiterungskandidaten für das Oberflächenmodell darstellen, erfolgt eine weitere Prüfung hinsichtlich der Validität der Kandidaten. Dabei wird geprüft, ob die Subzellen eine gemeinsame Kante mit einer *Boden*-Nachbarzelle direkt bzw. durch gemeinsame Kanten mit weiteren Erweiterungskandidaten ohne Unterbrechung verbunden werden können (transitive Relation). Dadurch wird verhindert, dass von der Hauptoberfläche getrennte Subzellen in das Oberflächenmodell integriert werden.

Die Idee hinter der iterativen Zell-Verbesserung ist in Abbildung 5.35 dargestellt. Ein Beispiel mit realen Daten für die Unterteilung ist in Abbildung 5.37 zu sehen.

Für die Berechnung der Oberflächen-Dreiecke für die Subzellen muss sichergestellt werden, dass diese nicht die Idee des geschlossenen Oberflächenmodells verletzen. Um dies zu erreichen, werden zwei Fälle unterschieden: Wenn die Subzelle Nachbar einer größeren (Sub-)Zelle ist, wird die angrenzende Höhe entsprechend den Höhen der Nachbarzellen interpoliert, sodass die größere Geometrieinheit die Höhe bestimmt. Ist die gesuchte Eckenhöhe nicht durch benachbarte Zellen bestimmbar, so wird das normale Verfahren analog zur Bestimmung der Zellhöhen angewandt. Generell werden zuerst Höhen von geometrisch größeren Subzellen bestimmt, da diese bei Betrachtung der nächst kleineren möglichen Zellgröße Einfluss auf deren Höhenbestimmung nehmen können. Der Zusammenhang ist in

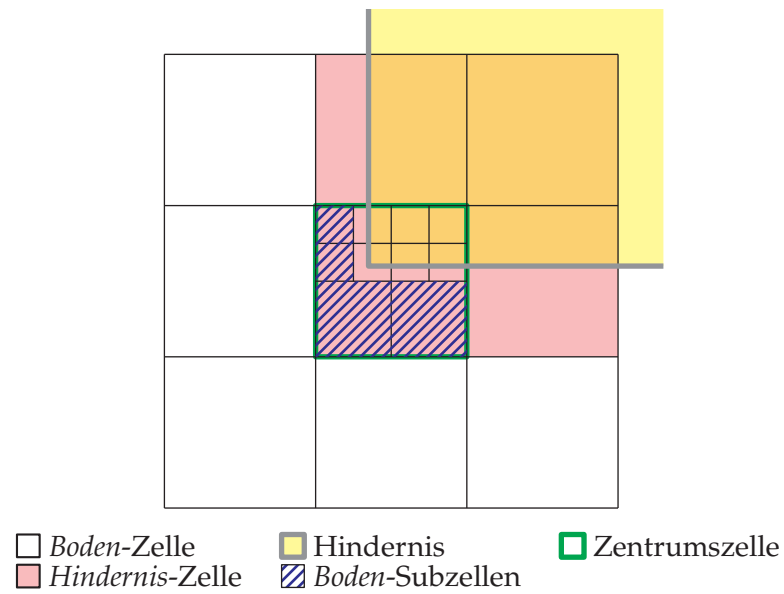


Abbildung 5.35: Prinzip der iterativen Zellunterteilung

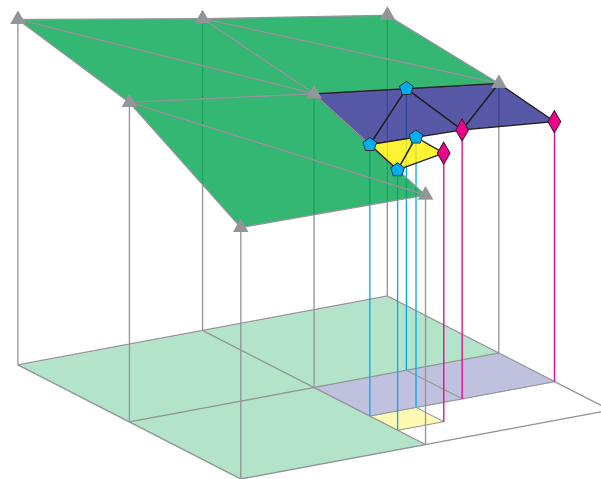
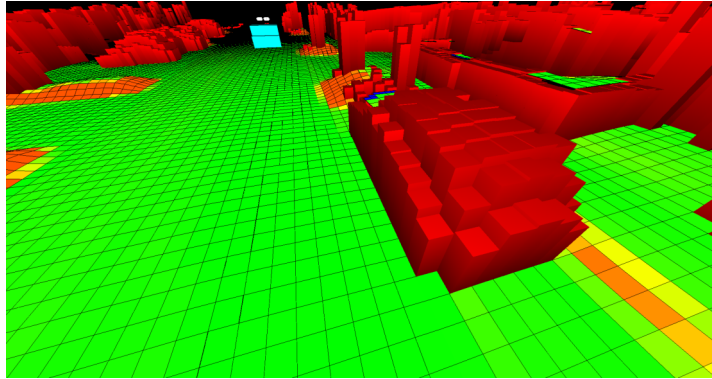
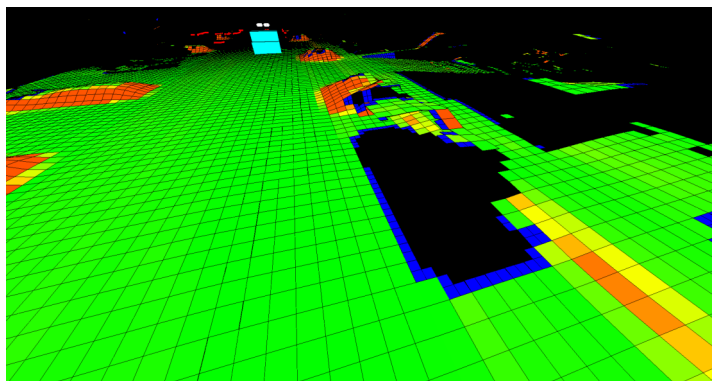


Abbildung 5.36: Höhenbestimmung für Subzellen. Die *Boden-Zellen* der Rasterkarte (grün) werden durch Subzellen (blau und gelb) erweitert. Die Eckhöhen der Subzellen werden wo möglich durch benachbarte, größere (Sub-)Zellen bestimmt (cyan). Andernfalls werden die Höhen nach vorgestelltem Verfahren bestimmt (magenta).

Abbildung 5.36 dargestellt. Die Eckpunkthöhen, die aus benachbarten Zellkanten interpoliert werden (cyan), werden anders behandelt als benachbarten Zellkanten abgewandte Eckpunkte (magenta).



(a) Oberflächenmodell mit durch den statistischen Ansatz berechneten Hindernissen (rot)



(b) Verbesserte Subzellen (blau)

Abbildung 5.37: Beispiel für die iterative Zellverbesserung: Die *Hindernis*-Zellen (rot) werden durch die Subzellen (blau) verfeinert und nähern die Bodenfläche besser an das Hindernis an.

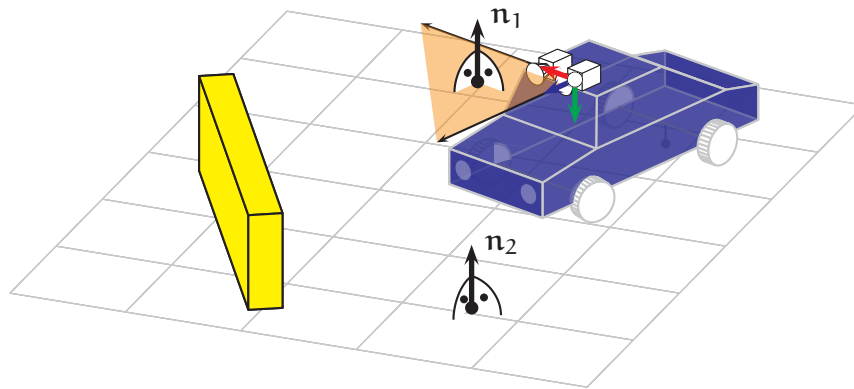


Abbildung 5.38: Modellannahme des „Stixel World“-Verfahrens nach Badino, Franke und Pfeiffer [BFP09]. Die  $xz$ -Ebene der Kamera (orange) wird als parallel zur Bodenebene angenommen.

## 5.8 ERWEITERTE STEREO-HINDERNISERKENNUNG

Die Hindernisdetektion im bisherigen Verfahren nutzt die gleitende lokale Rasterkarte, um für jede Zelle zu entscheiden, ob sie als *Hindernis*, *Boden* oder ggf. *Unbekannt* klassifiziert wird. Durch die iterative Zellverbesserung (siehe Kapitel 5.7) wird die Limitierung auf die Zellengröße als Grundgröße für Hindernisse aufgeweicht. Nichtsdestotrotz bleibt die lokale Betrachtung der einzelnen Zellen bestehen. Auch wenn dieses Verfahren eine gute Erkennung liefert, ist es für das Gesamtziel wünschenswert, eine Hindernisdetektion zu integrieren, die nicht alleine auf dem geometrischen Zusammenfassen der Punktwolken beruht. Dadurch soll eine präzisere Bestimmung der Hindernisgrenzen und den damit einhergehenden potentiellen Gefahrenpunkten ermöglicht werden. Für die Anzeige in der perspektivisch korrekten Ansicht aus der Vogelperspektive kann mit einem solchen Verfahren eine Visualisierung der Hindernisse und damit des zum Rangieren zur Verfügung stehenden Platzes realisiert werden. Die geplante Darstellung soll die Hindernisinformationen in der Vogelperspektivenansicht ergänzen.

Badino, Franke und Pfeiffer [BFP09] haben mit der „Stixel World“ ein effizientes Verfahren entwickelt und veröffentlicht, das direkt auf den Disparitätswerten eines Stereo-Matchings aufsetzt, um Hindernisse zu detektieren und zu Hinderniseinheiten zusammenzufassen. Im Folgenden werden die Grundlagen des Verfahrens dargestellt und im Anschluss werden die Erweiterungen und Anpassungen für die Integration in die perspektivisch korrekte Ansicht aus der Vogelperspektive vorgestellt.

Das „Stixel World“-Verfahren ist primär für den Einsatz in der Objektdetektion im vorwärtigen Fahrzeugumfeld eines PKW entwickelt. Das zugrundeliegende Modell geht von einer Stereokamera aus, die

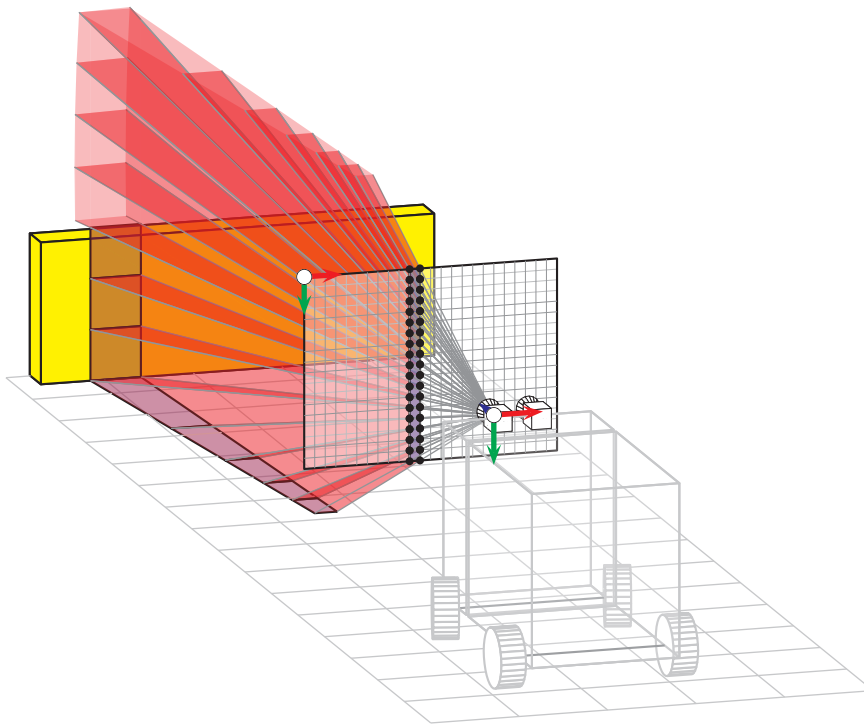


Abbildung 5.39: Pixelweise Abtastung der Umgebung mit Visualisierung der durch einzelne Pixel abgedeckten Bereiche

parallel zur Bodenebene orientiert ist und in Richtung des Horizonts blickt (siehe Abbildung 5.38).

Ein Stixel im Sinne des Verfahrens ist dabei ein rechteckiges Geometrielement, das für die Umgebungsrepräsentation verwendet wird. Stixel sind parallel zur Bildebene orientiert und beschreiben die vorderen Abgrenzungen von Hindernissen. Stixel haben eine maximale Breite (im einfachen Fall 1 px). Dies bedeutet, dass ein (größeres) Hindernis von mehreren Stixeln beschrieben wird, die säulenartig in der Tiefe des Hindernisses lokalisiert sind und das Hindernis in seiner Höhe durch ihre geometrische Ausdehnung approximieren.

Aus den synchronisierten Stereo-Aufnahmen werden zunächst dichte Disparitätenkarten berechnet, die die Grundlage für die folgende Verarbeitung darstellen. Die Pixel-Topologie der Disparitätenkarte bzw. des Kamerabildes wird verwendet, um die geometrische Relation zwischen den einzelnen Messwerten herzustellen. Im ersten Schritt werden die einzelnen Bildspalten der Disparitätenkarte betrachtet und jeweils ein Histogramm über die Disparitätswerte angelegt. Ein Beispiel der Abtastung einer einzelnen Bildspalte ist in Abbildung 5.39 visualisiert.

Der Boden wird im Verfahren durch eine Approximierung mit sog. *B-Splines* beschrieben und dem Boden zugeordneten lokalen Disparitätswerte bzw. die korrespondierenden Bereiche im Disparitätsbild werden vor der Weiterverarbeitung gefiltert.

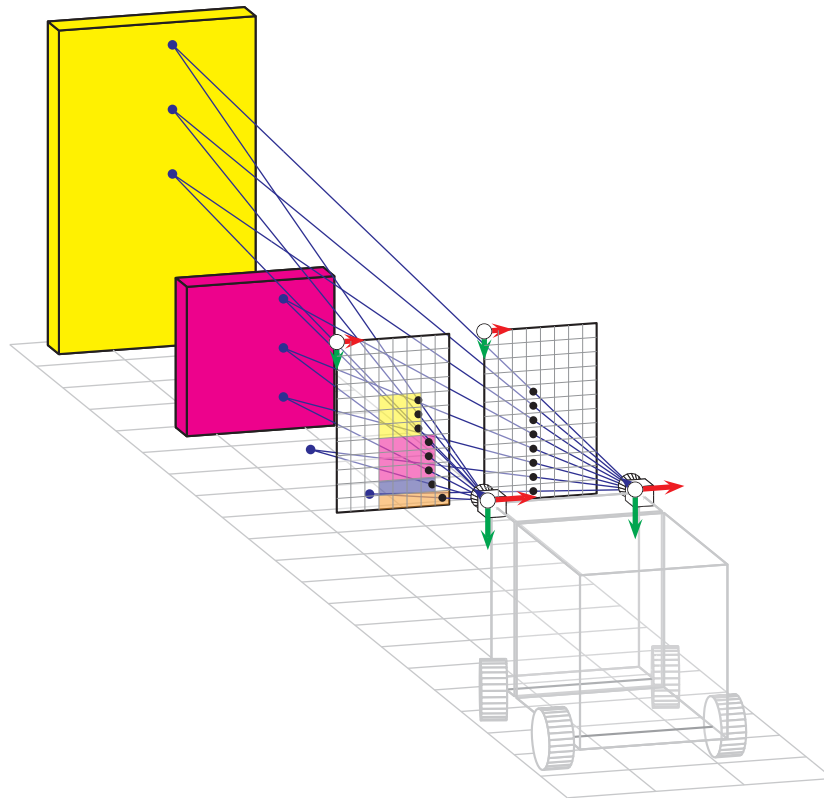


Abbildung 5.40: Bei Blickrichtung der Stereokamera in Richtung des Horizonts erzeugen vertikale Objekte, die parallel zur Bildebene sind, jeweils gleiche Disparitätswerte. Im dargestellten Beispiel wird für jede Wand jeweils der gleiche Disparitätswert (im linken Kamerabild dargestellt) ermittelt. Auf dem Boden verändern sich die Werte für jede Abtastung.

Unter Berücksichtigung der Ausrichtung der Kamera ergibt sich, dass vertikale Strukturen im Bild, wie z. B. Wände, in gleichen Disparitätswerten resultieren. Diese werden in Histogrammen zusammengefasst. Allgemein ist ein Histogramm  $\mathcal{H}$  eine Abbildung eines Wertes aus einem endlichen Wertebereich (ggf. ganzzahlig bzw. abgetastete reelle Werte) zur absoluten Anzahl der Vorkommen des Wertes in der Ausgangsmenge. Das Histogramm lässt sich folglich als Zählung von Vorkommen eindeutiger Entitäten des bzw. Werten innerhalb eines Intervalls im Wertebereich verstehen. Im Fall des Disparitäten-Histogramms wird das absolute Vorkommen von Disparitätswerten im Histogramm akkumuliert.

An entsprechendem Histogramm-Eintrag für ein vertikales Objekt ist folglich ein hoher Wert (für die Anzahl der Vorkommen) zu erwarten. Der Zusammenhang ist in Abbildung 5.40 dargestellt.

Gegeben nun ein Histogramm  $\mathcal{H}$  über eine Spalte des Disparitätenbildes hinsichtlich der Bildtopologie. Das Histogramm verfügt über



$\lambda_{\max}$  Klassen, wobei  $\lambda_{\max} \in \mathbb{N}$  den maximalen Disparitätswert beschreibt. Ziel ist es, ein *vertikales* Hindernis zu extrahieren. Ein vertikales Hindernis erzeugt im Histogramm eine hohe Anzahl gleicher bzw. sehr ähnlicher Disparitätswerte. Als Strategie kann folglich eine Maximum-Suche innerhalb des Histogramms  $\mathcal{H}$  identifiziert werden.

Die Suche nach einem globalen Maximum ist an dieser Stelle jedoch wenig zielführend, da das jeweils nächste Hindernis (Abstand zum Fahrzeug) dasjenige ist, dem die höchste Relevanz zukommt. In Bezug auf die Stereo-Disparitäten bedeutet ein hoher Disparitätswert einen niedrigen Abstand zur Kamera. Daher wird das nächstgelegene Hindernis durch ein lokales Maximum mit hohem Disparitätswert beschrieben. Es ergibt sich ein Suchproblem nach der das Hindernis beschreibenden Disparität(sklasse)  $\lambda_0$ , mit den folgenden Bedingungen:

- $\mathcal{H}(\lambda_0) > \vartheta$   
Die Häufigkeit der Disparität(sklasse)  $\lambda$  ist oberhalb eines Schwellwerts  $\vartheta$ .
- Der Disparitätswert von  $\lambda_0$  soll möglichst groß sein, d. h. das Hindernis möglichst nah an der Kamera.

Es ergibt sich:

$$\lambda_0 = \max \{ \lambda \mid \mathcal{H}(\lambda) > \vartheta \} \quad .$$

Für den Fall, dass keine geeignete Disparität  $\lambda_0$  ermittelt werden kann – d. h. die Suchmenge ist leer – wird für das entsprechende Histogramm kein Hindernis gefunden und der Bildbereich als „frei“ angenommen.

Im Verfahren nach Badino, Franke und Pfeiffer [BFP09] werden die einzelnen Histogramme in einem Belegtheitsgitter (engl. *occupancy grid*) angeordnet. Die resultierende 2-D-Matrix hat die Dimensionen  $\lambda_{\max} \times w_\lambda$  mit  $\lambda_{\max} \in \mathbb{N}$  dem maximalen Disparitätswert und  $w_\lambda \in \mathbb{N}$  der Anzahl Spalten bzw. der Bildbreite des Disparitätsbildes in Pixeln. Als Vorverarbeitungsschritt der Auswertung nutzt das Verfahren eine Vorfilterung des Histogramms, bei der Hintergrundobjekte eliminiert werden.

