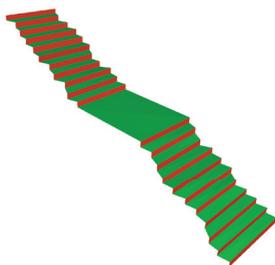
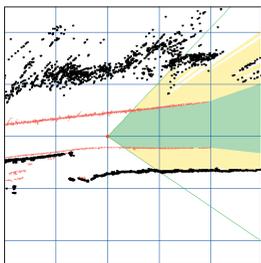


HANNES HARMS

Genauigkeitsuntersuchung von binokularen Normalenvektoren für die Umfeldwahrnehmung



Hannes Harms

**Genauigkeitsuntersuchung von
binokularen Normalenvektoren
für die Umfeldwahrnehmung**

Schriftenreihe
Institut für Mess- und Regelungstechnik,
Karlsruher Institut für Technologie (KIT)
Band 036

Eine Übersicht aller bisher in dieser Schriftenreihe erschienenen
Bände finden Sie am Ende des Buchs.

Genauigkeitsuntersuchung von binokularen Normalenvektoren für die Umfeldwahrnehmung

von
Hannes Harms

Dissertation, Karlsruher Institut für Technologie
KIT-Fakultät für Maschinenbau

Tag der mündlichen Prüfung: 9. Dezember 2016
Referenten: Prof. Dr.-Ing. Christoph Stiller
Prof. Dr.-Ing. Marcus Geimer

Impressum



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

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

www.ksp.kit.edu



*This document – excluding the cover, pictures and graphs – is licensed
under a Creative Commons Attribution-Share Alike 4.0 International License
(CC BY-SA 4.0): <https://creativecommons.org/licenses/by-sa/4.0/deed.en>*



*The cover page is licensed under a Creative Commons
Attribution-No Derivatives 4.0 International License (CC BY-ND 4.0):
<https://creativecommons.org/licenses/by-nd/4.0/deed.en>*

Print on Demand 2017 – Gedruckt auf FSC-zertifiziertem Papier

ISSN 1613-4214
ISBN 978-3-7315-0628-7
DOI 10.5445/KSP/1000065055

Genauigkeitsuntersuchung von binokularen Normalenvektoren für die Umfeldwahrnehmung

Zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften

der Fakultät für Maschinenbau
des Karlsruher Instituts für Technologie (KIT)
genehmigte

Dissertation

von

DIPL.-ING. HANNES HARMS

aus Stuttgart

Tag der mündlichen Prüfung: 9. Dezember 2016
Hauptreferent: Prof. Dr.-Ing. Christoph Stiller
Korreferent: Prof. Dr.-Ing. Marcus Geimer

Vorwort

Die vorliegende Dissertation entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Institut für Mess- und Regelungstechnik (MRT) des Karlsruher Instituts für Technologie (KIT). Herrn Prof. Dr.-Ing. Christoph Stiller danke ich herzlich für die Betreuung dieser Arbeit, die vielen konstruktiven Diskussionen und die Schaffung der ausgezeichneten Rahmenbedingungen für das wissenschaftliche Arbeiten.

Herrn Prof. Dr.-Ing. Marcus Geimer gilt mein Dank für die Übernahme des Korreferats und dem Interesse an meiner Arbeit.

Allen Kolleginnen und Kollegen am MRT möchte ich für die angenehme Zusammenarbeit und die vielen Diskussionen während der Kaffeerunden, Dienstreisen und Sommerseminare danken. Besonderen Dank gilt meinen Kollegen Eike Rehder, Tobias Schwarze, Tobias Strauß und Martin Lauer für das mühevoll Korrekturlesen dieser Arbeit.

Abschließend möchte ich mich herzlich bei meiner Familie für ihre vielfältige Unterstützung bedanken. Ihr ist diese Arbeit gewidmet.

Karlsruhe, im September 2016

Hannes Harms

Kurzfassung

In modernen Sicherheitsassistenzsystemen spielt die präzise Umfelderfassung eine zentrale Rolle für die Funktionssicherheit und -zuverlässigkeit. Zunehmend werden binokulare Kamerasysteme in diesem Bereich eingesetzt. Insbesondere die Rekonstruktion von dichten Punktwolken aus binokularen Kamerasystemen bietet die Möglichkeit, Objekte im Umfeld dreidimensional zu erfassen. Die Orientierung der Objekte liefert für viele Methoden der Umfeldwahrnehmung eine wertvolle Zusatzinformation. Diese Orientierungsschätzung unterliegt jedoch einer Unsicherheit, die abhängig von der betrachteten Szene und dem verwendeten Sensor variiert.