Die Histogramm-Auswertung wird für jede Bildspalte durchgeführt, sodass eine durch die Bildtopologie bedingte Ordnung in den ermittelten Hinderniskandidaten  $\lambda_0$  in Bezug auf die Nachbarschaftsrelation besteht. Durch die Annahme zur Kameraausrichtung ist die Auswertung je Bildspalte mit einer Auswertung je vertikalem Schnitt durch die Welt vergleichbar.

Aus dem Belegtheitsgitter kann pro Bildspalte eine Aussage über das Hindernis getroffen werden, das die geringste Distanz zur Kamera hat. Es ist an dieser Stelle zu erwarten, dass sich Hindernisse über mehrere Bildspalten erstrecken. Durch eine Korrelationsuche in benachbarten Spalten kann somit die Konfidenz hinsichtlich der

Korrektheit des gefundenen Hindernisses erfolgen. Durch eine einfache Gruppierung werden potentiell zusammenhängende Objektgrenzen identifiziert, indem die Ergebnisse der Histogramm-Auswertung nach der x-Koordinate der korrespondierenden Bildspalte sortiert der Reihe nach betrachtet werden. An dieser Stelle können auch zu kleine Gruppen-Kandidaten mit insignifikanter Breite (z. B. durch einen Schwellwert) aussortiert werden. Eine Heuristik, die die Nachbarschaftsrelationen auswertet, sorgt für die Zusammenfassung einzelner Hindernisspalten zu spaltenübergreifenden Hindernissen.

Gegeben eine Menge von über Spalten zusammengefassten Hindernissen kann die geometrische Ausdehnung des Hindernisses ermittelt werden. In einem ersten Schritt werden die Stixel für die Spalten in der Menge erzeugt. Die Anzahl der für ein Hindernis notwendigen Stixel ergibt sich trivial aus der maximalen Breite eines Stixels.

Zusätzlich werden Höhen für die jeweiligen Stixel ermittelt. Dazu werden die y-Koordinaten der durch die Histogramm-Auswertung ausgewählten Disparitätsklasse im Disparitätenbild betrachtet und ebenfalls entsprechend zusammengefasst. Bildlich ist das Ergebnis ein Polygonzug je Hindernis im Disparitätenbild. Im Verfahren nach Badino, Franke und Pfeiffer [BFP09] wird ein Ansatz mittels dynamischer Programmierung vorgeschlagen, die eine Trennung von Vorder- und Hintergrundbereichen realisiert. Im vorgestellten erweiterten Verfahren nimmt die präzise Ermittlung der Hindernishöhe lediglich eine niedrige Priorität ein, da eine Integration der detektierten Hindernisse in die perspektivisch korrekte Ansicht aus der Vogelperspektive das Zielszenario ist. Vielmehr ist die Abgrenzung in Bezug auf den Abstand der Kamera interessant. Badino, Franke und Pfeiffer [BFP09] beleuchten die Fragestellung nach der Hindernishöhe genauer. An dieser Stelle wird aus genannten Gründen auf eine genaue Untersuchung der Hindernishöhen verzichtet.

Die identifizierten Hindernisse werden mit der bekannten Stereokalibrierung in eine 3-D-Repräsentation transformiert. Unter Verwendung der ermittelten vertikalen Ausdehnung des Hindernisses wird eine Fläche berechnet, die die vordere Abgrenzung des Hindernisses in Richtung der Kamera bzw. des Fahrzeugs darstellt.

Bildlich entspricht diese Fläche einer Art „Gartenzaun“, der das Hindernis an seiner vorderen Spitze abgrenzt. Der Bereich zwischen der Kamera und den Abgrenzungen wird als *Freifläche* angenommen.

### 5.8.1 Verallgemeinerter Ansatz

Das von Badino, Franke und Pfeiffer [BFP09] vorgestellte Verfahren ermöglicht es, die jeweils nächstgelegenen Hindernisse zum Fahrzeug zu identifizieren. Auch wenn die Publikation einige Teilbereiche und Details hinsichtlich der genauen Umsetzung ausspart, wurde das Verfahren im Rahmen einer studentischen Abschlussarbeit nachim-

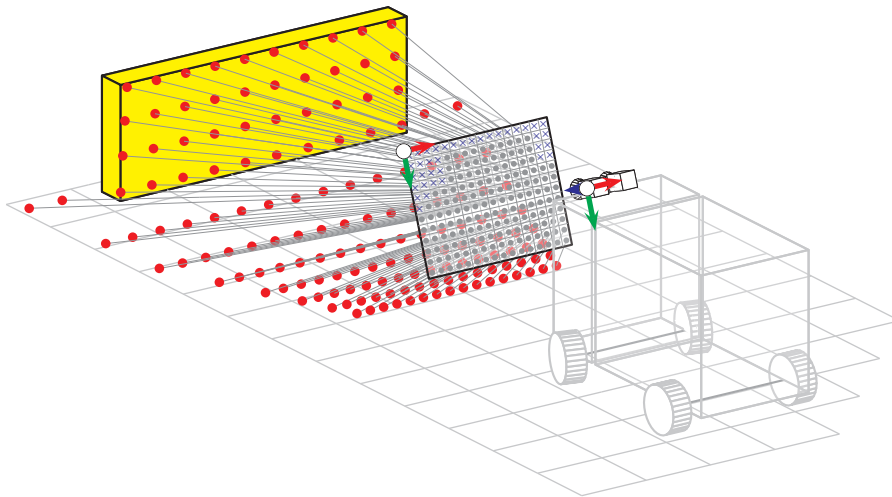


Abbildung 5.41: Verteilung der rückprojizierten 3-D-Punkte bei der Umgebungsabtastung mittels Stereokamera

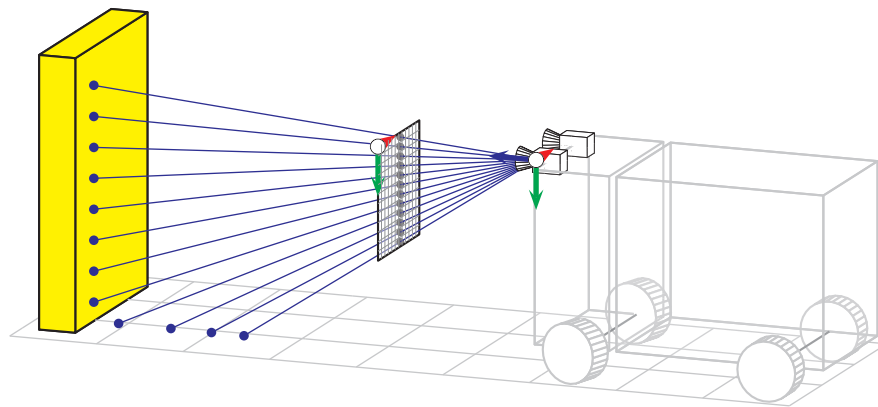
plementiert und erweitert.

Im Bereich der Generierung von perspektivisch korrekten Ansichten aus der Vogelperspektive ist zunächst eine Verletzung der im Ansatz nach Badino, Franke und Pfeiffer [BFP09] getroffenen Annahme hinsichtlich der Ausrichtung der Kamera festzustellen. Da es Ziel ist, die direkte Fahrzeugumgebung zu überwachen und mit aktuellen Kamerabildern zu visualisieren, sind Kameras mit Ausrichtung in Richtung des Bodens das sinnvollste Szenario. Ein Beispiel für die Abtastung einer Szene durch eine solche Kameraausrichtung ist in Abbildung 5.41 dargestellt.

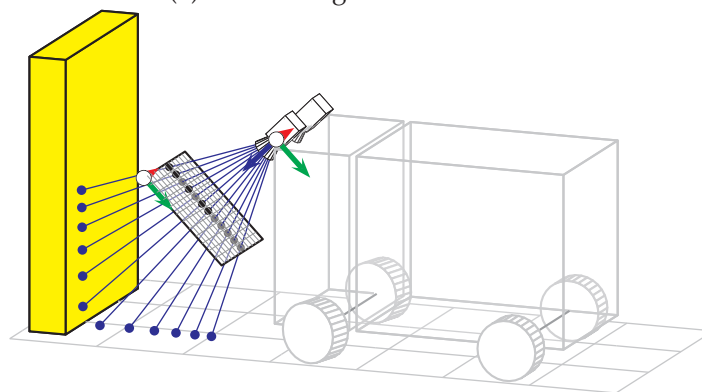
Diese Anordnung bedingt jedoch, dass die Modellannahme und die darauf aufbauende Auswertung je Bildspalte der Disparitätshistogramme nicht eingesetzt werden kann. Der Unterschied hinsichtlich der Verteilung der Tiefendaten bei unterschiedlicher Kameraausrichtung ist in Abbildung 5.42 dargestellt.

Im Folgenden werden Anpassungen und Erweiterungen für das „Stixel World“-Verfahren vorgestellt, die den Ansatz generischer formulieren und damit die Limitierung hinsichtlich der Kameraausrichtung aufheben. Die Erweiterung nutzt die aus der Disparitätenkarte zu berechnende Punktwolke der Stereokamera als Eingabe, um die „Stixel World“ zu generifizieren.

Gegeben eine initiale Stereo-Punktwolke  $P^c$  im Koordinatensystem der Stereo-Hauptkamera  $C_0$ . Die Lage der Stereo-Hauptkamera relativ zum Koordinatensystem des Fahrzeugs sei durch die Pose  $\mathfrak{s} = \langle \mathbf{p}_{\mathfrak{s}}, \mathbf{r}_{\mathfrak{s}} \rangle$  beschrieben. Da die  $xy$ -Ebene des Fahrzeug-Koordinatensystems parallel zur idealen Bodenebene des Fahrzeugs liegt, Beschreibt die Rotationskomponente  $\mathbf{r}_{\mathfrak{s}}$  der Pose  $\mathfrak{s}$  die relative Rotation der Stereo-Hauptkamera in Bezug auf die Ausrichtung der Modell-Ebenen und damit der Modell-Blickrichtung in Richtung des



(a) In Richtung des Horizonts blickende Kamera



(b) Zum Boden geneigte Kamera

Abbildung 5.42: Unterschiedliche Tiefenverteilung bei unterschiedlichen Kameraausrichtungen und gleicher Umgebungsgeometrie

Horizonts (unter Vernachlässigung der Ausrichtung der Koordinatensysteme).

Durch eine Transformation der Punktwolke in diese Ausrichtungsmodalität kann die Kompatibilität mit der Annahme nach dem „Stixel World“-Verfahren hergestellt werden. Das Ziel-Koordinatensystem, dessen Ursprung im selben Punkt wie das des Stereo-Hauptkamera-Koordinatensystems  $c$  liegt, ist mit  $h$  bezeichnet.

Die Kamerablickrichtung ist entlang der  $z$ -Achse im jeweils lokalen Kamera-Koordinatensystem definiert. Die  $y$ -Achse beschreibt die Rechts-Richtung (engl. *right vector*) der Kamera. Ziel der Transformation ist es folglich, die  $yz$ -Ebene des Kamera-Koordinatensystems so zu transformieren, dass diese parallel zur  $xy$ -Ebene des Fahrzeug-Koordinatensystems liegt.

Die Transformation zwischen den Koordinatensystemen sei durch die Pose  $\mathbf{h} \in \mathbb{R}^3 \times \text{SO}(3)$  mit  $\mathbf{h} = \langle \mathbf{p}_{\mathbf{h}}, \mathbf{r}_{\mathbf{h}} \rangle$  beschrieben. Da es sich um eine reine Änderung der Orientierung des Koordinatensystems handelt, gilt  $\mathbf{p}_{\mathbf{h}} = \mathbf{0}^T$ .

Zur Neutralisierung der standardmäßigen Ausrichtungen der Koordinatensysteme von Kamera  $c$  und Fahrzeug  $v$  wird die Pose  $\mathbf{b} \in \mathbb{R}^3 \times SO(3)$  als Hilfs-Korrekturpose verwendet. Sie beschreibt die reine Rotation zwischen den Standard-Ausrichtungen der Koordinatensysteme und akkumuliert eine  $90^\circ$ -Rotation um die  $x$ -Achse mit einer anschließenden  $90^\circ$ -Rotation um die  $y$ -Achse. Für die Pose  $\mathbf{b}$ , die von der *Standard-Ausrichtung* des Fahrzeug-Koordinatensystems in die *Standard-Ausrichtung* des Kamera-Koordinatensystems transformiert, ergibt sich:

$$\mathbf{b} = \langle \mathbf{0}^T, \mathbf{r}_{\mathbf{b}} \rangle$$

$$\text{rot}(\mathbf{r}_{\mathbf{b}}) = \text{angleaxis} \left( -90^\circ, (1, 0, 0)^T \right) \cdot \text{angleaxis} \left( 90^\circ, (0, 1, 0)^T \right) \quad .$$

Da die Ausrichtung der in der  $xy$ -Ebene (Horizont-Ebene) liegenden Basen zu vernachlässigen ist, kann die  $y$ -Achse des jeweiligen Koordinatensystems als Normale der Ebene betrachtet werden. Um die Rotation zu berechnen, müssen diese jedoch in ein einheitliches Koordinatensystem (hier: Fahrzeug-Koordinatensystem) transformiert werden. Da es sich um Normalen handelt, erfolgt keine Anwendung der Translationskomponente.

$$\mathbf{n}_c = \mathbf{T}(\langle \mathbf{0}^T, \mathbf{r}_{\mathbf{c}} \rangle) \cdot (0, 1, 0)^T$$

$$\mathbf{n}_h = \mathbf{T}(\mathbf{b}) \cdot (0, 1, 0)^T = (0, 0, -1)^T \quad .$$

Der Winkel  $\gamma$  zwischen den Normalen  $\mathbf{n}_c$  und  $\mathbf{n}_h$  beschreibt die für die Drehung des Stereo-Hauptkamera-Koordinatensystems in die Horizontebene notwendige Rotation. Die damit einhergehende Rotationsachse  $\mathbf{a} \in \mathbb{R}^3$  ergibt sich durch die Normale der Ebenen, die durch die Normalen  $\mathbf{n}_c$  und  $\mathbf{n}_h$  aufgespannt wird. Es ergibt sich:

$$\gamma = \arccos(\mathbf{n}_c \circ \mathbf{n}_h)$$

$$\mathbf{a} = \|\mathbf{n}_c \times \mathbf{n}_h\| \quad .$$

Die Rotation der Pose  $\mathbf{h}$  von dem Kamera-Koordinatensystem hin zur virtuellen Horizont-Kamera  $V_H$  ergibt sich durch die entsprechende Drehung:

$$\text{rot}(\mathbf{r}_{\mathbf{h}}) = \text{angleaxis}(\gamma, \mathbf{a}) \quad .$$

Für den Sonderfall  $\mathbf{n}_c \circ \mathbf{n}_h = 0$  (d. h. die Stereo-Hauptkamera blickt bereits in Richtung des Horizonts) gilt:

$$\text{rot}(\mathbf{r}_{\mathbf{h}}) = \mathbf{Id}_3 \quad .$$

Der geometrische Zusammenhang ist in Abbildung 5.43 dargestellt. Weist die Kamera zusätzlich eine Rotation in Roll-Richtung (entlang der  $z$ -Achse) auf, so kann der damit verbundene Effekt analog mit einer Korrekturpose neutralisiert werden.

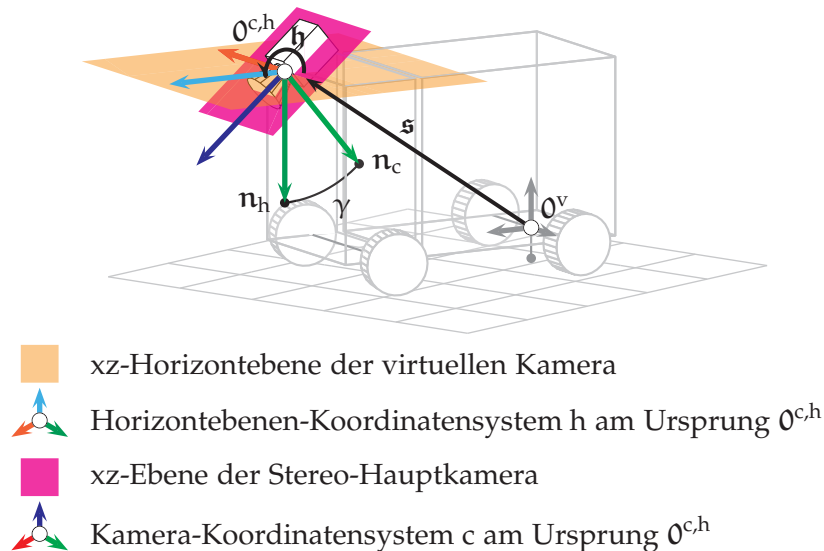


Abbildung 5.43: Geometrische Zusammenhänge für die virtuelle Horizontausrichtung der Kamera

Die in Kapitel 5.5.2 vorgestellte Technik kann für die Transformation der Koordinatensysteme zur Anwendung kommen. Sei die Menge der 3-D-Punkte  $P^c$  für die Stereo-Hauptkamera  $C_0$  im Koordinatensystem  $c$  der Kamera gegeben. Die Punktwolke im Koordinatensystem der virtuellen Horizont-Kamera  $V_H$  an der Position der Stereo-Hauptkamera  $C_0$  mit Blick in Richtung des Horizonts ergibt sich durch:

$$P^h = \Upsilon_{V_H} = \{T_{c \rightarrow h}(\mathbf{p}) \mid \mathbf{p} \in P^c\} = \{T(\mathbf{h}) \cdot \mathbf{p} \mid \mathbf{p} \in P^c\} \quad .$$

Die nun vorliegende Punktwolke  $P^h$  der virtuellen Horizont-Kamera  $V_H$  stellt die Grundlage für die weitere Verarbeitung dar. Durch die vorgenommene Transformation geht die Pixeltopologie des ursprünglichen Disparitätenbildes verloren, sodass das Histogrammbasierte Vorgehen aus der „Stixel World“ nicht direkt adaptierbar ist. Ziel ist es, eine alternative Struktur aufzubauen, mit der ein vergleichbares Konzept angewandt werden kann.

Theoretisch ist es möglich, die 3-D-Punkte in ein virtuelles Kamerabild – wie in Kapitel 5.5.2 vorgestellt – zu transformieren und aus den Abständen virtuelle Disparitäten zu berechnen. Dieses Vorgehen ist jedoch wenig zielführend, da keine vollständig besetzte Pixeltopologie zu erwarten ist und die Transformationen zwischen Disparitäten und 3-D-Punkten ambivalent wären. Da die ursprünglichen Disparitätswerte bereits in 3-D-Punkte konvertiert wurden, macht es Sinn, auf diesen Daten weiterzuarbeiten.

Die virtuelle Horizont-Kamera  $V_H$  bedingt, dass die Ausrichtung des Koordinatensystems  $h$  in Richtung des Horizonts schaut und die xz-Ebene der Kamera parallel zur idealen Bodenebene/dem Fahrzeug-Koordinatensystem ist. Analog zur spaltenweisen Bearbei-

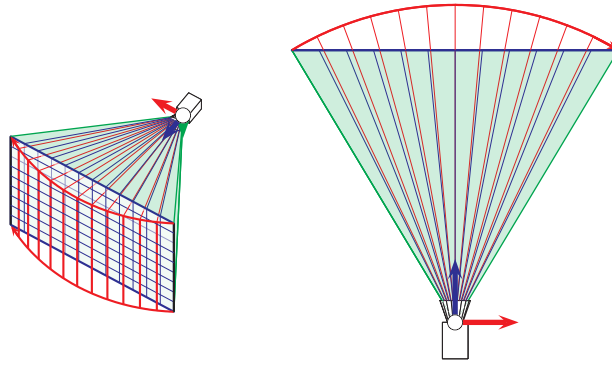


Abbildung 5.44: Unterschied der Sichtwinkelaufteilung im Vergleich zur Pixelabtastung der Bildebene

tion des Disparitätenbildes kann damit eine Raumaufteilung mittels Ebenen erreicht werden, deren Normale in der  $xz$ -Ebene der virtuellen Horizont-Kamera  $V_H$  liegt.