Wünschenswert ist deshalb eine Abschätzung der Orientierungsunsicherheit von Oberflächen aus binokularen Kamerasystemen, um die Funktionssicherheit von Assistenzsystemen abschätzen zu können.

In der vorliegenden Arbeit wird eine Genauigkeitsuntersuchung von Normalenvektoren aus binokularen Kamerasystemen vorgestellt, die eine Modellierung der Orientierungsunsicherheit der Normalenvektoren aus wenigen Parametern ermöglicht. Diese Orientierungsunsicherheit wird bestimmt, indem der Einfluss des konstanten Disparitätsrauschens bei der Variation der Position und Orientierung der betrachteten Oberfläche untersucht wird. Die Ergebnisse der Genauigkeitsuntersuchung werden verwendet, um zwei Methoden zur Umfelderfassung aus Orientierungsinformationen auszulegen. Sowohl die vorgestellte Freibereichsschätzung für Fahrzeuge als auch die Treppendetektion für Menschen mit eingeschränkter Sehfähigkeit stellen eine Verbesserung der Detektionsgenauigkeit gegenüber aus der Literatur bekannter Verfahren dar. Die Methoden werden mit realen Sequenzen evaluiert.

Schlagerworte: Genauigkeitsuntersuchung – Normalenvektoren – binokulares Kamerasystem – Freibereichsschätzung – Treppendetektion

Abstract

Precise environment perception plays a central role for functional reliability of modern safety assistance systems. Binocular camera systems are used increasingly in this area. In particular, the reconstruction of dense point clouds from binocular camera systems provides the potential to detect three-dimensional objects in the environment. The orientation of the objects provides valuable additional information for many perception methods. However the orientation estimation is subject to an uncertainty that depends highly on the observed scene and the used sensor.

Therefore an estimate of the surface orientation uncertainty of binocular camera systems is preferable in order to assess the reliability of assistance systems.

This orientation uncertainty is determined by analyzing the influence of the constant disparity noise and the variation of the position and orientation of the observed surface. The results of the accuracy analysis are used to design two methods for environment perception which use orientation information. The presented free space estimation for vehicles as well as the staircase detection for visually impaired persons represent an improvement in the detection accuracy compared to known procedures from literature. The methods are evaluated on the basis of real sequences.

Keywords: accuracy analysis – normal vectors – binocular camera system – free space estimation – stair detection

Inhaltsverzeichnis

1	Einleitung	1
1.1	Ziele der Arbeit	2
1.2	Struktur der Arbeit	3
2	Grundlagen	7
2.1	Modellierung von Kameras	7
2.1.1	Lochkameramodell	7
2.1.2	Kalibrierung	11
2.2	Binokulare Kamerasysteme	12
2.2.1	Tiefenrekonstruktion	13
2.2.2	Transformation von Normalenvektoren	14
2.3	Metrologie	16
2.3.1	Rauschen	16
2.3.2	Verteilungsdichtefunktion	17
2.3.3	Fortpflanzung von Messunsicherheiten	21
2.3.3.1	Linearisierung	21
2.3.3.2	Monte-Carlo-Simulation	22
2.4	Optimalfilter	23
3	Genauigkeitsuntersuchung von binokularen Normalenvektoren	27
3.1	Vorarbeiten	28
3.2	Simulationen	29
3.2.1	2d-Simulationen	30
3.2.1.1	Linearisierung	31
3.2.1.2	Monte-Carlo-Simulation	33
3.2.1.3	Vergleich von Linearisierung und Monte-Carlo-Methode	34
3.2.2	3d-Simulationen	45
3.3	Herleitung der optimalen Patchgröße für binokulare Normalenvektoren	47

3.4	Experimente	50
3.4.1	Untersuchung des Rauschverhaltens	53
3.4.2	Vergleich der Simulationsergebnisse mit Realdaten	55
3.5	Zusammenfassung	56
4	Anwendungsbeispiele mit binokularen Normalenvektoren	57
4.1	Genauigkeitsanforderungen der Normalenvektoren	58
4.2	Orientierungsbild	60
4.3	Freibereichsschätzung für Automobile	61
4.3.1	Stand der Technik	62
4.3.2	Methodik	64
4.3.2.1	Eingangsdaten	64
4.3.2.2	Mehrschichtiges Belegungsgitter	64
4.3.2.3	Freibereichsschätzung	67
4.3.3	Ergebnisse und Auswertung	69
4.4	Treppendetektion für Blindenassistenzsysteme	72
4.4.1	Stand der Technik	72
4.4.2	Methodik	76
4.4.2.1	Korrelationskosten	77
4.4.2.2	3d-Kantendetektion	78
4.4.2.3	Tracken der Liniensegmente	80
4.4.2.4	Treppenmodellierung	82
4.4.3	Systemkonfiguration	83
4.4.4	Ergebnisse und Auswertung	84
5	Zusammenfassung und Ausblick	91
A	Anhang	95
A.1	Transformation von Normalenvektoren zwischen Welt- und Bildkoordinaten	95
A.2	Berechnung der Jacobi-Matrix	96
A.3	Hauptkomponentenanalyse	97
A.4	Bordsteine	97
A.5	Automatisierungsgrade von Fahrzeugen	99
	Literaturverzeichnis	101

Symbolverzeichnis

Abkürzungen

2d/3d	2/3-dimensional
ADAS	Fahrerassistenzsystem, <i>engl.: Advanced Driver Assistance System</i>
ABS	Antiblockiersystem, <i>engl.: Anti-lock Braking System</i>
CPU	Hauptprozessor, <i>engl.: Central Processing Unit</i>
DEM	Digitales Höhenmodell, <i>engl.: Digital Elevation Model</i>
ESP	Elektronisches Stabilitätsprogramm, <i>engl.: Electronic Stability Program</i>
IMU	Intertiale Messeinheit, <i>engl.: Inertial Measurement Unit</i>
PCA	Hauptkomponentenanalyse, <i>engl.: Principal Component Analysis</i>
RGB-D	Kombination aus Farb- und Tiefeninformation, <i>engl.: Red Green Blue-Depth</i>
ROI	Interessensregion, <i>engl.: Region Of Interest</i>
SGBM	Stereomatcher, <i>engl.: Semi Global Block Matching</i>

Notationsvereinbarungen

Skalare	nicht fett, kursiv	$a, b, c, \sigma, \lambda, \dots$
Vektoren	fett, nicht kursiv	$\mathbf{a}, \mathbf{b}, \mathbf{c}, \boldsymbol{\sigma}, \boldsymbol{\lambda}, \dots$
Matrizen	fett, nicht kursiv, groß	$\mathbf{A}, \mathbf{B}, \mathbf{C}, \dots$
unsicherheitsbehaftete Größen	Dach	$\hat{a}, \hat{b}, \hat{C}, \dots$

Symbole (lateinisch)

0_C	Ursprung des Kamerakoordinatensystems mit den Achsen x_C, y_C, z_C
$0_L, 0_R$	Ursprung des linken (L) und rechten (R) Kamerakoordinatensystems
0_W	Ursprung des weltfesten Koordinatensystems

a', b', c', d'	Geradenpunkte in der Bildebene
A', B', C', D'	Geradenpunkte in Weltkoordinaten
$A(h(x))$	Fläche des Häufigkeitshistogramms
b_n	Binbreite des Histogramms
B	Basisbreite des binokularen Kamerasystems
C	Kostenfunktion
d	Disparität
d_{pix}	Abstand zweier Bildpunkte
d_{ps}	Seitenlänge eines Patches im Bild
d_{pp}	Euklidischer Abstand eines Bildpunkts zum Bildhauptpunkt
d_{Euc}	Euklidischer Abstand eines Weltpunkts zur Kamera
d_{min}	Mindestabstand
D_{JB}	Maß des Jarque-Bera-Tests
D_{KL}	Maß der Kullback-Leibler-Divergenz
D_{JS}	Maß der Jensen-Shannon-Divergenz
e	Messunsicherheit
$E(\cdot)$	Erwartungswert
f	Frequenz
$f(\mathbf{p}, \mathbf{n}_{disp})$	Funktion f
F	Brennweite der Kamera
F_u, F_v	Brennweiten der Kamera bzgl. der Kamerakoordinaten u, v
\mathbf{F}	Fundamentalmatrix
$g(\mathbf{x}, \mathbf{n}_{world})$	Funktion g
$h(X)$	Abbildungsfunktion der Zufallsvariablen X
h_i	Wert des i -ten Histogrammbins
$h(x)$	Häufigkeitshistogramm
h_{min}	Mindesthöhe einer Zelle eines Belegunggitters
I_{bin}	Binärbild
I_δ	Orientierungsbild
I_{rect}	rektifiziertes Bild (<i>engl.</i> : image)
\mathbf{J}	Jacobimatrix
K	σ -Intervall
\mathbf{l}	Epipolarlinie
\mathbf{M}	erweiterte Abbildungsmatrix
$\mathbf{n}_{world} = (n_x, n_y, n_z)^\top$	Normalenvektor in Weltkoordinaten
$\mathbf{n}_{disp} = (n_u, n_v, n_d)^\top$	Normalenvektor im Disparitätsraum