Um die Pixeltopologie des Disparitätenbildes anzunähern wird eine Abtastung des horizontalen Kameraöffnungswinkels  $\alpha$  vorgenommen, die die Schrittweite der Pixelspalten im Disparitätenbild abbildet. Gegeben  $w_\lambda \in \mathbb{N}$  die Anzahl der Pixel des Disparitätenbildes in  $x$ -Richtung und damit auch die Zielanzahl der Unterteilungen des Kamerasichtfelds. Die Schrittweite  $\Delta_\alpha \in \mathbb{R}$  für die Abtastung des Kameraöffnungswinkels  $\alpha$  ergibt sich mit:

$$\Delta_\alpha = \frac{\alpha}{w_\lambda} \quad .$$

Die Abtastung des Kameraöffnungswinkels (Kreisbogen) stellt dabei eine Abweichung im Vergleich zur Diskretisierung der Bildebene in eine Pixelstruktur (Ebene) dar. Der Zusammenhang ist in Abbildung 5.44 visualisiert. Systematisch liegt eine andere Abtastung vor, die jedoch nur insignifikante Auswirkung hat. Daher wird der Effekt im Folgenden ignoriert.

Gegeben eine Ebene  $E$ , die durch den Punkt  $0^h$  und die Normale  $\mathbf{n} \in \mathbb{R}^3$  bestimmt ist. Es gilt:

$$\mathbf{n} \circ (0, 1, 0)^T = 0 \quad .$$

Die Ebene  $E$  verläuft durch den Ursprung des Koordinatensystems  $h$  und teilt den 3- $D$ -Raum als Hyperebene. Für die Isolierung eines durch zwei Ebenen definierten Ausschnitts der Punktwolke  $P^h$  für die  $i$ -te Abtastung (mit  $i \in [0; w_\lambda]$ ,  $i \in \mathbb{N}$ ) des horizontalen Kameraöffnungswinkels werden die entsprechenden Winkel  $\beta_j$  mit  $j \in [0; w_\lambda + 1]$ ,  $j \in \mathbb{N}$  für die Abgrenzung der jeweiligen Sub-Räume berechnet (siehe Abbildung 5.45):

$$\beta_j = -0.5 \cdot \alpha + j \cdot \Delta_\alpha \quad .$$

Die Schnittebenen für die  $i$ -te Diskretisierung werden wie in Ab-

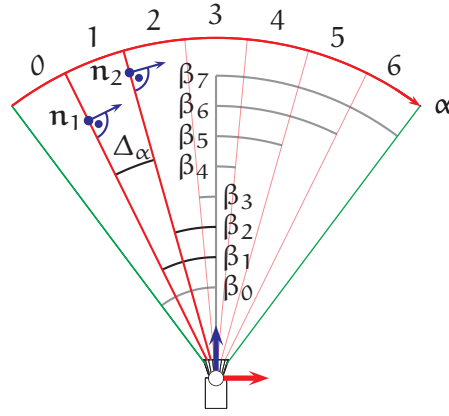


Abbildung 5.45: Abtastung des horizontalen Kameraöffnungswinkels. Die Winkel  $\beta_i$  beschreiben die Auslenkung des Strahls auf der  $xz$ -Ebene in Relation zur  $z$ -Achse. Die zu Spalte 1 korrespondierenden Begrenzungslinien bzw. Hyperebenen werden durch die Winkel  $\beta_1$  und  $\beta_2$  bzw. die Normalen  $\mathbf{n}_1$  und  $\mathbf{n}_2$  definiert.

Abbildung 5.45 durch die beiden Winkel  $\beta_i$  und  $\beta_{i+1}$  bestimmt. Die korrespondierenden Normalen ergeben sich durch:

$$\mathbf{n}_j = (\cos \beta_j, 0, \sin \beta_j)^T .$$

Die Normalen  $\mathbf{n}_j$  beschreiben zusammen mit dem Kamerazentrum und Ursprung des Koordinatensystems  $0^h$  die jeweilige Schnittebene  $E_j$ . Die Prüfung, ob ein Punkt  $\mathbf{p} \in P^h$  im Sub-Raum der  $i$ -ten Abtastung liegt, wird durch die boolesche Funktion  $\text{insubset}(i, \mathbf{p}) : \mathbb{N} \times \mathbb{R}^3 \rightarrow \{0, 1\}$  mit folgendem logischen Ausdruck definiert:

$$\text{insubset}(i, \mathbf{p}) = \mathbf{n}_i \circ \mathbf{p} \geq 0 \quad \wedge \quad \mathbf{n}_{i+1} \circ \mathbf{p} < 0 .$$

Die Menge  $A_i$  der zu einer Abtastung  $i$  gehörenden 3-D-Punkte ergibt sich durch:

$$A_i = \left\{ \mathbf{p} \mid \mathbf{p} \in P^h \wedge \text{insubset}(i, \mathbf{p}) \right\} .$$

Auf Basis der Menge  $A_i$  kann eine Auswertung erfolgen, die sich am Originalverfahren nach Badino, Franke und Pfeiffer [BFP09] orientiert. Das Prinzip hinter der angepassten Abtastung der Punktwolke ist in Abbildung 5.46 visualisiert. Da die Disparitäten an dieser Stelle jedoch bereits in 3-D-Punkte umgerechnet wurden, ist es – wie bereits erwähnt – sinnvoll, direkt auf diesen Daten weiterzuarbeiten.

Das Histogramm  $\mathcal{H}_{\text{dist}}$  wird über die Distanz  $\|\mathbf{p}\|$  der Punkte  $\mathbf{p} \in A_i$  zur Kamera gebildet. Da die Distanzwerte – im Gegensatz zu den Disparitätswerten – nicht als diskrete Werte vorliegen, muss eine entsprechende Schrittweite für die Klassen des Histogramms festgelegt werden, die aus den Kalibrierdaten abgeleitet werden kann. Das



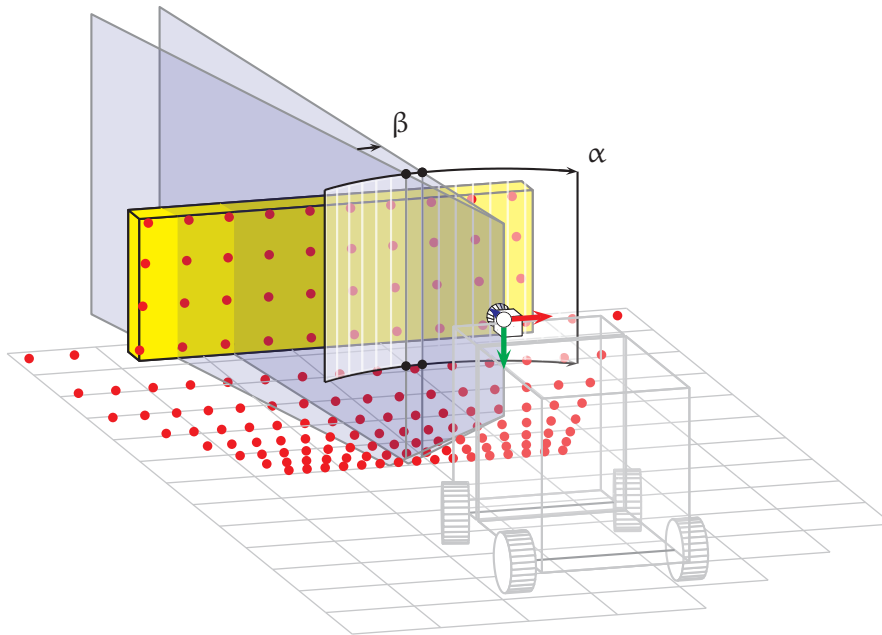


Abbildung 5.46: Abtastung des horizontalen Kameraöffnungswinkels  $\alpha$  mit der Schrittweite  $\beta$  durch Sichtebenen

so entstehende Tiefen-Belegtheitsgitter (engl. *depth occupancy grid*) hat eine äquivalente Semantik im Vergleich zum Belegtheitsgitter, in dem die Histogramme über die Disparitäten akkumuliert werden.

Das Suchproblem für die Ermittlung des nächsten Hindernisses ändert sich jedoch zur Suche eines lokalen Maximums mit *niedrigem* Tiefenwert (zuvor: *hoher* Disparitätenwert). Es gilt für eine Histogrammkategorie  $d_0$  des Tiefenhistogramms  $\mathcal{H}_{\text{dist}}$ :

$$d_0 = \min \{d \mid \mathcal{H}_{\text{dist}}(d) > \vartheta\} \quad .$$

Bei der Auswertung der Tiefen-Histogramme ist jedoch eine Adaption aufgrund der finiten Eingabedaten für das Histogramm notwendig, sofern sich Hindernisse in weiter Entfernung befinden. Durch in der Ferne steigende Ungenauigkeiten in der Rückprojektion der Tiefendaten und die dort zu erwartende dünne Punktdichte sind die Maxima in den Histogrammen potentiell wenig prägnant. Ein Hindernis kann daher ggf. mehreren benachbarten Klassen zugeordnet werden. Wird bei der Suche nach dem lokalen Maxima im Tiefenhistogramm keine Klasse gefunden, so werden in einem zweiten Schritt die benachbarten Klassen mit in die Analyse einbezogen, damit auch weit entfernte Hindernisse trotz der Tiefenungenauigkeit detektiert werden können.

Für diesen Fall werden aufeinanderfolgende Tiefenklassen im Tiefenhistogramm  $\mathcal{H}_{\text{dist}}$  betrachtet und deren Einträge addiert. Wird der Schwellwert  $\vartheta$  erst durch die akkumulierte Betrachtung benachbarter Zellen erreicht bzw. überschritten, muss die Tiefe für den gefundenen Eintrag bestimmt werden. Da es Ziel ist, den freien Raum zu

bestimmen, wird an dieser Stelle der Wert der Klasse verwendet, die einen geringeren Abstand zur Kamera vertritt. Auf diese Weise wird sichergestellt, dass kein zu hoher Tiefenwert – was bedeutet, dass ein Hindernis potentiell zu weit hinten verortet würde – selektiert wird.

Nach Einpassung der Stixel muss der Tiefenwert für den Stixel definiert werden. Die Koordinaten werden zunächst durch Rückprojektion der gefundenen Begrenzungspunkte des Stixels im Bild berechnet. Um den Stixel an die richtige Position zu verschieben, ist eine Skalierung mit der entsprechenden Tiefe notwendig.

Die Tiefe  $d_{\mathbf{o}}$  eines Stixels  $\mathbf{o}$  wird anhand der zugehörigen 3-D-Punkte  $\mathbf{p}_i \in P_{\mathbf{o}}$  bestimmt. Die Punkte  $P_{\mathbf{o}}$  sind diejenigen Punkte, die der Stixel aus Sicht der Kamera abdeckt. Beschreibe  $\mathcal{H}_{\mathbf{o}}$  ein Histogramm über die Abstände  $\|\mathbf{p}_i\|$  der Punkte  $P_{\mathbf{o}}$  zur Kamera. Mit einem adaptiven Schwellwert wird die Histogramm-Klasse mit dem niedrigsten Tiefenwert ermittelt, die genügend zugeordnete Punkte besitzt. Durch dieses Verfahren wird ein aussagekräftiger Tiefenwert für den Stixel bestimmt.

Insbesondere für die Anwendung in der Vogelperspektivenansicht ist es sinnvoll, einzelne Stixel zu Hindernissen zu gruppieren, um diese zusammenhängend darstellen zu können. Bildlich soll aus den einzelnen „Pfählen“, die durch die Stixel repräsentiert werden, ein „Zaun“ konstruiert werden. Dies erfolgt durch Betrachtung der einzelnen Stixel und ihrer jeweiligen geometrischen Nähe. Gegeben eine Menge von Stixeln einer Gruppe, werden diese in einem Nachbearbeitungsschritt miteinander zu einer geometrischen Einheit verbunden, sofern ihr Abstand einen Schwellwert nicht überschreitet.

Der Ablauf des Verfahrens mit den vorgestellten Erweiterungen ist in Abbildung 5.47 zu sehen. Die Abbildung skizziert den Ablauf von Stereo-Daten bis hin zu den ermittelten Hindernisbegrenzungen. Die berechneten Hindernisse werden in das System für die Berechnung der dichten, perspektivisch korrekten Ansicht aus der Vogelperspektive integriert und entsprechend in die 3-D-Visualisierung und die orthographische Texturschicht als Hinweislinie projiziert.

### 5.8.2 Testergebnisse

Das Verfahren zur iterativen Zellunterteilung und das zur erweiterten Stereo-Hinderniserkennung werden zusammen mit dem Verfahren zur Erzeugung der dichten, perspektivisch korrekten Ansicht aus der Vogelperspektive fusioniert. Im Folgenden werden Beispiele für mit dem Verfahren berechnete Ansichten vorgestellt. Als Datenbasis wird wie bei den in Kapitel 5.6.4 aufgezeigten Ergebnissen die *KITTI Odometry Datasets* [GLU12] verwendet. Dieser Datensatz beinhaltet Stereokameras, die die Horizont-Bedingung für das Verfahren nach Badino, Franke und Pfeiffer [BFP09] erfüllen. Nichtsdestotrotz kann die Adaption des Verfahrens wie vorgestellt mit dem Datensatz ge-

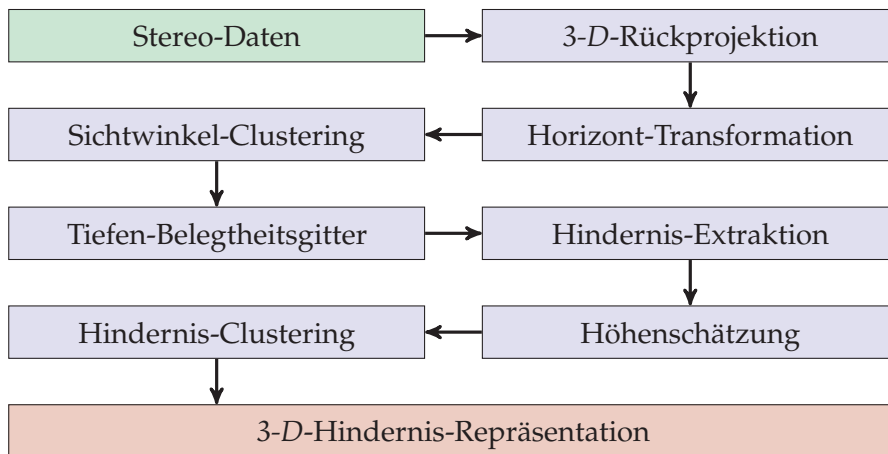


Abbildung 5.47: Komponentenübersicht über die erweiterte Stereo-Hinderniserkennung

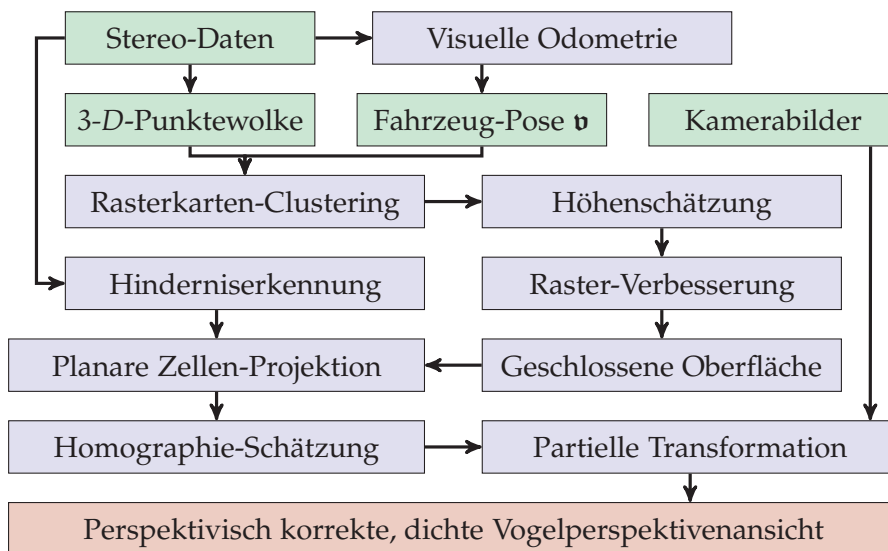


Abbildung 5.48: Übersicht über die Systemkomponenten und den Datenfluss für die Berechnung von perspektivisch korrekten Ansichten aus der Vogelperspektive mit iterativer Oberflächenverbesserung und erweiterter Stereo-Hinderniserkennung

testet werden. Es ist kein öffentlicher Datensatz mit Aufnahmen aus Fahrzeugen und zum Boden hin geneigten Kameras bekannt. Die Neigung der Kameras wird mit dem in Kapitel 5.9 vorgestellten, selbst aufgenommenen Datensatz getestet.

Abbildung 5.48 zeigt die einzelnen Systemkomponenten inklusive der Erweiterungen und dem darin stattfindenden Datenfluss. Die verwendeten Parameter für die Rasterkarte und die Texturauflösung entsprechen dabei denen aus Kapitel 5.6.4.

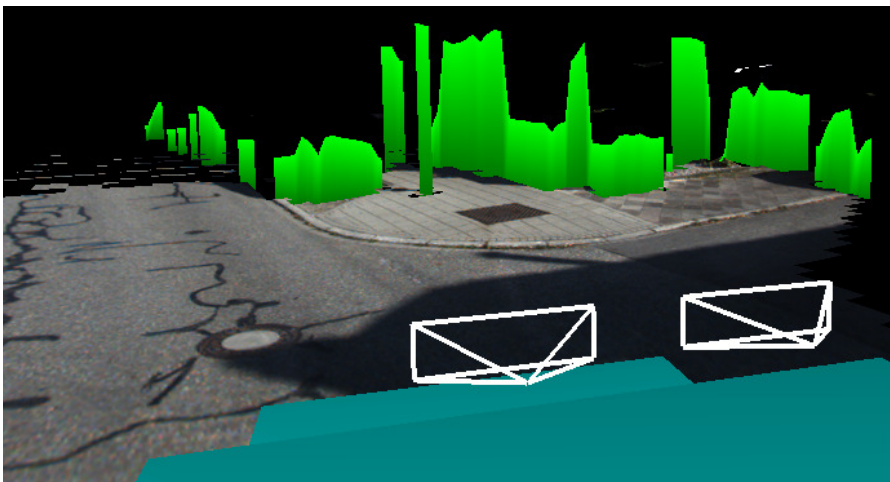
Repräsentative Ergebnisse für das vorgestellten Verfahren sind in den Abbildungen 5.49, 5.50 und 5.51 zu sehen.

In allen Beispielen sind die Umrisse der Hindernisse in der Szene klar zu erkennen. Sowohl Häuser, als auch Fahrzeuge und Straßenmöbel sind eindeutig zu erkennen und entsprechend hervorgehoben. Sogar die vergleichsweise sehr schmale Straßenlaterne in Abbildung 5.49 wird detektiert. Die beiden Fahrradfahrer in Abbildung 5.50 sind korrekt lokalisiert und der von ihnen eingenommene Bereich ist markiert. Dabei ist eine zackig anmutende Kontur zu sehen, die durch die Form der Räder zu erklären ist. Ein besonderer Vorteil der erweiterten Stereo-Hinderniserkennung lässt sich zudem in Abbildung 5.51 erkennen. Dort ist ein im Schatten parkendes Fahrzeug durch die Hinderniserkennung detektiert. Durch den dunklen Boden kommt der eingezeichneten Kontur eine besondere Bedeutung zu. In diesem Fall wäre das Auto aufgrund der perspektivisch korrekten Abbildung ohne eine zusätzliche Markierung nur sehr schwer zu erkennen.

Das vorgestellte Verfahren zeigt neben einer hohen Bildqualität einen Mehrwert hinsichtlich des erweiterten Bodenbereichs und der Hervorhebung von Hindernissen. Die Markierung von potentiellen Gefahrenstellen kann einen wertvollen Mehrwert für die sichere Navigation eines Fahrzeugs darstellen. Die vorgestellten Erweiterungen sind eine Verbesserung des in Kapitel 5.6 vorgestellten Verfahrens.



(a) Bodentextur (orthographisch). Die roten Linien markieren die gefundenen Hindernisse.