$\mathcal{N}(\mu, \sigma)$	Normalverteilung mit Mittelwert μ und Standardabweichung σ
n_t	Normierungsfaktor des Musters t
n_{min}	Mindestanzahl an Liniensegmenten
o_{min}	Mindestorientierung einer Zelle eines Belegunggitters
$p_X(x)$	Wahrscheinlichkeitsdichtefunktion der Zufallsvariablen X
p_δ^{LIN}	Dichtefunktion der Gauß'schen Fehlerfortpflanzung
p_δ^{MC}	Dichtefunktion der Monte-Carlo-Simulation
$\mathbf{p} = (u, v, d)^\top$	Bildpunkt mit Disparitätswert
$\mathbf{p}_{pp} = (c_u, c_v)^\top$	Bildhauptpunkt (<i>engl.</i> : principal point)
p_i	Korrelationsmaske mit Index i
P_X	Wahrscheinlichkeitsfunktion der Zufallsvariablen X
\mathbf{P}	Projektionsmatrix
R	Kreuzkorrelationsfunktion
\mathbf{R}	Rotationsmatrix
s	Schiefe
s_p	Höhenparameter der Korrelationsmaske
S	Probenmenge
t	Korrelationsmuster (<i>engl.</i> : template)
\mathbf{t}	Translationsvektor
T_p	Zeitintervall
u, v	Bildkoordinaten eines Pixels
v_i	Wert des normierten und korrigierten i-ten Histogrammbins
w	Wölbung
x, y, z	Weltkoordinaten
x	Realisierung der Zufallsvariablen X
\hat{x}	unsicherheitsbehaftete Messgröße x
$x(t), y(t)$	Signale der Zeit t
$x^*(t)$	komplex-konjugierte von $x(t)$
X	Zufallsvariable
$\mathbf{x} = (x, y, z)^\top$	Punktvektor in Weltkoordinaten
$\tilde{\mathbf{x}} = (x, y, z, 1)^\top$	erweiterter Punktvektor in Weltkoordinaten
$\mathbf{x}_C = (x_C, y_C, z_C)^\top$	Punktvektor in Koordinatensystem 0_C

Symbole (griechisch)

α	Winkel zur x-Achse
γ	95%-Konfidenzintervall
δ	Differenzwinkel
δ_{norm}	normierte Winkelabweichung
Δh_{step}	Höhenunterschied von Liniensegmenten
Δo_{step}	Orientierungsdifferenz von Liniensegmenten
Δh_{stair}	Höhenunterschied von Stufen
Δo_{stair}	Orientierungsdifferenz von Stufen
$\Phi(x)$	Standardnormalverteilung
μ	Mittelwert
σ	Standardabweichung
σ^2	Varianz
ς	Scheerungsparameter
τ	Zeitverschiebung
θ, φ	Winkel der Polarkoordinatendarstellung

1 Einleitung

In den letzten Jahren sind erhebliche Fortschritte bei Fahrerassistenzsystemen im Automobil erzielt worden. Die zentrale Motivation für diese Entwicklung ist neben dem Komfortgewinn zunehmend die Erhöhung der Sicherheit im Straßenverkehr. Neben den seit Jahren eingeführten Funktionen der aktiven Sicherheit, wie ABS und ESP, bieten moderne Serienfahrzeuge bereits teilautomatisierte Assistenzfunktionen für Autobahnen, Landstraßen und den Stadtverkehr an. Laut Definition übernimmt ein teilautomatisiertes System Aufgaben der Längs- und Querverführung des Fahrzeugs, der Fahrer ist aber dauerhaft verpflichtet, das System zu überwachen [16]. Grundvoraussetzung aller Assistenzsysteme ist neben der Kenntnis des Fahrzeugzustands eine präzise Fahrzeugumfelderfassung, nur so können sichere und zuverlässige automatisierte Funktionen gewährleistet werden. Stand der Technik für die Umfelderfassung ist eine Vielzahl unterschiedlicher Sensoren, wie Radar, Lidar, Videokameras und Ultraschallsensoren. Sie werden zukünftig durch Informationen aus Car-to-Car und Car-to-Infrastructure Kommunikationen ergänzt. Die Verwendung verschiedener Sensoren stellt eine robuste Umfelderfassung sicher, da eine Sensordatenfusion störende Mess- und Umfeldeinflüsse durch die unterschiedlichen physikalischen Messprinzipien eliminieren kann.

Als vielversprechende Sensortechnologie bewähren sich Kamerasysteme in Forschungsprojekten und in Serienfahrzeugen mehr und mehr. Der Grund dafür ist neben der laufend erhöhten Leistungsfähigkeit von Bildaufnehmer und Optik die gleichzeitige Steigerung der Rechenleistung, so dass die entsprechenden Bildverarbeitungsalgorithmen in Echtzeit berechnet werden können. Kamerasysteme stellen damit eine leistungsstarke, gut integrierbare und kostengünstige Lösung zur Realisierung vieler Funktionen dar. Monokulare Kameras werden beispielsweise zur Verkehrszeichen-, Ampel- und Fahrbahnmarkierungserkennung genutzt. Seit einigen Jahren kommen auch sogenannte binokulare Kamerasysteme zum Einsatz. Sie bestehen aus zwei Kameras mit überlappendem Sichtbereich, die zumeist in Fahrtrichtung orientiert sind. Diese Anordnung ermöglicht insbesondere eine Tiefenrekonstruktion der Umgebung durch Triangulation der Punktkorrespondenzen zwischen den Kameras. Diese Abstandsinformation wird als Disparitätsbild bezeichnet und kann gut zur Schätzung des Freibereichs und zur Detektion von Objekten genutzt werden. Damit sind Sicherheitsassistenzsys-



Bild 1.1: Sensorsetup des Forschungsfahrzeugs 'Bertha One' des Instituts für Mess- und Regelungstechnik (MRT) am Karlsruher Institut für Technologie (KIT)

teme beispielsweise in der Lage, automatische Notbremsungen und Ausweichmanöver auszuführen.

Auch im Bereich der Assistenzsysteme für Menschen mit eingeschränkter Sehfähigkeit werden Ansätze zur Navigationshilfe erforscht. In diesem Feld gibt es die Möglichkeit, mithilfe von binokularen Kamerasystemen die sicherheitsrelevanten Umfeldinformationen den betroffenen Menschen zu vermitteln. Insbesondere in unbekanntem oder sich dynamisch ändernden Umgebungen sind diese Zusatzinformationen für die Betroffenen sehr wertvoll. Die relevanten Zusatzinformationen können beispielsweise durch ein akustisches oder taktiles Feedbacksignal gegeben werden. Das zur Umfelderkennung verwendete binokulare Kamerasystem könnte in Kopfhöhe getragen werden, integriert in einer Brille oder einem Helm.

1.1 Ziele der Arbeit

Die vorliegende Arbeit leistet einen Beitrag im Bereich der Umfeldwahrnehmung mithilfe von binokularen Kamerasystemen. Im Speziellen wird die Genauigkeit der Oberflächenschätzung aus Disparitätsbildern binokularer Kameras untersucht. Die zu erwartende Qualität der Oberflächenorientierung bekannter

binokularer Kamerasysteme wird über die Genauigkeitsuntersuchung hergeleitet.

Die Ziele dieser Arbeit liegen zum einen in der Vorhersage der Orientierungsgenauigkeit der Oberflächenschätzung aus Disparitätsbildern und zum anderen im Nutzen von Oberflächenorientierungen für Anwendungen zur Umfeldwahrnehmung.

Um die lokale Oberfläche eines 3d-Punkts zu bestimmen, wird eine Ebene durch den betrachteten Punkt und seine Nachbarpunkte geschätzt. Für jede lokal betrachtete Oberfläche ist deren geschätzte Orientierung maßgeblich. Der Fehler der Oberflächenorientierung wird besonders durch die Tiefenunsicherheit beeinflusst und hängt damit hauptsächlich vom Disparitätsrauschen ab. Für ein bekanntes binokulares Kamerasystem wird dieser Einfluß auf die geschätzte Oberflächenorientierung untersucht. Für Assistenzfunktionen lassen sich damit die Grenzen ihres Funktionsbereichs definieren. Ebenso besteht damit die Möglichkeit die Anforderungen an den Sensor abzuleiten, wenn die Anforderungen an die Assistenzfunktion bekannt sind.