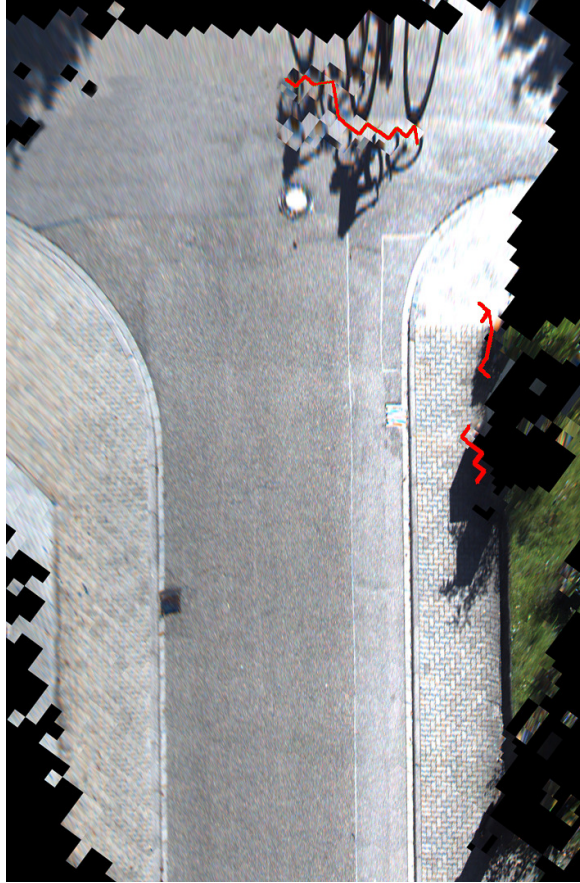


(b) 3-D-Ansicht mit Fahrzeugpose. Die grünen Wände markieren die ermittelten Hindernisgrenzen.

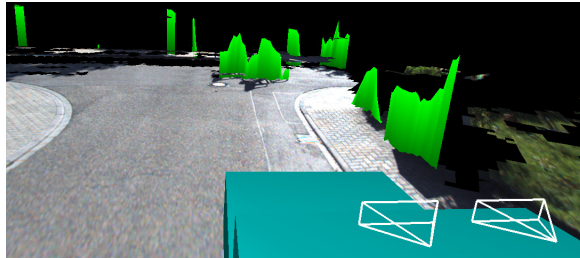


(c) Szenenbild („KITTI Odometry Dataset“ #1, Bild 102 [GLU12])

Abbildung 5.49: Perspektivisch korrekte, dichte Ansicht aus der Vogelperspektive. *KITTI Odometry Dataset* #1.



(a) Bodentextur (orthographisch). Die roten Linien markieren die gefundenen Hindernisse.



(b) 3-D-Ansicht mit Fahrzeugpose. Die grünen Wände markieren die ermittelten Hindernisgrenzen.

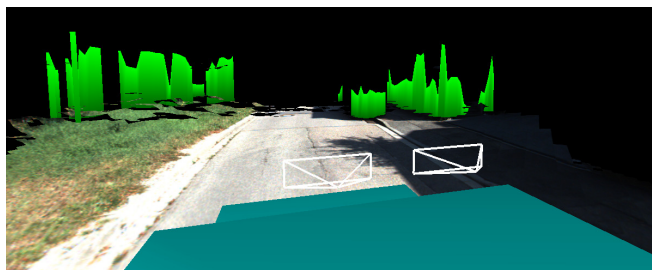


(c) Szenenbild („KITTI Odometry Dataset“ #5, Bild 2070 [GLU12])

Abbildung 5.50: Perspektivisch korrekte, dichte Ansicht aus der Vogelperspektive. *KITTI Odometry Dataset* #5.



(a) Bodentextur (orthographisch). Die roten Linien markieren die gefundenen Hindernisse.



(b) 3-D-Ansicht mit Fahrzeugpose. Die grünen Wände markieren die ermittelten Hindernisgrenzen.



(c) Szenenbild („KITTI Odometry Dataset“ #3, Bild 316 [GLU12])

Abbildung 5.51: Perspektivisch korrekte, dichte Ansicht aus der Vogelperspektive. *KITTI Odometry Dataset* #3.

## 5.9 ANWENDUNGSSZENARIO UND TESTDATENSATZ

Das vorgestellte Verfahren zur Erzeugung einer perspektivisch korrekten Ansicht aus der Vogelperspektive wird im Folgenden in einem Anwendungsszenario getestet. Das Szenario entstammt dem in Kapitel 1 vorgestellten Projekt. Für die Aufnahme von Testdaten, die dem Projektszenario entsprechen, wird ein Testfahrzeug mit mehreren Stereo-Paaren ausgestattet. Das Projekt hat es zum Ziel, insbesondere den *vorderen* Fahrzeugbereich abzudecken.

Für den Testaufbau sind drei Stereokameras (Stereoverbünde mit  $n = 1$ ) vorgesehen, die im vorderen Bereich des Fahrzeugs montiert sind. Dabei wird eine Stereokamera in Dachhöhe mittig in Fahrtrichtung angebracht. Die Blickrichtung der Kamera ist dabei leicht in Richtung des Bodens geneigt (ca. 20 Grad). An der linken und rechten Seite werden zur jeweiligen Seite blickende Kameras angebracht, die den vorwärtigen Bereich mit erfassen. Diese Kameras sind ebenfalls auf Dachhöhe montiert und in der Flucht der mittigen Kamera angebracht. Die Anordnung der Kameras auf dem Testfahrzeug ist in Abbildung 5.52 schematisch dargestellt.

Für den Aufbau kommen unterschiedliche Kameras zur Anwendung. Die mittlere Stereokamera besteht aus zwei industriellen Bildverarbeitungskameras und wird daher als primäre Stereokamera definiert. Die beiden Einzelkameras sind Hardware-seitig getriggert, so dass synchronisierte Bilder aufgenommen werden. An der linken und rechten Seite der Fahrzeugfront werden preisgünstigere Stereokameras verbaut. Dieser Kameratyp besteht aus einer Einheit von zwei Bildsensoren, deren Bilder jeweils gemeinsam aufgenommen werden können.

Die Synchronisierung zwischen den einzelnen Stereokameras erfolgt durch bei der Aufnahme erzeugte Zeitstempel (Software-seitig).

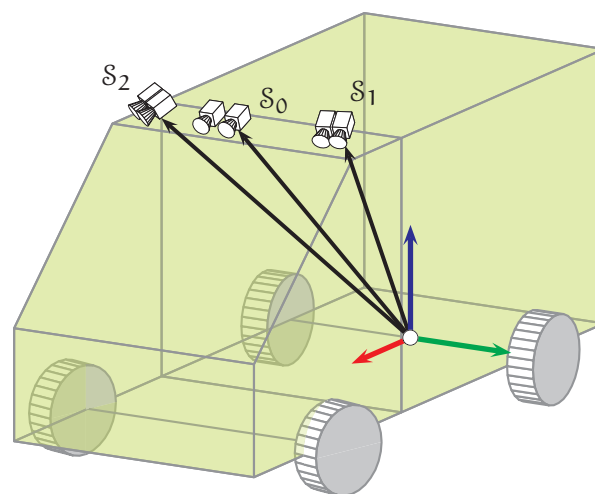


Abbildung 5.52: Anordnung der Sensoren auf dem Testfahrzeug



Der Aufbau auf dem Testfahrzeug ist in Abbildung 5.53 zu sehen.

Im Kontext des in Kapitel 5.3 beschriebenen Modells werden drei Stereoverbünde  $S_i$  mit  $i \in \{1, 2, 3\}$  eingesetzt. Jeder der Stereoverbünde  $S_i$  stellt dabei den Sonderfall einer Stereokamera ( $n = 1$ ) dar. Die verwendeten Kameras sind in Tabelle 5.1 aufgeführt.

Die jeweiligen intrinsischen Kameraparameter  $J$  der einzelnen Kameras wurden mit den in Kapitel 3.2 vorgestellten Verfahren ermittelt. Die extrinsische Kalibrierung der Aufnahmen wurde mit einem in der Arbeitsgruppe Aktives Sehen der Universität Koblenz-Landau entwickelten Software durchgeführt.

Für die vorliegende Problemstellung ist es nicht möglich, eine fundierte Grundwahrheit für die quantitative Evaluation des Verfahrens zu definieren, da ein Referenzsystem weder bekannt noch verfügbar ist. Um einen visuellen Vergleich der Ergebnisse zu ermöglichen, wird daher eine Drohne eingesetzt, die eine Vogelperspektive parallel zur Aufnahme der Testdatensätze ermöglicht. An der Drohne wird dabei eine zum Boden blickende Kamera (*GoPro HERO7 Black*) befestigt, die eine entsprechende Ansicht nach unten aufzeichnet. Diese Kamera kann selbstverständlich lediglich eine *perspektivisch* projizierte Ansicht aus der Vogelperspektive einfangen, die aber eine gute Annäherung darstellt. Die Kamera an der Drohne ist intrinsisch kalibriert (siehe Kapitel 3.2) und die Aufnahmen sind mit den ermittelten Parametern entzerrt. Eine Synchronisierung der Drohnen-Referenzbilder zum Vergleich sind manuell ausgewählt.

Der Testdatensatz beinhaltet zwei zentrale Szenarien. Auf der einen Seite Aufnahmen eines Parkplatzgeländes, auf der anderen Seite Aufnahmen eines Feldwegs.

Nicht alle Stereokameras werden für alle Komponenten verwendet, da die eingesetzten Kameras eine unterschiedliche Aufnahmequalität haben. Die Stereokamera  $S_0$  zeichnet qualitativ sehr hochwertige Daten mit großer Schärfe auf. Die Bilder der Kamera sind jedoch in Graustufen (siehe Tabelle 5.1). Diese Stereokamera wird primär als Datenquelle für die visuelle Odometrie, die erweiterte Stereo-

#	Kameras	Konfiguration
$S_0$	2x PointGrey Blackfly BFLY-U3-23S6M Graustufen	2x 1920 px × 1200 px, 15 Hz
$S_1$	Stereolabs ZED RGB/Farbe	2x 1280 px × 720 px, 15 Hz
$S_2$	Stereolabs ZED RGB/Farbe	2x 1280 px × 720 px, 15 Hz

Tabelle 5.1: Stereoverbünde für die Testaufnahmen



(a) Testfahrzeug



(b) Stereokameras auf dem Fahrzeug

(c) Nahaufnahme der zentralen Stereokamera  $S_0$ 

Abbildung 5.53: Testfahrzeug und Testaufbau für die Aufnahme des Datensatzes

Konf.	$S_0$				$S_1$				$S_2$			
	O	H	M	T	O	H	M	T	O	H	M	T
I	✓	✓	✓	✓	-	-	-	-	-	-	-	-
II	✓	✓	✓	-	-	✓	✓	✓	-	✓	✓	✓

Abkürzung    Komponente

O            Visuelle Odometrie

H            Erweiterte Stereo-Hinderniserkennung

M            Geschlossenes Oberflächenmodell

T            Oberflächentexturierung

Tabelle 5.2: Verwendung der einzelnen Stereokameras in den beiden vorgestellten Konfigurationen

Hinderniserkennung und zur Erzeugung des geschlossenen Oberflächenmodells verwendet.

Die beiden Stereokameras  $S_1$  und  $S_2$  bestehen aus vergleichsweise kostengünstigen Kamerasensoren und Optiken, die eine geringere Qualität in der Bildaufnahme aufweisen (siehe Tabelle 5.1). Nichtsdestotrotz stellen diese Kameras Farbbilder (*RGB*) bereit. Die Daten der beiden Kameras werden für die erweiterte Stereo-Hinderniserkennung, die Generierung des geschlossenen Oberflächenmodells und dessen Texturierung verwendet. Bauartbedingt nehmen die Stereokameras  $S_1$  und  $S_2$  eine automatische Bildanpassung hinsichtlich der Helligkeit und des Weißabgleichs vor, die sich nicht zwischen den Stereokameras synchronisieren lässt. Dies führt zu leichten Verschiebungen in der Farb- und Helligkeitsdarstellung. Dieser Effekt bzw. dessen Neutralisierung wird an dieser Stelle nicht weiter untersucht.

Ziel dieser Kombination ist es, eine hochwertige Kamera für die Ermittlung der Odometrie und Geometrie zur Verfügung zu haben, jedoch die Erweiterung des Sichtfeldes sowie die Farbinformation durch günstigere Sensoren zu ermöglichen.

Die vorgestellten Ergebnisse auf dem eigens aufgenommenen Testdatensatz sind in unterschiedlichen Konfigurationen berechnet. Die Konfigurationen sind in Tabelle 5.2 ersichtlich. Für Konfiguration I wird nur die zentrale Stereokamera  $S_0$  verwendet, d. h. in dieser Konfiguration wird nur die Stereokamera mit den besten Sensor- und Optikeigenschaften genutzt. In Konfiguration II werden alle drei Stereokameras eingesetzt, wobei nur die Farbkameras zur Texturierung verwendet werden. Dadurch wird die Bereichsabdeckung vergrößert und zudem ist eine Texturierung mit Farbdaten möglich, wenngleich die lokale Texturqualität niedriger ist. Die Parametrierung des ge-

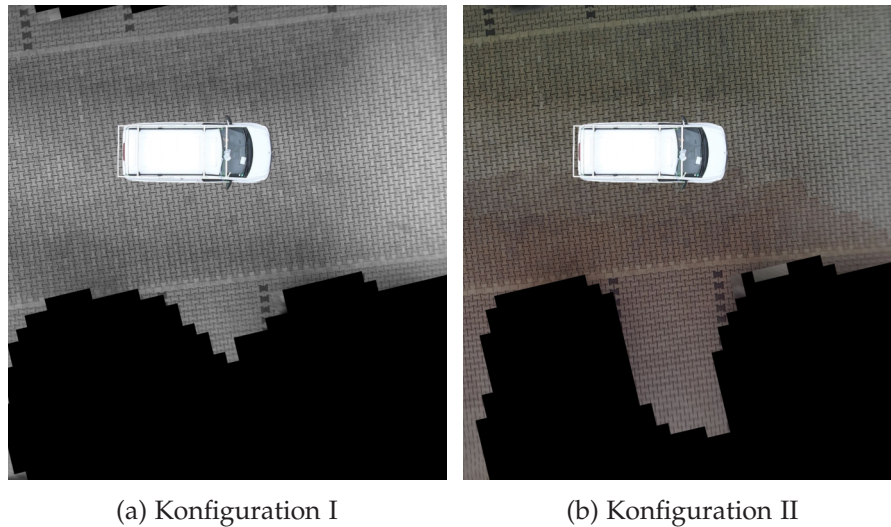


Abbildung 5.54: Szenenausschnitt einer Vorbeifahrt an einer einzelnen leeren Parklücke (ohne Hindernisvisualisierung)

schlossenen Oberflächenmodells bzw. der Texturschicht entspricht der in Kapitel 5.6.4.

Für die Darstellung der perspektivisch korrekten Ansicht aus der Vogelperspektive wird eine Ansicht des Testfahrzeugs von oben integriert. Diese wird an korrekter Position in der Vogelperspektivenansicht eingeblendet. Die Skalierung ist dabei entsprechend den Parametern der Rasterkarte und im Hinblick auf die extrinsische Kamerakalibrierung und die Auflösung der Textur angepasst. Dies ermöglicht eine natürliche Ansicht und hilft, die geometrischen Proportionen einzuschätzen. Die Ansicht des Fahrzeugs von oben wurde dabei mittels der Drohne aufgenommen und als freigestellte Grafik aufbereitet. Die Einblendung erfolgt direkt durch den Algorithmus.

Für die Darstellung in der vorliegenden Arbeit werden die Ansichten aus der Vogelperspektive bewusst mit der Fahrzeugorientierung nach rechts (d. h. scheinbar nach rechts fahrend) verwendet, um den Platz bestmöglich zu nutzen. Intuitiv ist selbstverständlich eine Ausrichtung des Fahrzeugs nach oben sinnvoll (d. h. Rotation um 90 Grad nach links im Vergleich zur abgedruckten Ausrichtung) und ist für den Fahrer im Fahrzeug natürlicher.

Abbildung 5.54 zeigt ein Beispiel für die bessere Bereichsabdeckung durch den Einsatz aller Stereokameras in Konfiguration II. Durch mehrere Sensoren kann die leere Parklücke dargestellt werden (siehe Abbildung 5.54b). Die Ansicht mit Konfiguration I (siehe Abbildung 5.54a) kann nur eine eingeschränkte Darstellung der Parklücke aufgrund von Geometrieinschränkungen erreichen.

Die Genauigkeit der Projektion auf das geschlossene Oberflächenmodell ist in Abbildung 5.55 verdeutlicht. Die Ausschnittsvergröße-

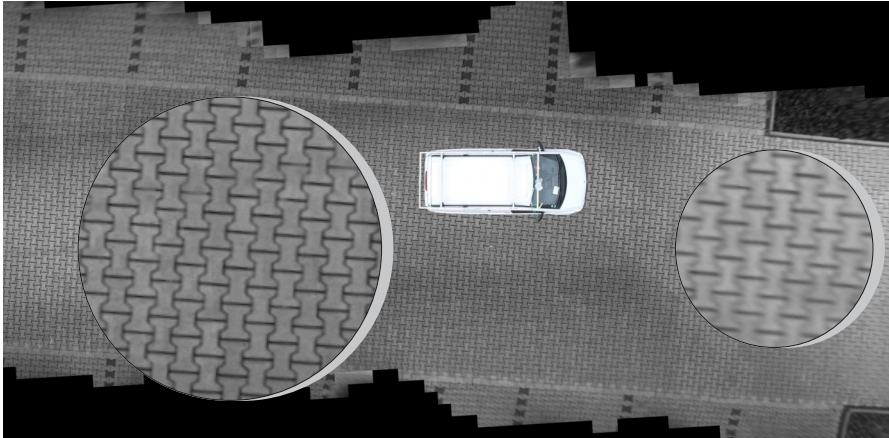
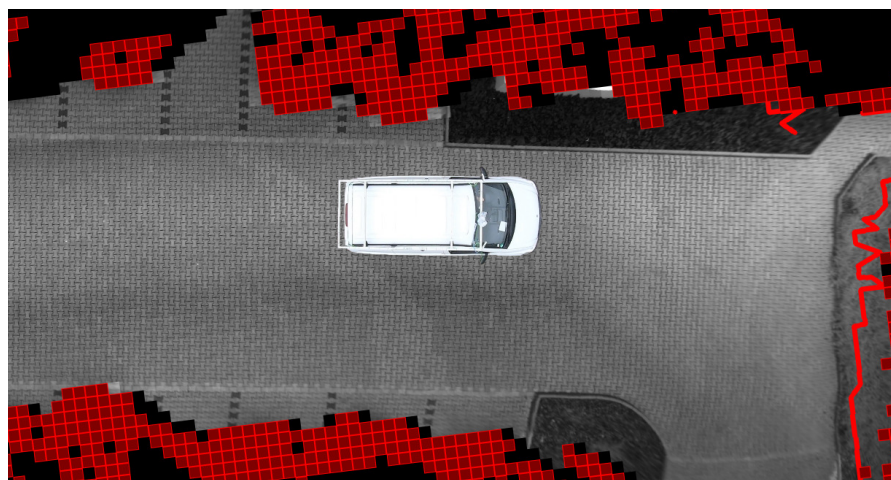


Abbildung 5.55: Ausschnittsvergrößerungen der Bodentextur (Konfiguration I)

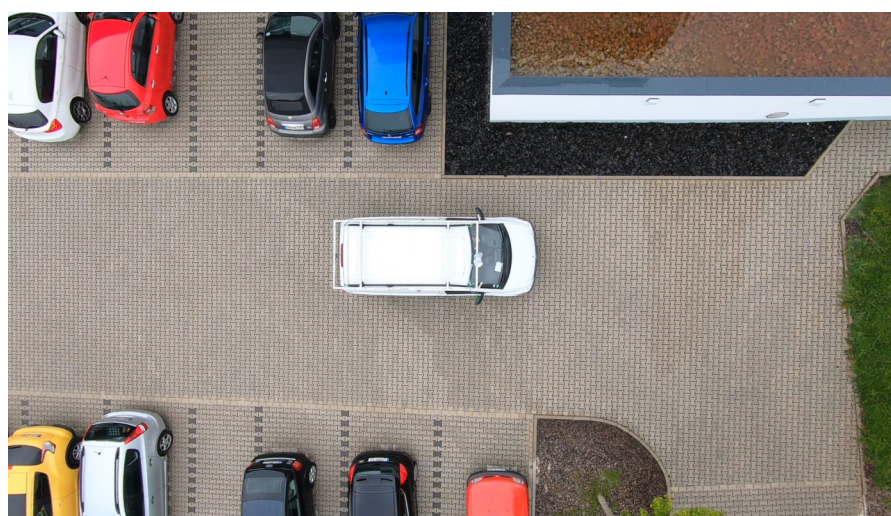
rungen zeigen die erreichte Qualität im Hinblick auf die Genauigkeit der Abbildung und die Texturqualität. Der linke Ausschnitt zeigt einen Bereich hinter dem Fahrzeug. Das bedeutet, dass dieser Bereich mit Bildern hoher lokaler Auflösung (nah zur Kamera) bestückt ist. Der rechte Ausschnitt zeigt einen Bereich vor dem Fahrzeug in größerer Entfernung. Hier kann geometrisch bedingt (noch) keine vergleichsweise so hochauflösende Bodentextur wie für den linken Ausschnitt verfügbar sein, da das Fahrzeug bzw. die Kameras zu weit entfernt sind. Nichtsdestotrotz erfolgt eine korrekte Abbildung, die die Bodenstruktur klar erkennen lässt.

Abbildung 5.56 zeigt die berechneten Ansichten mit Konfiguration I (5.56a) und Konfiguration II (5.56c) sowie ein vergleichbares Bild der Drohnenkamera (5.56b). Auch hier ist die vergrößerte Bereichsabdeckung durch den Aufbau mit drei Stereokameras erkennbar (Parkplatz am unteren Bildrand). Der Grünstreifen am jeweils rechten Bildrand wird durch die erweiterte Stereo-Hinderniserkennung erkannt und entsprechend im Bild markiert. Aufgrund der Vegetation und den damit einhergehenden inhomogenen Tiefenwerten ist die resultierende Hindernislinie leicht gezackt. Der Grünstreifen verfügt über eine Steigung und wird durch das geschlossene Oberflächenmodell approximiert. Eine 3-D-Ansicht dieses Bereichs ist in Abbildung 5.57 zu sehen.

Eine perspektivisch korrekte, dichte Vogelperspektivenansicht ist in Abbildung 5.58 zu sehen. Das Fahrzeug passiert in dieser Szene ein Feld mit hohen Pflanzen auf der rechten Seite des Fahrzeugs. In der auf Konfiguration II basierenden Ansicht ist der Feldweg aufgrund der Farbinformation besser zu erkennen. Die zusätzlichen Stereo-Daten zu den Seiten hin ermöglichen zudem eine bessere Modellierung mit dem geschlossenen Oberflächenmodell. In der Darstellung auf Basis von Konfiguration I kann in den seitlichen Bereichen weni-



(a) Konfiguration I



(b) Drohnenaufnahme



(c) Konfiguration II

Abbildung 5.56: Vergleichsansicht mit beiden Konfigurationen und einer Drohnenaufnahme (Parkplatz)

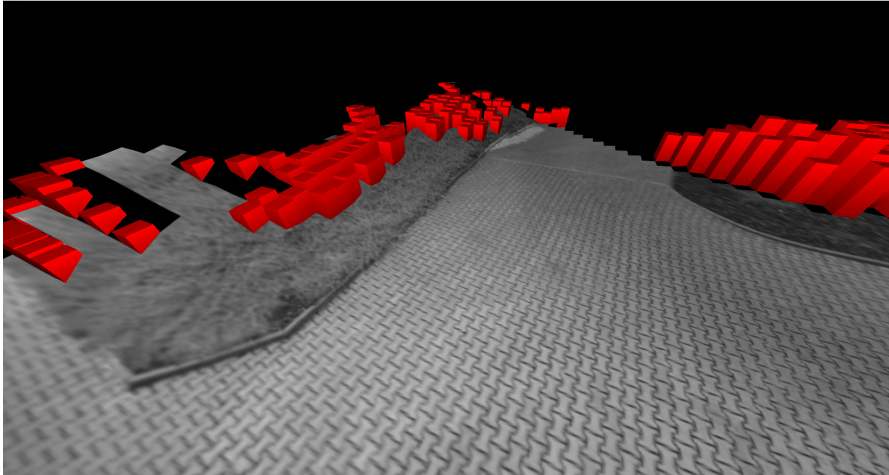


Abbildung 5.57: 3-D-Ansicht des geschlossenen Oberflächenmodells im Bereich des Grundstreifens für die Abbildung 5.56. Die roten Säulen markieren als *Hindernis* klassifizierte Zellen mit entsprechender Höhenschätzung.

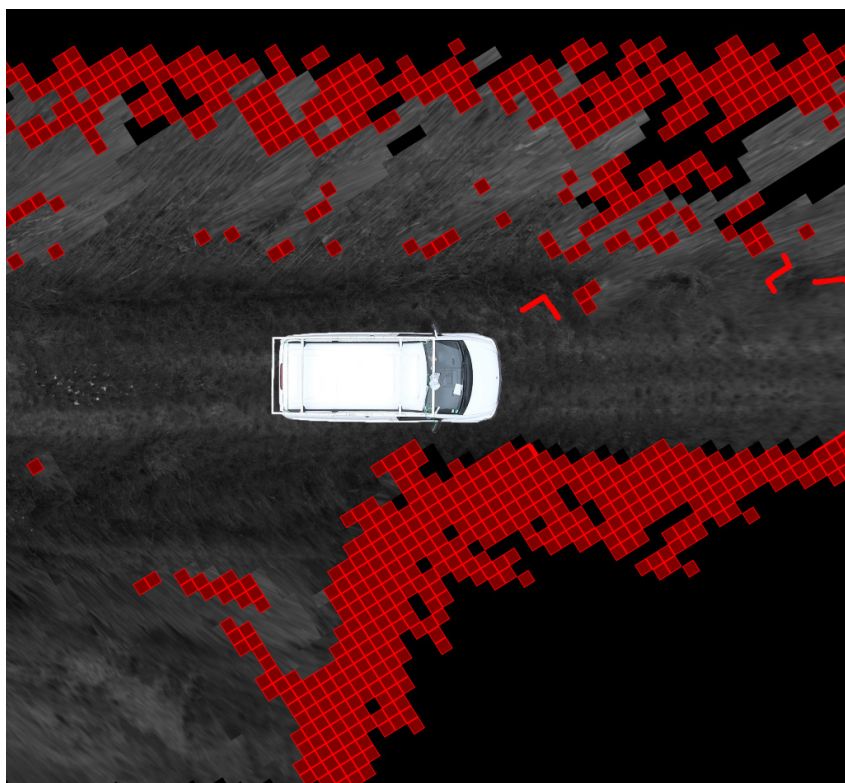
ger Präzise approximiert werden (z. B. links vom Fahrzeug).

Abbildung 5.59 zeigt zwei Kamerabilder, die beim Verständnis der Szenengeometrie unterstützen sollen. Diese beiden Bilder (5.59a, 5.59b) entstammen dabei früheren Zeitschritten im Vergleich zu den in Abbildung 5.58 dargestellten Ansichten aus der Vogelperspektive. Dies ist notwendig, damit die Feldgeometrie sichtbar ist. Aus derselben Szene ist das geschlossene Oberflächenmodell mit seiner Geometrie (Visualisierung der Steigung durch Einfärbung (5.59c)) und angewandter Texturschicht (5.59d) – am Beispiel des unebenen Feldwegs – zu sehen.

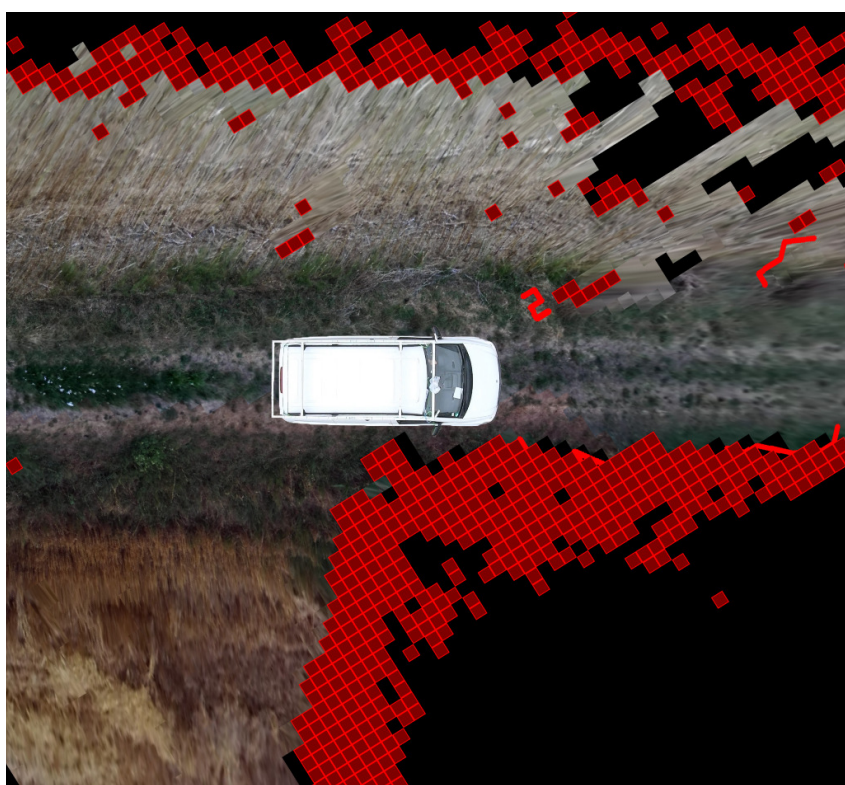
Eine weitere Vogelperspektivenansicht auf einem Feldweg ist in Abbildung 5.60 dargestellt. Das Hindernis am oberen Rand ist durch den Strohhallen auf dem Feld bedingt. In der linken unteren Ecke ist eine Szenenansicht in Form eines Kamerabildes der Hauptkamera der linken Stereokamera  $S_1$  zu sehen. Das Kamerabild ist eine Aufnahme aus einem früheren Zeitschritt, in dem der Strohhallen zu sehen ist.

Die berechnete Vogelperspektive einer Kurvenfahrt (Rechtskurve) im Vergleich mit einer Drohnenaufnahme ist in Abbildung 5.61 zu sehen. Die mit Konfiguration I (5.61a) berechnete Ansicht kann durch den Einsatz von nur einer Stereokamera nur einen kleineren Umgebungsausschnitt abdecken. Die Verwendung von mehreren Stereokameras in Konfiguration II (5.61c) erweitert den sichtbaren Bereich. Die Farbinformation unterstützt zudem die Unterscheidbarkeit zwischen Feld und Weg. Der in der Drohnenansicht (5.61c) sichtbare Baum wird in beiden Konfigurationen erkannt und als Hindernis markiert.

Eine Gegenüberstellung zwischen dem initial entwickelten punkt-basierten Verfahren und Verfahren unter Verwendung des geschlosse-



(a) Konfiguration I



(b) Konfiguration II

Abbildung 5.58: Berechnete Vogelperspektive auf einem Feldweg entlang eines Feldes auf der rechten Fahrzeugseite

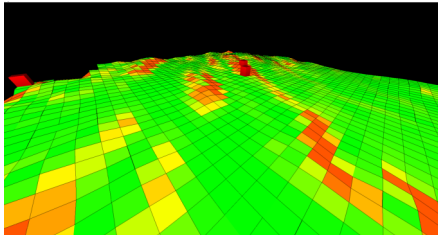




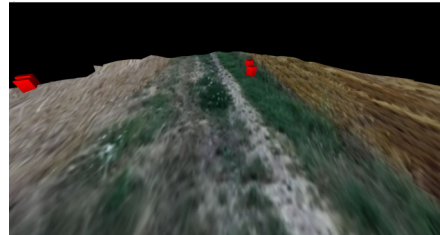
(a) Szenenansicht der Stereo-Hauptkamera von  $S_0$



(b) Szenenansicht der Stereo-Hauptkamera von  $S_2$



(c) 3-D-Ansicht des Oberflächenmodells

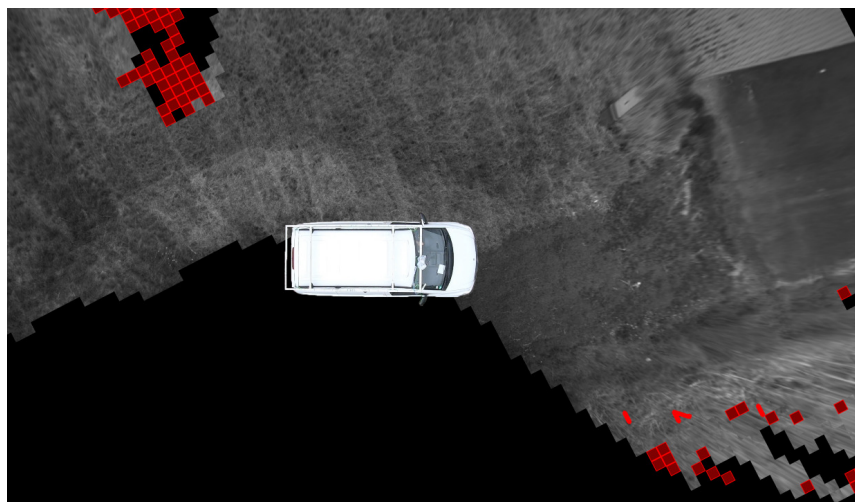


(d) Texturierte 3-D-Ansicht des Oberflächenmodells

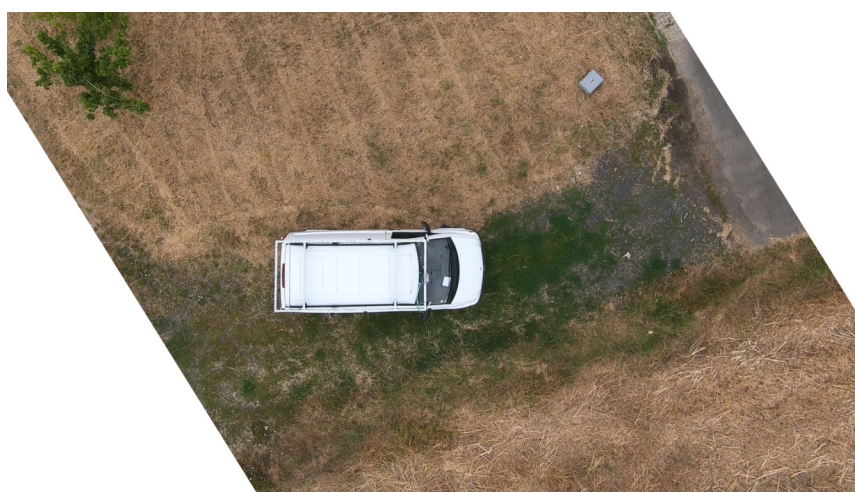
Abbildung 5.59: Szenenvisualisierungen (5.59a, 5.59b) der Szene aus Abbildung 5.58 aus vorangegangenen Zeitschritten und Beispiel für die Bodenapproximation mit dem geschlossenen Oberflächenmodell auf dem Feldweg (5.59c, 5.59d)



Abbildung 5.60: Vogelperspektivenansicht auf einem Feldweg. Unten links: Kameraansicht (früherer Zeitschritt mit sichtbarem Strohballen).



(a) Konfiguration I



(b) Drohnenaufnahme (rotierter Ausschnitt)

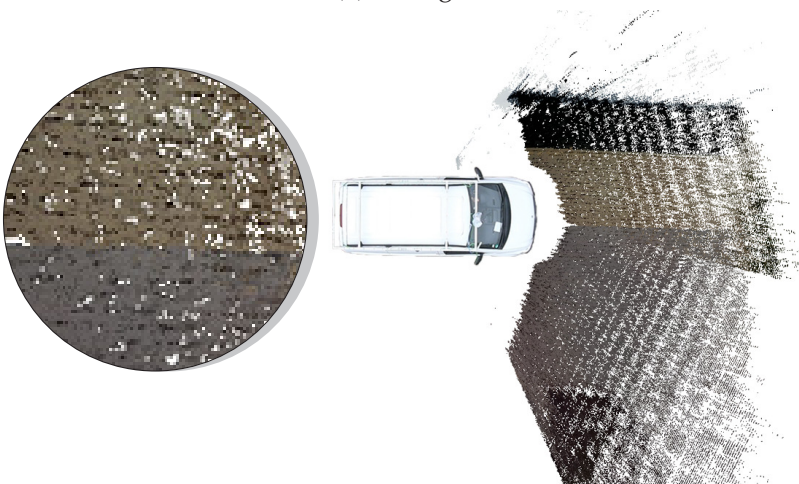


(c) Konfiguration II

Abbildung 5.61: Vergleichsansicht mit beiden Konfigurationen und einer Drohnenaufnahme (Feldweg)



(a) Konfiguration II



(b) Punktbasiertes Verfahren (siehe Kapitel 5.5)

Abbildung 5.62: Vergleichsansicht zwischen dem punktbasierten Verfahren und dem Verfahren unter Verwendung des geschlossenen Oberflächenmodells. Die Ausschnittsvergrößerung zeigt jeweils einen Bereich vor dem Fahrzeug.

nen Oberflächenmodells ist in Abbildung 5.62 zu sehen. Die Eingabedaten für die beiden dargestellten Varianten sind äquivalent. Analog zu den Ergebnissen in Kapitel 5.5.4 zeigen die Ansichten eine vergleichsweise geringe Punktdichte. Die Auflösung der virtuellen Ansicht ist dabei höher als bei den Ergebnissen in Kapitel 5.5.4 und an die Auflösung des Verfahrens auf Basis des Oberflächenmodells angepasst. Die Fahrbahngrenzen lassen sich auch in der Ansicht des punktbasierten Verfahrens erkennen – allerdings in schlechterer Qualität. Wie in der Vergrößerungsansicht dargestellt, lässt sich die Pflastersteinstruktur nicht erkennen. In Abbildung 5.62a ist diese klar erkennbar. Selbstverständlich ist der darstellbare Bereich im punktbasierten

sierten Verfahren auf die Daten des aktuellen Zeitschritts begrenzt, da keine zeitliche Integration erfolgt. Nichtsdestotrotz werden mehrere Stereokameras (in diesem Fall  $S_1$  und  $S_2$ ) verwendet und fusioniert. Mit Abbildung 5.62 wird die Überlegenheit des geschlossenen Oberflächenmodells gegenüber dem punktbasierten Verfahren dokumentiert.

## FAZIT

---

Die in den vorangegangenen Kapiteln vorgestellte Arbeit befasst sich mit zwei Themengebieten. Das erste Themenfeld der Arbeit befasst sich mit der Gespannstatuserkennung artikulierter Fahrzeuge.

In diesem Bereich wurde ein vorhandenes und patentiertes Verfahren reimplementiert und mittels einer eigens geschaffenen Simulationsumgebung evaluiert. Dabei wurde eine Software-Architektur, die über entsprechende Schnittstellen einen flexiblen Einsatz und Tausch der einzelnen Komponenten ermöglicht, entwickelt. Mit der Intention auch Fahrzeugartikulationen auf unebenem Grund korrekt erkennen zu können, wurde ein Modell der in diesem Fall auftretenden Artikulationen erstellt. Mit einer Vereinfachung des generischen Modells (Reduzierung auf vier Winkel) wurde die Rekonstruktion der Fahrzeugartikulation ermöglicht.

Das geschaffene *Software-in-the-Loop*-System ermöglicht ein flexibles und gut zu reproduzierendes Testen des Gesamtsystems. Um die Stabilität und Genauigkeit der Gespannstatuserkennung zu erhöhen, wurde mit einem Kalman-Filter ein Tracking-Mechanismus integriert. Dieser nutzt Bildsequenzen und erhöht dadurch die Präzision der Schätzung. Die drei Verfahren wurden gegeneinander mit der *Software-in-the-Loop*-Simulationsumgebung evaluiert.

Um die Auswirkung der Präzision der drei unterschiedlichen Verfahren auf ein Standard-Fahrmanöver beurteilen zu können, wurde ein erweitertes Simulationssystem geschaffen, das ein einfaches Fahrzeugmodell, eine virtuelle Fahrerkomponente sowie ein Assistenzsystem beinhaltet, das mit der Statuserkennung zusammen zu einem weiteren *Software-in-the-Loop*-System verknüpft wird.