Praktischerweise soll die Position eines betrachteten Oberflächenpunkts aus einer dichten 3d-Punktwolke ausreichen, um die Genauigkeit der Oberflächenorientierung abschätzen zu können. Dazu sollen alle relevanten Kameraparameter ermittelt werden, von denen die Orientierungsunsicherheit abhängt. Insbesondere soll festgestellt werden, ob mithilfe einer Linearisierung eine Abschätzung der Orientierungsgenauigkeit in ausreichender Qualität getroffen werden kann oder ob dazu eine Monte-Carlo-Methode benötigt wird. Anschließend werden in realen Experimenten die Schlussfolgerungen der simulativen Untersuchungen validiert.

Für automatisierte Fahrzeuge wird eine Freibereichsschätzung und für Menschen mit eingeschränkter Sehfähigkeit eine Treppendetektion vorgestellt. Es soll der Nutzen der verwendeten Orientierungsinformation für beide Anwendungen gezeigt werden. Zudem soll die Funktionssicherheit der Assistenzfunktionen in der Planungsphase sichergestellt werden, indem eine Abschätzung der Orientierungsgenauigkeit der verwendeten Oberflächeninformationen vorgenommen wird.

1.2 Struktur der Arbeit

Die vorliegende Arbeit ist in folgende Kapitel gegliedert:

In **Kapitel 2** werden die Grundlagen von Kamerasystemen erläutert, die in dieser Arbeit vorausgesetzt oder verwendet werden. Nach der Einführung der

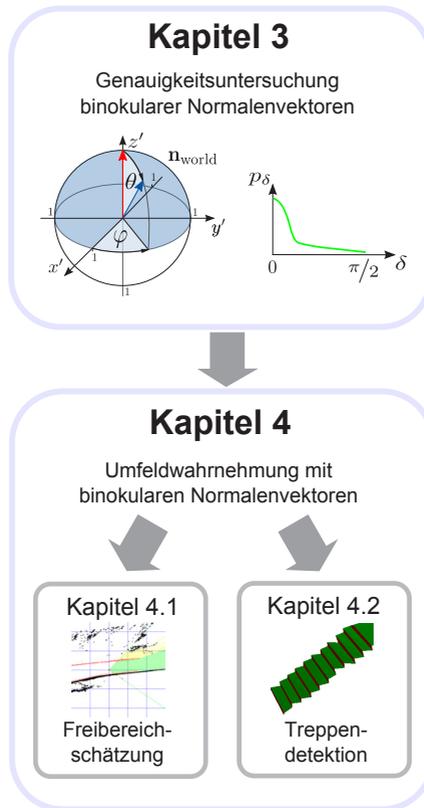


Bild 1.2: Thematischer Aufbau der Hauptkapitel dieser Arbeit in Blöcken

Modellierung monokularer und binokularer Kamerasysteme und deren Kalibrierung wird die Tiefenrekonstruktion analysiert. Die daraus abgeleiteten Positionsunsicherheiten werden mittels einer Linearisierung hergeleitet. Anschließend wird eine allgemeingültige Transformation von Normalenvektoren zwischen Disparitätsraum und Weltkoordinaten ermittelt. Die zugehörigen messtechnischen Grundlagen werden erläutert. Neben den Rauscheffekten in Kamerasystemen wird eine Einführung in Verteilungsdichtefunktionen und deren Divergenzmaße gegeben. Die Fortpflanzung von Messunsicherheiten wird anhand von Linearisierung und Monte-Carlo-Simulation eingeführt. Das Kapitel schließt mit der Beschreibung des in der Bildverarbeitung weit verbreiteten Korrelationsfilters.

Kapitel 3 untersucht die Genauigkeit von Normalenvektoren, die aus binokularen Kamerasystemen bestimmt werden. Diese werden im Folgenden als binokulare Normalenvektoren bezeichnet. Die Unsicherheit der Oberflächenorientierung wird durch Variation der Position des Oberflächenpunktes und der Ausrichtung der Oberfläche analysiert. Anhand verschiedener Simulationen wird eine Monte-Carlo-Methode mit einer Lösung durch Linearisierung verglichen. So können die Haupteinflussgrößen identifiziert werden, die eine Modellierung der Orientierungsunsicherheit der Oberfläche bezüglich der Position erlauben. Aus diesen Ergebnissen lassen sich zwei Methoden ableiten, die zur Berechnung von binokularen Normalenvektoren eingesetzt werden können. Abschließend werden die Simulationsergebnisse mit Experimenten realer binokularer Kamerasysteme verglichen.

Kapitel 4 beschreibt zwei mögliche Anwendungsszenarien für binokulare Normalenvektoren: eine Freibereichsschätzung und eine Treppendetektion. Beide Anwendungen berücksichtigen die Oberflächenorientierung der berechneten Normalenvektoren, indem sie auf ein rotationsinvariantes Orientierungsbild zurückgreifen.

Die Freibereichsschätzung detektiert große Objekte aus der 3d-Punktwolke und zeigt, dass mithilfe des Orientierungsbilds zusätzlich niedrige Infrastrukturelemente als Freibereichsbegrenzung erkannt werden können. Dazu wird eine nicht-parametrische Freibereichsform in einem mehrschichtigen Belegungsgitter in Weltkoordinaten geschätzt. Das Belegungsgitter nimmt eine zeitliche Filterung der detektierten Objekte vor. Daher eignet sich die Freibereichsschätzung insbesondere für lange Erfassungssequenzen, wie sie typischerweise in Fahrzeuganwendungen vorzufinden sind.

Die Treppendetektion verwendet ein Optimalfilter auf dem Orientierungsbild, um konkave und konvexe 3d-Liniensegmente zu schätzen, die anschließend zeitlich getrackt werden. Diese Verfolgung ermöglicht die Detektion einzelner Stufen, zusammenpassende Stufen werden zu Treppen fusioniert. Die Evaluation wird auf einer vorher vermessenen Treppe durchgeführt, das Ergebnis ist ein hochpräzises Treppenmodell. Auch diese Methode spielt ihre Stärken in langen Sequenzen aus und ist allen aus der Literatur bekannten Ansätzen insbesondere bzgl. der Präzision überlegen.

Kapitel 5 fasst die wesentlichen Aussagen dieser Arbeit zusammen und gibt einen Ausblick über mögliche fortführende Forschungsarbeiten.

2 Grundlagen

Dieses Kapitel gibt zunächst eine Einführung in die theoretischen Grundlagen der Modellierung und Kalibrierung monokularer Kamerasysteme. Im Weiteren werden der Aufbau und die Eigenschaften binokularer Kamerasysteme beschrieben. Insbesondere wird die Tiefenrekonstruktion der Szene und die Schätzung von Normalenvektoren aus dem binokularen Kamerasystem betrachtet. Für die Untersuchung der Positionsunsicherheiten der 3d-Punkte aus binokularen Kamerasystemen wird die Fortpflanzung von Messunsicherheiten näher betrachtet. Dies bildet die Voraussetzung der in Kapitel 3 vorgestellten Verfahren zur Untersuchung der Orientierungsgenauigkeit von Normalenvektoren aus 3d-Punktwolken binokularer Kamerasysteme. Eine Einführung in die Korrelationsfilterung für Bildverarbeitungssysteme bildet den Abschluss des Kapitels.

2.1 Modellierung von Kameras

Im Folgenden wird eine allgemeine Einführung in die Modellierung von monokularen Kamerasystemen gegeben. Speziell werden der Aufbau des Lochkameramodells und dessen Eigenschaften, sowie die Kalibrierung von Kamerasystemen betrachtet.

2.1.1 Lochkameramodell

Monokulare Kameras bilden Objektpunkte aus dem dreidimensionalen (3d) Raum auf eine zweidimensionale (2d) diskrete Bildebene ab. Dabei kommt es offensichtlich zu einem Informationsverlust durch die Dimensionsreduktion und die Diskretisierung der Information auf Pixel der Bildebene bei digitalen Kameras. Dieser Abbildungsprozess wird im Allgemeinen durch eine zentrale Projektion mittels eines Lochkameramodells beschrieben. Kameraoptiken mit gewöhnlichen Öffnungswinkeln können damit ausreichend gut angenähert werden. Kameraoptiken mit außergewöhnlichen Öffnungswinkeln werden durch andere mathematische Modelle betrachtet und werden in [62] näher behandelt.