Methodisch wurde in diesem Themengebiet zunächst die Vorarbeit reimplementiert und mittels eines *Software-in-the-Loop*-Ansatzes evaluiert und auf Szenarien mit unebenem Grund erweitert. Die Testszenarien wurden so gewählt, dass eine Vergleichbarkeit der jeweiligen Verfahren erhalten bleibt.

Zur Fortführung der Arbeiten an diesem Themenbereich wäre eine Evaluation unter Verwendung eines großen Nutzfahrzeuggespanns wünschenswert. Dieses Gespann sollte mit Drehschemelanhänger und einem entsprechenden Referenzmesssystem, das eine hochpräzise Aussage über den Gespannzustand als Grundwahrheit liefern kann, ausgestattet sein. Im Rahmen der Arbeit war weder ein großes Nutzfahrzeug mit Drehschemelanhänger noch ein entsprechendes Referenzmesssystem verfügbar.

Das zweite Themenfeld befasst sich mit der Erzeugung dichter, per-

spektivisch korrekter Ansichten aus der Vogelperspektive. Die verfügbaren Systeme zeigen nur für Bereiche akzeptable Ansichten, die sich auf dem als planar angenommenen Boden befinden. Bereiche wie Hindernisse, denen in der Anwendung besonderes Interesse zukommt, können nur unnatürlich verzerrt dargestellt werden.

Nach einer initialen Untersuchung von Veröffentlichungen und kommerziellen Systemen wurde die Ursache für diese Verzerrungen identifiziert und mit dem Homographie-Verschattungseffekt beschrieben. In einem ersten Verfahren, das auf aus Stereo-Daten rekonstruierten Punktwolken beruht, kann perspektivische Korrektheit erreicht werden. Die erreichte Bildqualität ist auf hochpräzisen Stereo-Datensätzen durchaus vielversprechend. Mit Daten „einfacherer“ Stereokameras, wie sie in Fahrzeugen eingesetzt werden, kann zwar eine perspektivische Korrektheit erreicht werden, jedoch nur eine weniger zufriedenstellende Bildqualität.

Mit der Entwicklung des auf Rasterkarten beruhenden geschlossenen Oberflächenmodells wird eine Geometriemodellierung der Umgebung vorgenommen, die u. a. unebenen Grund korrekt abbilden kann. Vorteil dieses Modells ist die partielle Planarität der Oberflächenelemente. Diese Planarität bzw. lokal planare Approximation der Umgebung kann genutzt werden, um die Bereiche partiell korrekt mit einer Ebenenannahme zu texturieren. Dadurch kann eine perspektivisch korrekte *und* dicht texturierte Ansicht generiert werden.

Das Oberflächenmodell liefert jedoch durch die Rasterkarte bedingt nur eine diskrete Approximation der Oberfläche durch die einzelnen Zellen. Um dadurch bedingte Einschränkungen bei der vollständigen Abdeckung der Bodenfläche zu minimieren, wurde die iterative Zellverbesserung entwickelt, die die visualisierbare Bodenfläche maximiert.

In einer perspektivisch korrekten Ansicht aus der Vogelperspektive sind senkrecht zum Boden verlaufende Hindernisse – wie beispielsweise Wände – nur schwer zu erkennen, da sie potentiell auf nur einen Punkt bzw. eine Linie abgebildet werden. Um diesem Problem zu begegnen, wurde eine Stereo-Hinderniserkennung in das System integriert.

Die Stereo-Hinderniserkennung erweitert dabei ein etabliertes Verfahren für das konkrete Szenario. Im ursprünglichen „Stixel World“-Verfahren nutzen die Autoren eine implizite Blickrichtung der Kameras in Richtung des Horizonts. Dies ist im Kontext der Vogelperspektive jedoch nicht gegeben. Daher wird die im Verfahren enthaltene Voraussetzung der Kameraausrichtung durch Erweiterung des Verfahrens entfernt bzw. neutralisiert. Im Resultat wurde eine erweiterte Stereo-Hinderniserkennung entwickelt, die generischer einsetzbar ist. Das Verfahren wurde in die Vogelperspektivenansicht integriert, so dass Hindernisse entsprechend gekennzeichnet werden können.

Die Übertragbarkeit der zuvor auf öffentlich verfügbaren Daten-

sätzen entwickelten Verfahren auf andere Szenarien ist anhand des eigens aufgenommen Datensatzes dokumentiert. Dabei wird auch die Integration mehrerer Stereokameras für das Verfahren getestet. Da kein entsprechendes Referenzsystem verfügbar oder bekannt ist, werden Drohnenaufnahmen angefertigt und zum Vergleich zu den berechneten Ansichten exemplarisch gezeigt. Die Kombination von günstigen mit einer industriellen Stereokameras stellt eine kostengünstige Variante dar. Allerdings führt auch die automatische Bildanpassung der günstigen Kameras zur Farb- und Helligkeitsartefakten, die beispielsweise mit entsprechenden Algorithmen ausgeglichen werden könnten. Diese Thematik ist jedoch nicht Bestandteil der vorliegenden Arbeit.

Methodisch wird im zweiten Großbereich der Arbeit initial eine Analyse der verfügbaren Ansätze bzw. Systeme durchgeführt und ein Verfahren realisiert, das dem Problem der unnatürlichen Verzerrung mit perspektivischer Korrektheit begegnet. Nichtsdestotrotz resultiert das Verfahren aufgrund mangelnder Qualität der Bild- und Stereo-Daten auf Fahrzeugdatensätzen in einer gering aufgelösten Ansicht. Als Resultat wird ein hybrides Verfahren vorgeschlagen, in dem eine Oberflächenmodellierung der Umgebung Grundlage für die perspektivische Korrektheit schafft. Die Struktur der Modellierung lässt die Anwendung von Homographie-Transformationen zu, die prinzipiell auch in den gewöhnlichen Systemen zum Einsatz kommen. Im Unterschied werden die Homographien jedoch nur in lokal begrenzten Strukturen eingesetzt, in denen eine Konfidenz – im Gegensatz zu einer bloßen Annahme bei den klassischen Systemen – hinsichtlich der Korrektheit der Ebenenannahme gegeben ist.

Projektbedingt werden im vorgestellten Testdatensatz keine Rundum-Stereo-Aufnahmen verwendet. Für eine vollständige Ansicht aus der Vogelperspektive wäre eine vollständige Abdeckung des Fahrzeugumfelds mit Kameras erforderlich. Methodisch ist das vorgestellte Vorgehen jedoch äquivalent. Wünschenswert wäre in einer Weiterführung der Arbeit die Aufnahme bzw. der Aufbau eines rundum mit Stereokameras bestückten Fahrzeugs. Dabei könnte auch die Fragestellung der automatischen Farb- und Helligkeitsanpassungen thematisiert werden.

Die in dieser Dissertation vorgestellten Arbeiten beinhalten in den beiden Großthemen eine Erweiterung zum jeweiligen Stand der Technik. Die beiden Verfahren stellen Grundlagen für die Entwicklung erweiterter Fahrerassistenzsysteme und für Systeme im Bereich des semi-autonomen und autonomen Fahrens dar und adressieren insbesondere Nutzfahrzeuge.





## ABBILDUNGSVERZEICHNIS

---

Abb. 2.1	Koordinatensystem-Darstellung eines Kartesischen Koordinatensystems mit Orthonormalbasis . . . . .	10
Abb. 2.2	Relative Lage zwischen zwei Koordinatensystemen beschrieben durch Posen . . . . .	12
Abb. 3.1	Prinzip des Lochkameranmodells . . . . .	18
Abb. 3.2	Geometrische Verzerrungen: Tonnen- und Kissenverzerrung. . . . .	20
Abb. 3.3	Beispiele für AprilTags . . . . .	24
Abb. 4.1	Künstliches Muster für den Testaufbau . . . . .	30
Abb. 4.2	Mustergeometrie und Winkelrelationen nach [BZW06] . . . . .	31
Abb. 4.3	Winkelrelationen, Hilfspunkte und Koordinatensystem nach [BZW06] . . . . .	31
Abb. 4.4	Koordinatensysteme auf dem Zugfahrzeug . . . . .	35
Abb. 4.5	Maulkupplung an einem Zugfahrzeug . . . . .	36
Abb. 4.6	Koordinatensysteme im Zweiachsanhänger . . . . .	37
Abb. 4.7	Koordinatensysteme im zusammengekuppelten Fahrzeuggespann . . . . .	38
Abb. 4.8	Software-Architektur für das Messsystem . . . . .	40
Abb. 4.9	Die virtuelle Testumgebung als Inverse zur Statusschätzung . . . . .	42
Abb. 4.10	Laboraufbau im Maßstab 1 : 16 . . . . .	43
Abb. 4.11	Konfigurationsraum für eine vollständig detektierbare Markerpyramide . . . . .	45
Abb. 4.12	Absolute Fehler für die Gierwinkel (640 px × 480 px)	46
Abb. 4.13	Absolute Fehler für die Gierwinkel (1920 px × 1080 px) . . . . .	47
Abb. 4.14	Fehlerverhalten mit <i>KPF</i> . . . . .	48
Abb. 4.15	Modellbildung des Gespannstatus . . . . .	49
Abb. 4.16	Modellierung der Artikulation mit Kugelgelenken	51
Abb. 4.17	Projektion des künstlichen Musters . . . . .	52
Abb. 4.18	Fehler der Gierwinkelschätzung zu den Nick- und Rollzuständen („unebener“ Boden) (640 px × 480 px)	58
Abb. 4.19	Fehler der Nick- und Rollwinkelschätzung (640 px × 480 px) . . . . .	59
Abb. 4.20	Schematischer Aufbau des Kalman-Filters . . . . .	61
Abb. 4.21	Modellfunktionen zur kontinuierlichen Fahrzeugbewegung . . . . .	65
Abb. 4.22	Fehler $\Delta_E$ für die einzelnen Gespannstatuswinkel .	66
Abb. 4.23	Fehlermaße für das 3-D-Kalman-Verfahren . . . . .	67

Abb. 4.24	Anschauungsfall für die Evaluation der Trajektorienqualität . . . . .	68
Abb. 4.25	Datenfluss zwischen den Komponenten . . . . .	73
Abb. 4.26	Trajektorien bei Rauschlevel $\eta_l = 30$ px . . . . .	75
Abb. 4.27	Trajektorienqualität mittels <i>EMD</i> . . . . .	75
Abb. 4.28	Lenkwinkel für ein Rauschlevel von $\eta_l = 15$ px . . . . .	76
Abb. 4.29	Endpunktverteilung für ein Rauschlevel $\eta_l = 30$ px . . . . .	77
Abb. 5.1	Grundlagen der Epipolargeometrie . . . . .	80
Abb. 5.2	Ausrichtung der Kamerabilder durch Rektifizierung . . . . .	82
Abb. 5.3	Rektifizierungsprozess (Schema) . . . . .	83
Abb. 5.4	Sensorhierarchie in einem Fahrzeug (Beispiel) . . . . .	94
Abb. 5.5	Kommerzielles System für eine Vogelperspektivenansicht . . . . .	96
Abb. 5.6	Homographie-Verschattungseffekt . . . . .	97
Abb. 5.7	Homographie-Verschattungseffekt (Beispiel) . . . . .	97
Abb. 5.8	Perspektivische und Orthographische Projektion . . . . .	98
Abb. 5.9	Kamera-Frustum einer perspektivischen Kamera . . . . .	100
Abb. 5.10	Kamera-Frustum einer orthographischen Kamera . . . . .	100
Abb. 5.11	Koordinatensysteme für die Berechnung einer virtuellen Kamera . . . . .	102
Abb. 5.12	Schema zur punktbasierten Vogelperspektivenansicht . . . . .	103
Abb. 5.13	Uneindeutigkeit der Pixel-Rasterisierung . . . . .	104
Abb. 5.14	Testergebnisse für den Datensatz „playtable“ . . . . .	106
Abb. 5.15	Testergebnisse für den Datensatz „pipes“ . . . . .	107
Abb. 5.16	Testergebnisse für den Datensatz „recycle“ . . . . .	109
Abb. 5.17	Testergebnisse für den Datensatz „piano“ . . . . .	110
Abb. 5.18	Testergebnisse für den Datensatz „shelves“ . . . . .	111
Abb. 5.19	Ergebnis auf <i>Daimler Urban Segmentation Dataset</i> „test_2“, #656 . . . . .	113
Abb. 5.20	Virtuelle Vogelperspektive zu <i>Daimler Urban Segmentation Dataset</i> „test_2“, Bild #656 . . . . .	113
Abb. 5.21	Virtuelle Vogelperspektive zu <i>Daimler Urban Segmentation Dataset</i> „test_2“, Bild #394 . . . . .	114
Abb. 5.22	Virtuelle Vogelperspektive zu <i>Daimler Urban Segmentation Dataset</i> „train_1“, Bild #951 . . . . .	115
Abb. 5.23	Rasterkarte unter einem Fahrzeug . . . . .	117
Abb. 5.24	Anpassung einer lokalen gleitenden Rasterkarte bei veränderter Referenzposition . . . . .	119
Abb. 5.25	Koordinatensysteme von Rasterkarte und Fahrzeug . . . . .	120
Abb. 5.26	Abbilden der Punktwolken auf die Raster-Struktur . . . . .	121
Abb. 5.27	Geschlossene Oberflächenkarte . . . . .	124
Abb. 5.28	Partielle Homographie-Transformation anhand eines geschlossenen Oberflächenmodells . . . . .	126
Abb. 5.29	Schritte zur partiellen Homographie-Transformation (Schema) . . . . .	128
Abb. 5.30	Gewichtungsfunktion für die Höhenschätzung . . . . .	129

Abb. 5.31	Perspektivisch korrekte dichte Vogelperspektive, KITTI Odometry Dataset #16. . . . .	131
Abb. 5.32	Perspektivisch korrekte dichte Vogelperspektive, KITTI Odometry Dataset #16. . . . .	132
Abb. 5.33	Perspektivisch korrekte dichte Vogelperspektive, KITTI Odometry Dataset #10. . . . .	133
Abb. 5.34	Perspektivisch korrekte dichte Vogelperspektive, KITTI Odometry Dataset #3. . . . .	134
Abb. 5.35	Prinzip der iterativen Zellunterteilung . . . . .	138
Abb. 5.36	Höhenbestimmung für Subzellen . . . . .	138
Abb. 5.37	Beispiel für die iterative Zellverbesserung . . . . .	139
Abb. 5.38	Modellannahme des „Stixel World“-Verfahrens . . . . .	140
Abb. 5.39	Pixelweise Abtastung der Umgebung . . . . .	141
Abb. 5.40	Disparitäten-Korrespondenzen bei Horizontausrichtung . . . . .	142
Abb. 5.41	Punkteverteilung der Umgebungsabtastung mittels Stereokamera . . . . .	145
Abb. 5.42	Unterschiedliche Tiefenverteilung bei unterschiedlichen Kameraausrichtungen . . . . .	146
Abb. 5.43	Geometrische Zusammenhänge für die virtuelle Horizontausrichtung der Kamera . . . . .	148
Abb. 5.44	Sichtwinkelaufteilung im Vergleich zur Pixelabtastung . . . . .	149
Abb. 5.45	Abtastung des horizontalen Kameraöffnungswinkels	150
Abb. 5.46	Sichtebenenabtastung des Kameraöffnungswinkels	151
Abb. 5.47	Komponentenübersicht über die erweiterte Stereo-Hinderniserkennung . . . . .	153
Abb. 5.48	Systemkomponenten und Datenfluss für die perspektivisch korrekte, dichte Vogelperspektivenansicht . . . . .	153
Abb. 5.49	Perspektivisch korrekte, dichte Ansicht aus der Vogelperspektive. <i>KITTI Odometry Dataset #1</i> . . . . .	155
Abb. 5.50	Perspektivisch korrekte, dichte Ansicht aus der Vogelperspektive. <i>KITTI Odometry Dataset #5</i> . . . . .	156
Abb. 5.51	Perspektivisch korrekte, dichte Ansicht aus der Vogelperspektive. <i>KITTI Odometry Dataset #3</i> . . . . .	157
Abb. 5.52	Anordnung der Sensoren auf dem Testfahrzeug . . . . .	158
Abb. 5.53	Testfahrzeug und Testaufbau für die Aufnahme des Datensatzes . . . . .	160
Abb. 5.54	Vorbeifahrt an einer leeren Parklücke . . . . .	162
Abb. 5.55	Ausschnittsvergrößerungen der Bodentextur . . . . .	163
Abb. 5.56	Vergleichsansicht berechneter Ansicht und Drohnenaufnahme . . . . .	164
Abb. 5.57	3-D-Ansicht des geschlossenen Oberflächenmodells	165
Abb. 5.58	Vogelperspektive eines Feldwegs . . . . .	166

Abb. 5.59	Szenenvisualisierungen und geschlossenes Oberflächenmodell . . . . .	167
Abb. 5.60	Vogelperspektivenansicht auf einem Feldweg. Unten links: Kameraansicht (früherer Zeitschritt mit sichtbarem Strohhallen). . . . .	167
Abb. 5.61	Vergleichsansicht berechneter Ansichten und Drohnenaufnahme (Feldweg) . . . . .	168
Abb. 5.62	Vergleichsansicht zwischen dem punktbasierten Verfahren und dem Oberflächenmodell-Verfahren .	169

## TABELLENVERZEICHNIS

---

Tab. 4.1	Modellparameter für die Evaluation mittels Simulationskomponente . . . . .	42
Tab. 4.2	Evaluationsergebnisse für das 2- <i>D</i> -Verfahren . . . . .	44
Tab. 4.3	Evaluationsergebnisse mit <i>KPF</i> (2- <i>D</i> ) . . . . .	48
Tab. 4.4	Evaluationsergebnisse für das 3- <i>D</i> -Verfahren . . . . .	57
Tab. 5.1	Stereoverbünde für die Testaufnahmen . . . . .	159
Tab. 5.2	Verwendung der einzelnen Stereokameras . . . . .	161

## ALGORITHMENVERZEICHNIS

---

Alg. 5.1	Höhenschätzung für eine Zelle . . . . .	123
Alg. 5.2	Rekursive Zellenerweiterung . . . . .	137



## VERZEICHNIS MATHEMATISCHER SYMBOLE

---

### FORMELZEICHEN UND FUNKTIONEN

<b>Zeichen</b>	<b>Bedeutung</b>
$\mathbf{0}$	Nullvektor (Spaltenvektor)
$\mathbf{a}$	Hilfsvariable
$\mathbf{A}$	Matrix (Hilfsvariable)
$A, B$	Mengen
$\mathbf{a}, \mathbf{b}$	Vektoren (Hilfsvariablen)
$\mathbf{a}, \mathbf{b}, \mathbf{f}$	Posen (mit unterschiedlicher Semantik, Hilfsvariablen)
$\alpha, \beta, \gamma, \delta$	Winkel (Hilfsvariable)
angleaxis ( $\beta, \mathbf{a}$ )	Rotationsmatrix um den Winkel $\beta$ um den Vektor $\mathbf{a}$
$\mathbf{C}$	Position der Kamera
$\mathbf{c}$	Zelle (einer Rasterkarte)
$\mathbf{c}$	Kamera-Pose
$c_x, c_y$	Kamerahauptpunkt
$\mathbf{C}$	Kamera
$\chi$	Gewichtungsfunktion
$D$	Menge von Markereckpunkten
$d$	Distanz
$\mathfrak{d}$	Transformationspose zwischen Drehschmel und Kupplungsauge des Anhängers
$d_{\Delta}$	Schenkellänge der Markerpyramide
$\mathbf{d}$	Eckpunkt eines künstlichen Markers
$\Delta_i$	Differenz/Delta in Referenz zu $i$
$\Delta_t$	Zeitdifferenz zwischen zwei Schritten/Durchläufen
$E$	(Homographie-)Ebene
$\mathbf{e}$	Epipol
$e$	Dimension einer lokalen Rasterkarte
$\mathcal{E}$	Essentielle Matrix
$\mathbf{e}$	Geschätzte Pose des künstlichen Musters relativ zur Kamera

$\epsilon$	Rauschvektor (Kalman)
$\eta_l$	Rauschlevel
$\eta$	Rauschvektor
$F$	Brennweite (einer Kamera)
$\mathcal{F}$	Fundamental-Matrix
$f$	Vorhersage-Funktion (Kalman-Filter)
$F$	Vorhersage-Matrix (Kalman-Filter)
$G$	Rasterkarte
$g$	Geometriekonfiguration bzw.-schätzung
$g$	Seitenlänge einer Zelle in einer Rasterkarte
$\mathfrak{g}$	Artikulationspose am Drehschemel des Anhängers
$\Gamma_C$	Projektionsfunktion einer Menge von Punkten für Kamera C
$\gamma_C$	Projektionsfunktion eines einzelnen Punktes für Kamera C
$\text{ground}(c)$	Prädikat das angibt, ob $c$ <i>Boden</i> ist
$H$	Homographie-Matrix nach [HZ03]
$h$	Bildhöhe in Pixeln
$\mathcal{H}$	Histogramm
$H$	Messmatrix
$\mathfrak{h}$	Horizontpose (Delta-Pose)
$h_\Delta$	Tiefe der Markerpyramide
$\mathfrak{h}$	Messmodell
$\mathbb{H}$	Menge der Hamilton'schen Quaternionen (Hamilton-Raum)
$\text{hom}_H(q)$	Transformation mittels Homographie $H$ für Punkt $q$
$\mathcal{J}$	Intrinsikkonfiguration einer Kamera
$I_V$	Transformationsmatrix in Pixelkoordinaten einer virtuellen Kamera
$i, j$	Indizes
$\text{insubset}(i, p)$	Prädikat das angibt, ob $p$ im Sub-Raum $i$ liegt
$\iota$	Kalman-Innovation
$J_{i,j}$	Punktewolke einer Zelle $c_{i,j}$
$K$	Intrinsische Kameramatrix
$k_x, k_y$	Kamerakonstanten
$\mathcal{K}$	Kalman-Matrix



$\mathfrak{k}$	Artikulationspose an der Anhängerkupplung
$\mathbf{k}$	Kupplungspunkt
$\kappa$	Verzerrungskoeffizienten einer Kamera
$\varphi$	Funktion für die Artikulationssimulation
$l$	Länge (Distanz)
$L_{i,j}$	Subzelle $i,j$
$\lambda$	Disparität
$\mathbf{M}$	Markerpunkt
$\mathbf{m}$	Relativpose des künstlichen Musters am Anhänger
$\mu$	Mittelwert
$n, m$	Anzahlen ( $\in \mathbb{N}$ )
$\mathbf{n}$	Normalenvektor
$\mathcal{N}_{\mu,\sigma}$	Normal- bzw. Gaußverteilung
$\nu$	Nachbarschaftsrelation
$\mathbf{o}$	Stixel
$O_{i,j}$	Normalisierte Punktwolke einer Zelle $\mathbf{c}_{i,j}$
$\mathbf{0}^a$	Ursprung eines Koordinatensystems $a$
$\text{obstacle}(\mathbf{c})$	Prädikat das angibt, ob $\mathbf{c}$ <i>Hindernis</i> ist
$\Omega(\mathbf{c})$	Ursprung einer Zelle $\mathbf{c}$
$\omega$	Winkelgeschwindigkeit im Kalman-Bewegungsmodell
$\mathbf{o}$	Null-Pose (Identität)
$P$	Menge von Punkten im 3- <i>D</i> Raum
$\mathbf{P}(\text{conf})$	Funktion zur Ermittlung einer Projektionsmatrix anhand einer Konfiguration $\text{conf}$
$\mathbf{p}$	Pose der Anhängerkupplung/des Kupplungspunkts
$\mathbf{p}$	Punkt im 3- <i>D</i> Raum
$\phi$	Rollwinkel
$\Phi$	Abbildung einer 2- <i>D</i> -Rasterkartenposition auf die korrespondierende Zell-Koordinate
$\Pi$	Menge der <i>Hindernis</i> -Zellen
$\omega$	Höhen des geschlossenen Oberflächenmodells
$\mathcal{P}$	Abbildungsfunktion (einer Kamera)
$\psi$	Lenkwinkel
$Q$	Menge von Punkten im 2- <i>D</i> Raum
$\mathbf{q}$	Punkt im 2- <i>D</i> Raum

$\mathcal{Q}$	Rotationsfunktion
$Q$	Rückprojektionsmatrix für Disparitätenkarten
$r$	Texturauflösung
$R$	Rotationsmatrix
$r$	Rotationsquaternion
$\text{rot}(\mathbf{r})$	Liefert die Rotationsmatrix zu Einheitsquaternion $\mathbf{r}$
$\rho$	Fahrzeugorientierung (auf der Ebene)
$\rho_G$	Projektionsfunktion eines Punktes auf die 2-D-Rasterkarte $G$
$s$	Skalar (Hilfsvariable)
$S$	Stereoverbund
$S$	Menge der Subzellen mit gemeinsamer Mutterzelle
$\mathbf{s}$	Sensor-Pose
$s$	Gespannzustand hinsichtlich der Artikulation
$\mathfrak{z}(\zeta_\tau, \mathbf{u})$	Transitfunktion zum Zeitschritt $\tau + 1$ anhand des Steuervektors $\mathbf{u}$ und des aktuellen Zustands $\zeta_\tau$
$\sigma$	Standardabweichung
$SO(3)$	Menge der Einheitsquaternione, $SO(3) \subset \mathbb{H}$
$\zeta$	Gespann-Konfigurationsvektor im Fahrmanöver
$T_{a \rightarrow b}$	Transformation von Koordinatensystem $a$ in das Koordinatensystem $b$
$t$	Zeit (meist Parametervariable)
$T(\mathbf{a})$	Transformationsmatrix zu Pose $\mathbf{a}$
$\mathcal{T}$	Texturschicht
$\mathbf{t}$	Lage des Drehschemels relativ zum Ursprung des Anhängerkoordinatensystems
$t$	Texturkoordinate
$\tau$	Zeitschritt (Zählvariable)
$\theta$	Nickwinkel
$\vartheta$	Schwellwert
$\mathbf{U}$	Prozess-Rausch-Kovarianzmatrix
$u, v$	Raster-Indizierung ( $\in \mathbb{Z}$ )
$\mathbf{u}$	Steuervektor bzw. -kommando
$\mathbf{up}$	Up-Vektor
$\Upsilon_V$	Funktion zur Transformation einer Punktmenge in das Koordinatensystem von $V$

$v$	Geschwindigkeit (Skalar)
$\mathbf{v}$	Fahrzeugpose
$\mathbf{v}$	Geschwindigkeitsvektor im Kalman-Bewegungsmodell
$V$	Virtuelle Kamera
$w$	Bildbreite in Pixeln
$\mathbf{W}$	Kovarianzmatrix
$\mathbf{w}$	Pose der virtuellen Kamera
$w_{\Delta}$	Halbe Breite der Markerpyramide
$w_m$	Seitenlänge eines Markers
$\mathbf{x}$	System-Status des Kalman-Filters
$\xi$	Gierwinkel
$\mathbf{y}(u,v)$	Oberflächenkoordinate mit Referenz zum Ursprung der Zelle $(u,v)$
$\mathbf{Z}$	Matrix der aktuellen Messwerte
$z$	Messung (Kalman-Filter)
$\zeta$	Messrauschen

## OPERATOREN

<b>Zeichen</b>	<b>Bedeutung</b>
$\ominus$	Subtraktionsoperator für Posen
$\oplus$	Additionsoperator für Posen

## KOORDINATENSYSTEME

<b>Zeichen</b>	<b>Bedeutung</b>
a, b	Generische Koordinatensysteme
c	Kamera-Koordinatensystem
d	Koordinatensystem des Drehschemels
h	Horizont-Koordinatensystem
i	Bild-Koordinatensystem
j	Bild-Koordinatensystem (für Referenzpunkte)
k	Koordinatensystem der Anhängerkupplung
m	Koordinatensystem des künstlichen Musters
p	Pixel-Koordinatensystem im digitalen Bild
s	Sensorkoordinatensystem
t	Koordinatensystem des Anhängers
v	Koordinatensystem des (Zug-)Fahrzeugs
w	Welt-Koordinatensystem
z	Zell-Koordinatensystem

## LITERATURVERZEICHNIS

---

- [AG95] H. Alt und M. Godau. Computing the Fréchet distance between two polygonal curves. *Journal of Computational Geometry & Applications*, 5:75–91, 1995. DOI: 10.1142/S0218195995000064.
- [AJNo5] A. Agarwal, C. V. Jawahar und P. J. Narayanan. A survey of planar homography estimation techniques. In *Tech. Rep. IIT/TR/2005/12*. Technischer Bericht 3, Centre for Visual Information Technology, 2005. DOI: 10.1002/chin.200444283.
- [Alto1] C. Altafini. Some properties of the general n-trailer. *International Journal of Control*, 74(4):409–424, 2001. DOI: 10.1080/00207170010010579.
- [Alto2] C. Altafini. Following a path of varying curvature as an output regulation problem. *IEEE Transactions on Automatic Control*, 47(9):1551–1556, 2002. DOI: 10.1109/TAC.2002.802750.
- [ASJo2] C. Altafini, A. Speranzon und K. H. Johansson. Hybrid control of a truck and trailer vehicle. In C. J. Tomlin und M. R. Greenstreet (Editoren), *Hybrid Systems: Computation and Control*, Seiten 21–34. Springer, Berlin, Heidelberg, 2002. DOI: 10.1007/3-540-45873-5\_5. ISBN: 978-3-540-43321-7.
- [ASWo1] C. Altafini, A. Speranzon und B. Wahlberg. A feedback control scheme for reversing a truck and trailer vehicle. *IEEE Transactions on Robotics and Automation*, 17(6):915–922, 2001. DOI: 10.1109/70.976025.
- [BB98] M. Bertozzi und A. Broggi. GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection. *IEEE Transactions on Image Processing*, 7(1):62–81, 1998. DOI: 10.1109/83.650851.
- [BBC<sup>+</sup>14] N. Bernini, M. Bertozzi, L. Castangia, M. Patander und M. Sabbatelli. Real-time obstacle detection using stereo vision for autonomous ground vehicles: A survey. In *IEEE Conference on Intelligent Transportation Systems (ITSC)*, Seiten 873–878. IEEE, 2014. DOI: 10.1109/ITSC.2014.6957799. ISSN: 2153-0009.

- [BFP09] H. Badino, U. Franke und D. Pfeiffer. The Stixel World - A Compact Medium Level Representation of the 3D-World. In J. Denzler, G. Notni und H. Süße (Editoren), *Pattern Recognition, DAGM 2009. LNCS Vol. 5748*, Seiten 51–60, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg. DOI: 10.1007/978-3-642-03798-6\_6. ISBN: 978-3-642-03798-6.
- [BFPF11] F. Boem, F. A. Pellegrino, G. Fenu und T. Parisini. Trajectory clustering by means of Earth Mover's Distance. *2011 International Federation of Automatic Control, 18th IFAC World Congress*, 44(1):4741–4746, 2011. DOI: 10.3182/20110828-6-IT-1002.01230.
- [BR16] M. Burkhart und T. Rösch. ZF Friedrichshafen AG. Vorrichtung und Verfahren zur Erfassung eines Schwenkwinkels zwischen einem Fahrzeug und einer Anhängervorrichtung, Patent DE10 2014 212 821 A1. Deutsches Patent- und Markenamt, München, 2016.
- [Bro71] D. C. Brown. Close-range camera calibration. *Photogrammetric Engineering*, 37(8):855–866, 1971.
- [BTGo6] H. Bay, T. Tuytelaars und L. V. Gool. SURF : Speeded Up Robust Features. In A. Leonardis, H. Bischof und A. Pinz (Editoren), *European Conference on Computer Vision. LNCS Vol. 3951.*, Seiten 404–417, Heidelberg, Berlin, 2006. Springer. DOI: 10.1007/11744023\_32.
- [Buc10] S. J. Buckley. FCA US LLC. Trailer detection system, U.S. Patent 7786849. Patent and Trademark Office, Washington, DC, 2010.
- [BZo6] U. Berg und D. Zöbel. Visual Steering Assistance for Backing-Up Vehicles with One-axle Trailer. In A. Gale (Editor), *Vision in Vehicles 11*, Dublin, 2006.
- [BZW06] E. Balcerak, D. Zöbel und T. Weidenfeller. Universität Koblenz-Landau. Verfahren zum Bestimmen einer Position, Vorrichtung und Computerprogrammprodukt, Patent DE 10 2006 056 408 A1. Deutsches Patent- und Markenamt, München, 2006.
- [CC18] J.-R. Chang und Y.-S. Chen. Pyramid stereo matching network. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seiten 5410–5418. IEEE, 2018. DOI: 10.1109/CVPR.2018.00567.
- [CFS<sup>+</sup>10] L. Chu, Y. Fang, M. Shang, J. Guo und F. Zhou. Estimation of Articulation Angle for Tractor Semi-trailer

- Based on State Observer. In *2010 International Conference on Measuring Technology and Mechatronics Automation*, Band 2, Seiten 158–163. IEEE, 2010. DOI: 10.1109/ICMTMA.2010.342. ISSN: 2157-1473.
- [CPY<sup>+</sup>11] W. Chung, M. Park, K. Yoo, J. I. Roh und J. Choi. Backward-motion control of a mobile robot with n passive off-hooked trailers. *Journal of mechanical science and technology*, 25(11):2895–2905, 2011. DOI: 10.1007/s12206-011-0909-7.
- [CSMH13] L. Caup, J. Salmen, I. Muharemovic und S. Houben. Video-based Trailer Detection and Articulation Estimation. In *IEEE Intelligent Vehicles Symposium (IV)*, Seiten 1179–1184. IEEE, 2013. DOI: 10.1109/IVS.2013.6629626. ISBN: 9781467327541.
- [DPW13] D. G. Dansereau, O. Pizarro und S. B. Williams. Decoding, calibration and rectification for lenselet-based plenoptic cameras. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Seiten 1027–1034. IEEE, 2013. DOI: 10.1109/CVPR.2013.137. ISBN: 978-0-7695-4989-7. ISSN: 10636919.
- [dSC19] C. de Saxe und D. Cebon. Measurement of articulation angle by image template matching. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 0(0):0954407019833819, 2019. DOI: 10.1177/0954407019833819.
- [DW97] A. W. Divelbiss und J. T. Wen. A path space approach to nonholonomic motion planning in the presence of obstacles. *IEEE Transactions on Robotics and Automation*, 13(3):443–451, 1997. DOI: 10.1109/70.585905.
- [EE14] N. Einecke und J. Eggert. Block-matching stereo with relaxed fronto-parallel assumption. In *IEEE Intelligent Vehicles Symposium (IV)*, Seiten 700–705. IEEE, 2014. DOI: 10.1109/IVS.2014.6856414. ISBN: 978-1-4799-3638-0.
- [EM94] T. Eiter und H. Mannila. Computing discrete Fréchet distance. In *CD-TR 94/64*. Technischer Bericht, TU Wien, Wien, 1994.
- [Fia05] M. Fiala. ARTag, a fiducial marker system using digital techniques. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Band 2, Seiten 590–596. IEEE, IEEE, 2005. DOI: 10.1109/CVPR.2005.74.

- [GD01] C. Geyer und K. Daniilidis. Catadioptric projective geometry. *International Journal of Computer Vision*, 45(3):223–243, 2001. DOI: 10.1023/A:1013610201135. ISSN: 09205691.
- [GHJV94] E. Gamma, R. Helm, R. Johnson und J. Vlissides. *Design Patterns – Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional, Boston, 1. Auflage, 1994. ISBN: 978-0201633610.
- [GJK<sup>+</sup>10] A. Gaedke, T. Joos, T. Kropf, M. Noll und D. Schupp. Robert Bosch GmbH. Erfassung und Auswertung einer Lagebeziehung zwischen einem Kraftfahrzeug und einem Anhänger, Patent DE 10 2010 008 324 A1. Deutsches Patent- und Markenamt, München, 2010.
- [GLU12] A. Geiger, P. Lenz und R. Urtasun. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Seiten 3354–3361. IEEE, 2012. DOI: 10.1109/CVPR.2012.6248074. ISBN: 978-1-4673-1228-8. ISSN: 10636919.
- [Gri85] W. E. Grimson. Computational experiments with a feature based stereo algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(1):17–34, 1985. DOI: 10.1109/TPAMI.1985.4767615. ISSN: 0162-8828.
- [GS94] Guo-Qing Wei und Song De Ma. Implicit and explicit camera calibration: theory and experiments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5):469–480, 1994. DOI: 10.1109/34.291450. ISSN: 0162-8828.
- [Ham50] R. W. Hamming. Error Detecting and Error Correcting Codes. *Bell System Technical Journal*, 29(2):147–160, 1950. DOI: 10.1002/j.1538-7305.1950.tb00463.x. ISBN: 0005-8580. ISSN: 15387305.
- [Har97] R. I. Hartley. In defense of the eight-point algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(6):580–593, 1997. DOI: 10.1109/34.601246. ISBN: 0162-8828. ISSN: 01628828.
- [Hiro8] H. Hirschmüller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):328–341, 2008. DOI: 10.1109/TPAMI.2007.1166. ISBN: 0769523722. ISSN: 01628828.



- [Hit41] F. L. Hitchcock. The Distribution of a Product from Several Sources to Numerous Localities. *Journal of Mathematics and Physics*, 1941. DOI: 10.1002/sapm1941201224. ISBN: 9788578110796. ISSN: 00971421.
- [HK13] S. Hermann und R. Klette. Iterative semi-global matching for robust driver assistance systems. In K. M. Lee, Y. Matsushita, J. M. Rehg und Z. Hu (Editoren), *Asian Conference on Computer Vision, LNCS Vol. 7726*, Seiten 465–478, Berlin, Heidelberg, 2013. Springer. DOI: 10.1007/978-3-642-37431-9\_36. ISBN: 9783642374302. ISSN: 03029743.
- [HS89] R. Horaud und T. Skordas. Stereo correspondence through feature grouping and maximal cliques. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(11):1168–1180, 1989. DOI: 10.1109/34.42855. ISSN: 01628828.
- [HS97] J. Heikkilä und O. Silvén. A Four-step Camera Calibration Procedure with Implicit Image Correction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Seiten 1106–1112. IEEE, 1997. DOI: 10.1109/CVPR.1997.609468.
- [HS07] H. Hirschmüller und D. Scharstein. Evaluation of Cost Functions for Stereo Matching. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Seiten 1–8. IEEE, 2007. DOI: 10.1109/CVPR.2007.383248. ISBN: 1-4244-1179-3. ISSN: 1063-6919.
- [HWFS13] C. Hertzberg, R. Wagner, U. Frese und L. Schröder. Integrating generic sensor fusion algorithms with sound state representations through encapsulation of manifolds. *Information Fusion*, 14(1):57–77, 2013. DOI: 10.1016/j.inffus.2011.08.003.
- [HZ03] R. Hartley und A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York City, 2. Auflage, 2003. ISBN: 0-521-54051-8.
- [JU04] S. Julier und J. Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3):401–422, 2004. DOI: 10.1109/JPROC.2003.823141. ISBN: 9780470747049. ISSN: 0018-9219.
- [Kae15] M. Kaess. Simultaneous localization and mapping with infinite planes. In *IEEE International Conference on*

- Robotics and Automation (ICRA)*, Band 2015-June, Seiten 4605–4611. IEEE, 2015. DOI: 10.1109/ICRA.2015.7139837. ISSN: 1050-4729.
- [Kal60] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 1960. DOI: 10.1115/1.3662552.
- [Kan42] L. V. Kantorovich. On the translocation of masses. In *Dokl. Akad. Nauk SSSR*, Band 37, Seiten 227–229, 1942. DOI: 10.1007/s10958-006-0049-2.
- [KB99] H. Kato und M. Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *2nd IEEE and ACM International Workshop on Augmented Reality (IWAR'99)*, Seiten 85–94. IEEE, 1999. DOI: 10.1109/IWAR.1999.803809.
- [KB16] A. Kaehler und G. Bradski. *Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library*. O'Reilly UK Ltd., Farnham, 1. Auflage, 2016. ISBN: 978-1491937990.
- [KK92] S.-G. Kong und B. Kosko. Adaptive fuzzy systems for backing up a truck-and-trailer. *Neural Networks, IEEE Transactions on*, 3(2):211–223, 1992. DOI: 10.1109/72.125862.
- [KMD<sup>+</sup>17] A. Kendall, H. Martirosyan, S. Dasgupta, P. Henry, R. Kennedy, A. Bachrach und A. Bry. End-To-End Learning of Geometry and Context for Deep Stereo Regression. In *IEEE International Conference on Computer Vision (ICCV)*, Seiten 65–75. IEEE, 2017. DOI: 10.1109/ICCV.2017.17.
- [LA03] R. Labayrade und D. Aubert. A single framework for vehicle roll, pitch, yaw estimation and obstacles detection by stereovision. In *IEEE Intelligent Vehicles Symposium (IV)*, Seiten 31–36. IEEE, 2003. DOI: 10.1109/IVS.2003.1212878.
- [LAT02] R. Labayrade, D. Aubert und J.-P. Tarel. Real time obstacle detection in stereovision on non flat road geometry through "v-disparity"representation. In *IEEE Intelligent Vehicle Symposium (IV)*, Band 2, Seiten 646–651. IEEE, 2002. DOI: 10.1109/IVS.2002.1188024. ISBN: 0-7803-7346-4. ISSN: 19449224.
- [Lau90] J.-P. Laumond. Nonholonomic motion planning versus controllability via the multibody car system example. Technischer Bericht, Stanford University, Stanford, 1990.

- [LCKSo4] J.-H. Lee, W. Chung, M. Kim und J.-B. Song. A passive multiple trailer system with off-axle hitching. *International Journal of Control, Automation and Systems*, 2(3):289–297, 2004.
- [LCS11] S. Leutenegger, M. Chli und R. Y. Siegwart. BRISK: Binary robust invariant scalable keypoints. In *2011 IEEE International Conference on Computer Vision (ICCV)*, Seiten 2548–2555. IEEE, 2011. DOI: 10.1109/ICCV.2011.6126542. ISBN: 145771101X.
- [LCS19] P. Li, X. Chen und S. Shen. Stereo r-cnn based 3d object detection for autonomous driving. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Seiten 7644–7652. IEEE, 2019.
- [Lev44] K. Levenberg. A Method for the Solution of Certain Problems in Least Squares. In *Quarterly of Applied Mathematics*, Band 2, Seiten 164–168. American Mathematical Society, 1944. DOI: 10.1090/qam/10666. ISSN: 1552-4485.
- [LF94] S. Laveau und O. Faugeras. 3-D scene representation as a collection of images. In *International Conference on Pattern Recognition*, Band 1, Seiten 689–691. IEEE, 1994. DOI: 10.1109/ICPR.1994.576404. ISBN: 0-8186-6265-4.
- [LGF15] D. Levi, N. Garnett und E. Fetaya. StixelNet: A Deep Convolutional Network for Obstacle Detection and Road Segmentation. In Xianghua Xie Mark W. Jones und G. K. L. Tam (Editoren), *British Machine Vision Conference (BMVC)*, Seiten 109.1–109.12. BMVA Press, sep 2015. DOI: 10.5244/C.29.109. ISBN: 1-901725-53-7.
- [LGJH12] Y. Lu, J. S. Gibson, T. G. Johnson und S. T. Hunt. Magna Electronics Inc. Rear vision system with trailer angle detection, Patent WO 2012 103 193 A1, PCT/US2012/022,517. World Intellectual Property Organization, 2012.
- [LH81] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293(5828):133–135, 1981. DOI: 10.1038/293133a0. ISBN: 8880551515. ISSN: 0028-0836.
- [LH11] S. Li und Y. Hai. Easy calibration of a blind-spot-free fisheye camera system using a scene of a parking space. *IEEE Transactions on Intelligent Transportation Systems*, 12(1):232–242, 2011. DOI: 10.1109/TITS.2010.2085435. ISSN: 15249050.

- [LK11] Y. H. Lee und A. Kade. GM Global Technology Operations LLC. Trailer articulation angle estimation, U.S. Patent 8 073 594. Patent and Trademark Office, Washington, DC, 2011.
- [LKP<sup>+</sup>09] L. Luo, I. Koh, S. Park, R. Ahn und J. Chong. A software-hardware cooperative implementation of bird's-eye view system for camera-on-vehicle. In *IEEE Conference on Network Infrastructure and Digital Content*, Seiten 963–967. IEEE, 2009. DOI: 10.1109/ICNIDC.2009.5360920. ISBN: 978-1-4244-4898-2.
- [LLCo8] Y. C. Liu, K. Y. Lin und Y. S. Chen. Bird's-eye view vision system for vehicle surrounding monitoring. In G. Sommer und R. Klette (Editoren), *RobVis 2008: Robot Vision*, Band 4931 LNCS, Seiten 207–218, Berlin, Heidelberg, 2008. Springer. DOI: 10.1007/978-3-540-78157-8\_16. ISBN: 978-3-540-78156-1.
- [Low04] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. DOI: 10.1023/B:VISI.0000029664.99615.94. ISSN: 0920-5691.
- [LT88] R. K. Lenz und R. Y. Tsai. Techniques for calibration of the scale factor and image center for high accuracy 3-D machine vision metrology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(5):713–720, 1988. DOI: 10.1109/34.6781. ISSN: 0162-8828.
- [LZ99] C. Loop und Z. Zhang. Computing rectifying homographies for stereo vision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Band 1, Seiten 125–131. IEEE, 1999. DOI: 10.1109/CVPR.1999.786928. ISSN: 1063-6919.
- [Mar63] D. W. Marquardt. An Algorithm for Least-Squares Estimation of Nonlinear Parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2):431–441, 1963. DOI: 10.1137/0111030. ISBN: 0368-4245. ISSN: 0368-4245.
- [Mic12] M. Michalek. Tracking Control Strategy for the Standard N-trailer Mobile Robot—A Geometrically Motivated Approach. In K. Kozłowski (Editor), *Robot Motion and Control 2011, LNCIS Vol. 422*, Seiten 39–51. Springer, London, 2012.
- [MIH<sup>+</sup>16] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy und T. Brox. A Large Dataset to Train

- Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Seiten 4040–4048. IEEE, 2016.
- [MM96] M. C. M. C. Martin und H. P. Moravec. Robot Evidence Grid. Technischer Bericht März, Carnegie Mellon University, Pittsburgh, 1996.
- [Mon81] G. Monge. Mémoire sur la théorie des déblais et remblais, *Mémoires Acad. Royale Sci. Paris*, 3:666–704, 1781.
- [NETCo1] Y. Nakamura, H. Ezaki, Y. Tan und W. Chung. Design of steering mechanism and control of nonholonomic trailer systems. *IEEE Transactions on Robotics and Automation*, 17(3):367–374, 2001. DOI: 10.1109/70.938393.
- [NWW<sup>+</sup>16] F. Neuhaus, N. Wojke, C. Winkens, B. Kraye, D. Paulus und M. Häselich. Autonomous 3d Terrain Mapping and Object Localization for the Spacebot Camp 2015. In *International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS)*, Seiten 2b1.1–2b1.8. ESA, 2016.
- [OGABVV02] R. Orosco-Guerrero, E. Aranda-Bricaire und M. Velasco-Villa. Global path-tracking for a multi-steered general n-trailer. In *IFAC World Congress*, Band 35, Seiten 477–482. Elsevir, 2002. DOI: 10.3182/20020721-6-ES-1901.00241.
- [Ols11] E. Olson. AprilTag: A robust and flexible visual fiducial system. In *IEEE International Conference on Robotics and Automation (ICRA)*, Seiten 3400–3407. IEEE, 2011. DOI: 10.1109/ICRA.2011.5979561.
- [ON10] F. Oniga und S. Nedeveschi. Processing Dense Stereo Data Using Elevation Maps: Road Surface, Traffic Isle, and Obstacle Detection. *IEEE Transactions on Vehicular Technology*, 59(3):1172–1182, 2010. DOI: 10.1109/TVT.2009.2039718. ISSN: 0018-9545.
- [PN12] C. D. Pantilie und S. Nedeveschi. SORT-SGM: Subpixel Optimized Real-Time Semiglobal Matching for Intelligent Vehicles. *IEEE Transactions on Vehicular Technology*, 61(3):1032–1042, 2012. DOI: 10.1109/TVT.2012.2186836. ISSN: 0018-9545.
- [Raj12] R. Rajamani. *Vehicle Dynamics and Control*. Springer, New York City, 2. Auflage, 2012. DOI: 10.1007/978-1-4614-1433-9. ISBN: 978-1-4614-1432-2.

- [RRKB11] E. Rublee, V. Rabaud, K. Konolige und G. Bradski. ORB - an efficient alternative to SIFT or SURF. In *IEEE International Conference on Computer Vision (ICCV)*, Seiten 2564–2571. IEEE, 2011. DOI: 10.1109/ICCV.2011.6126544. ISBN: 978-1-4577-1102-2.
- [RTGoo] Y. Rubner, C. Tomasi und L. J. Guibas. The Earth Mover’s Distance as a Metric for Image Retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000. DOI: 10.1023/A:1026543900054. ISSN: 0920-5691.
- [SBZ09] J. Schikora, U. Berg und D. Zöbel. Berührungslose Winkelbestimmung zwischen Zugfahrzeug und Anhänger. In W. A. Halang und P. Hollecsek (Editoren), *Aktuelle Anwendungen in Technik und Wirtschaft*, Seiten 11–20, Berlin, Heidelberg, 2009. Springer. DOI: 10.1007/978-3-540-85324-4\_2. ISBN: 978-3-540-85324-4.
- [Scao7] D. Scaramuzza. *Omnidirectional vision: from calibration to robot motion estimation*. Dissertation, ETH Zürich, 2007.
- [SEFR13] T. Scharwächter, M.ENZweiler, U. Franke und S. Roth. Efficient Multi-Cue Scene Segmentation. In J. Weickert, M. Hein und B. Schiele (Editoren), *German Conference on Pattern Recognition (GCPR), LNCS Vol. 8142*, Seiten 435–445, Berlin, Heidelberg, 2013. Springer. DOI: 10.1007/978-3-642-40602-7\_46. ISBN: 978-3-642-40601-0.
- [SERF14] T. Scharwächter, M.ENZweiler, S. Roth und U. Franke. Stixmantics: a medium-level model for real-time semantic scene understanding. In D. Fleet, T. Pajdla, B. Schiele und T. Tuytelaars (Editoren), *European Conference on Computer Vision (ECCV)*, Seiten 533–548. Springer, Cham, 2014. DOI: 10.1007/978-3-319-10602-1\_35. ISBN: 978-3-319-10601-4.
- [SFH<sup>+</sup>02] D. Schmalstieg, A. Fuhrmann, G. Hesina, Z. Szalavári, L. M. Encarnação, M. Gervautz und W. Purgathofer. The studierstube augmented reality project. *Presence: Teleoperators and Virtual Environments*, 11(1):33–54, 2002. DOI: 10.1162/105474602317343640.
- [SHK<sup>+</sup>14] D. Scharstein, H. Hirschmüller, Y. Kitajima, G. Krauthwohl, N. Nešić, X. Wang und P. Westling. High-Resolution Stereo Datasets with Subpixel-Accurate Ground Truth. In X. Jiang, J. Hornegger und R. Koch (Editoren), *German Conference on Pattern Recognition*,

- Band 1, Seiten 31–42. Springer, Berlin, Heidelberg, 2014. ISBN: 978-3-319-11751-5.
- [SK00] H.-Y. Shum und S. B. Kang. A review of image-based rendering techniques. In *SPIE Visual Communications and Image Processing (Vol. 4067)*, Seiten 2–13. SPIE, 2000. DOI: 10.1117/12.386541. ISBN: 0-8194-3703-4. ISSN: 0277786X.
- [SMS<sup>+</sup>13] T. Sato, A. Moro, A. Sugahara, T. Tasaki, A. Yamashita und H. Asama. Spatio-temporal bird’s-eye view images using multiple fish-eye cameras. In *IEEE/SICE International Symposium on System Integration*, Seiten 753–758. IEEE, 2013. DOI: 10.1109/SII.2013.6776674. ISBN: 978-1-4799-2625-1.
- [SP07] D. Scharstein und C. Pal. Learning conditional random fields for stereo. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Seiten 1–8. IEEE, 2007. DOI: 10.1109/CVPR.2007.383191. ISSN: 1063-6919.
- [SS02] D. Scharstein und R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1-3):7–42, 2002. DOI: 10.1023/A:1014573219977.
- [SS03] D. Scharstein und R. Szeliski. High-Accuracy Stereo Depth Maps Using Structured Light. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Band 1, Seiten I–195–I–202. IEEE, 2003. DOI: 10.1109/CVPR.2003.1211354. ISBN: 0-7695-1900-8. ISSN: 1063-6919.
- [TCPC11] B. Thomas, R. Chithambaran, Y. Picard und C. Cougnard. Development of a cost effective bird’s eye view parking assistance system. In *2011 IEEE Recent Advances in Intelligent Computational Systems*, Seiten 461–466. IEEE, 2011. DOI: 10.1109/RAICS.2011.6069355. ISBN: 978-1-4244-9478-1.
- [Ten16] G. Tenckhoff. ZF Friedrichshafen AG. Verfahren zur berührungslosen Messung des Winkels zwischen den Längsachsen eines Zugfahrzeugs und eines Anhängers oder zweier Anhänger, Patent DE 10 2014 224 80 8A. Deutsches Patent- und Markenamt, München, 2016.
- [Tsa87] R. Y. Tsai. A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology

- Using Off-the-Shelf TV Cameras and Lenses. *IEEE Journal on Robotics and Automation*, 3(4):323–344, 1987. DOI: 10.1109/JRA.1987.1087109. ISBN: 0867202947. ISSN: 0882-4967.
- [VKS<sup>+</sup>04] F. Vogt, S. Krüger, J. Schmidt, D. Paulus, H. Niemann, W. Hohenberger und C. H. Schick. Light fields for minimal invasive surgery using an endoscope positioning robot. *Methods of Information in Medicine*, 43(4):403–408, 2004. DOI: 10.1267/METH04040403. ISSN: 0026-1270.
- [VL01] E. Vincent und R. Laganière. Detecting planar homographies in an image pair. In *2nd International Symposium on Image and Signal Processing and Analysis*, Band 0, Seiten 182–187. IEEE, 2001. DOI: 10.1109/ISPA.2001.938625. ISBN: 953-96769-4-0.
- [WCG<sup>+</sup>19] Y. Wang, W.-L. Chao, D. Garg, B. Hariharan, M. Campbell und K. Q. Weinberger. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In *EEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Seiten 8445–8453. IEEE, 2019.
- [XTE<sup>+</sup>14] L. Xu, E. H. Tseng, T. Edward, S. Yellin, D. D. Hrovat und J. P. Joyce. Ford Global Technologies. Trailer motion and parameter estimation system, U.S. Patent 934 022 8B2. Patent and Trademark Office, Washington, DC, 2014.
- [ZDD10] S. Zinger, L. Do und P. H. N. De With. Free-viewpoint depth image based rendering. *Journal of Visual Communication and Image Representation*, 21(5-6):533–541, 2010. DOI: 10.1016/j.jvcir.2010.01.004. ISSN: 10473203.
- [Zha98] Z. Zhang. Determining the Epipolar Geometry and its Uncertainty: A Review. *International Journal of Computer Vision*, 27(2):161–195, 1998. DOI: 10.1023/a:1007941100561. ISSN: 0920-5691.
- [Zha00] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000. DOI: 10.1109/34.888718. ISBN: MSR-TR-98-71. ISSN: 01628828.
- [ZL15] J. Zbontar und Y. LeCun. Computing the Stereo Matching Cost With a Convolutional Neural Network. In



- IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Seiten 1592–1599. IEEE, 2015. DOI: 10.1109/CVPR.2015.7298767.
- [ZPW03] D. Zöbel, D. Pollock und P. Wojke. Steering assistance for backing up articulated vehicles. *Journal of Systemics, Cybernetics and Informatics*, 1(5):101–106, 2003.
- [ZPYT19] F. Zhang, V. Prisacariu, R. Yang und P. H. S. Torr. GA-Net: Guided Aggregation Net for End-to-end Stereo Matching. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [ZW18] F. Zhang und B. W. Wah. Fundamental Principles on Learning New Features for Effective Dense Matching. *IEEE Transactions on Image Processing*, 27(2):822–836, feb 2018. DOI: 10.1109/TIP.2017.2752370. ISSN: 1057-7149.



## INTERNETQUELLEN

---

- [@1] Daimler AG. Mercedes-Benz E-Klasse, 360°-Kamera Doppelanzeige, *Daimler Global Media Site*, 2016. <https://media.daimler.com/marsMediaSite/de/instance/picture.xhtml?oid=9269501>. Zuletzt abgerufen am 14.10.2019.
- [@2] Daimler AG. Mercedes-Benz Remote Park-Pilot: Um Parkvorgänge zu erleichtern, bietet die Mercedes-Benz E-Klasse als Sonderausstattung drei Varianten von Park-Assistenten: den Park-Pilot - optional mit 360° Kamera - sowie den neuen Remote Park-Pilot., *Daimler Global Media Site*, 2016. <https://media.daimler.com/marsMediaSite/de/instance/picture/Mercedes-Benz-Remote-Park-Pilot.xhtml?oid=9269690>. Zuletzt abgerufen am 14.10.2019.
- [@3] Matthias Brunnert. Verkehrsgerichtstag: Technik soll Zahl tödlicher Lkw-Unfälle senken, *Heise Online*, 2019. <https://heise.de/-4288545>. Zuletzt abgerufen am 14.10.2019.
- [@4] Statistisches Bundesamt und Kraftfahrt-Bundesamt. Bestand an Kraftfahrzeugen und Schienenfahrzeugen für die Jahre 2015 bis 2019, *DeStatis*, 2019. <https://www.destatis.de/DE/Themen/Branchen-Unternehmen/Transport-Verkehr/Unternehmen-Infrastruktur-Fahrzeugbestand/Tabellen/fahrzeugbestand.html>. Zuletzt abgerufen am 14.10.2019.
- [@5] Bundesamt für Güterverkehr. Anteil der Lkw an der Transportleistung im Güterverkehr in Deutschland in den Jahren von 2013 bis 2021 (laut Modal-Split), *Statista*, 2019. <https://de.statista.com/statistik/daten/studie/12195/umfrage/anteil-der-lkw-am-gueterverkehr-in-deutschland/>. Zuletzt abgerufen am 14.10.2019.
- [@6] Bundeszentrale für politische Bildung und Landeszentrale Politische Bildung Rheinland-Pfalz. Wahl-O-Mat Rheinland-Pfalz 2016, *Wahl-O-Mat*, 2016. <https://archiv.wahl-o-mat.de/rlp2016/>. Zuletzt abgerufen am 14.10.2019.
- [@7] Kristina Gnirke, Philipp Seibt und Ansgar Siemens. Streit über Notfallbremssysteme: Tödliche Laster, *Spiegel Online*, 2019. <https://spon.de/afv0X>. Zuletzt abgerufen am 14.10.2019.
- [@8] ZDF Heute. Antrag zu Abbiegeassistenten: Bundestag will Lkw sicherer machen, *ZDF heute*, 2018.

<https://www.zdf.de/nachrichten/heute/antrag-zu-abbiegeassistenten-bundestag-will-lkw-sicherer-machen-100.html>. Zuletzt abgerufen am 14.10.2019.

- [@9] Michael Kaess. AprilTags C++ Library, *Projektwebsite*, 2019. <http://people.csail.mit.edu/kaess/apriltags/>. Zuletzt abgerufen am 14.10.2019.
- [@10] Spiegel Online. Unternehmen setzen freiwillig auf Abbiegeassistenten, *Spiegel Online*, 2018. <https://spon.de/afgWD>. Zuletzt abgerufen am 14.10.2019.
- [@11] Dirk Rodenkirch. Abbiegeassistent - rechtlich kein Problem, *tagesschau.de*, 2019. <https://www.tagesschau.de/inland/abbiegeassistent-101.html>. Zuletzt abgerufen am 14.10.2019.
- [@12] Statistisches Bundesamt und Kraftfahrt-Bundesamt. Beförderungsmenge nach Hauptverkehrsrelationen und Verkehrsträgern, sowie die Veränderung zum Vorjahr in 1 000 Tonnen und die Veränderung zum Vorjahr in % für das Jahr 2018, *Destatis*, 2018. <https://www.destatis.de/DE/Themen/Branchen-Unternehmen/Transport-Verkehr/Gueterverkehr/Tabellen/verkehrstraeger-hauptverkehrs-relation-a.html>. Zuletzt abgerufen am 14.10.2019.