Der Aufbau eines Lochkameramodells [51][22] ist in Bild 2.1 dargestellt. Das Kamerakoordinatensystem 0_C , und damit die Position der Kamera, ist eindeutig

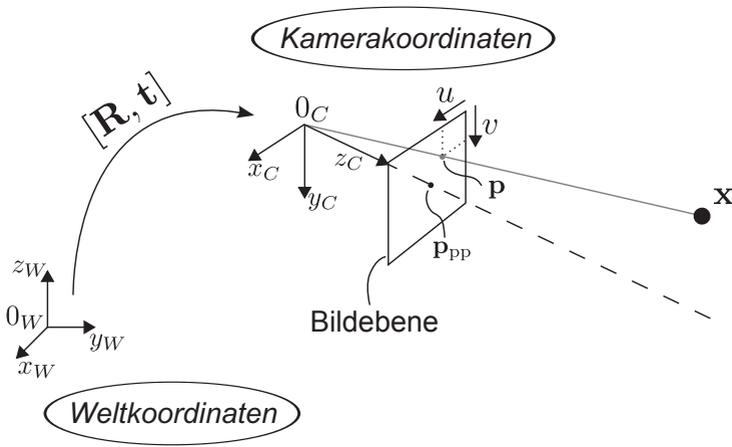


Bild 2.1: Perspektivische Abbildung eines Lochkammermodells und verwendete Koordinatensysteme

durch eine Rotationsmatrix $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ und einen Translationsvektor $\mathbf{t} \in \mathbb{R}^3$ relativ zu einem weltfesten Koordinatensystem 0_W beschrieben. \mathbf{R} und \mathbf{t} werden als extrinsische Kameraparameter bezeichnet. Um Beobachtungen von Kameras zeitlich zu integrieren, wird oft das weltfeste Koordinatensystem 0_W als Referenz genutzt. Dabei besteht folgende Beziehung zwischen $\mathbf{x}_W = (x_W, y_W, z_W)^\top$ in 0_W und $\mathbf{x}_C = (x_C, y_C, z_C)^\top$ in 0_C eines Objektpunkts \mathbf{x} :

$$\mathbf{x}_C = \mathbf{R}\mathbf{x}_W + \mathbf{t} \quad (2.1)$$

Bei Verwendung von homogenen affinen Koordinaten kann die Rotation und Translation zwischen 0_W und 0_C durch eine Multiplikation mit der erweiterten Abbildungsmatrix $\mathbf{M} \in \mathbb{R}^{4 \times 4}$ ersetzt werden:

$$\bar{\mathbf{x}}_C = \mathbf{M}\bar{\mathbf{x}}_W, \text{ mit } \mathbf{M} = \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{pmatrix}. \quad (2.2)$$

Diese affine Transformation stellt bei der Berechnung von mehrfachen Rotationen und Translationen der Kamera eine Vereinfachung dar und wird z.B. bei der Kompensation der Eigenbewegung von Kameras verwendet (siehe Kapitel 4.4.2.3). Dazu werden die Position und Orientierung (Pose) der Kamera zu mehreren Zeitpunkten ineinander überführt, siehe Bild 2.2. Die resultierende Pose \mathbf{M}_{res} nach n Zeitschritten ergibt sich aus dem Produkt der erweiterten Abbildungsmatrizen.

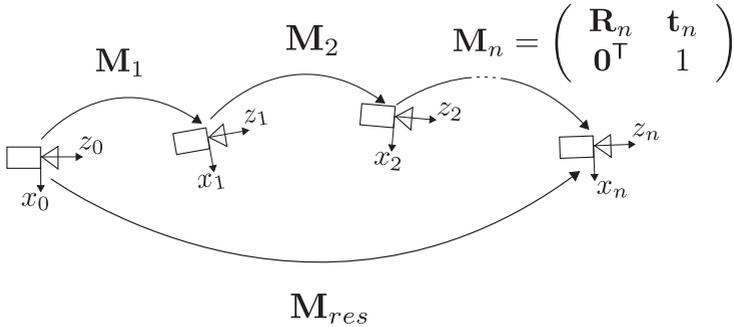


Bild 2.2: Schematische Darstellung der Kompensation der Kamerabewegung mit affinen Transformationen

$$\mathbf{M}_{res} = \begin{pmatrix} \mathbf{R}_{res} & \mathbf{t}_{res} \\ \mathbf{0}^T & 1 \end{pmatrix} = \prod_{i=1}^n \begin{pmatrix} \mathbf{R}_i & \mathbf{t}_i \\ \mathbf{0}^T & 1 \end{pmatrix} \quad (2.3)$$

Das euklidische Kamerakoordinatensystem 0_C hat im optischen Zentrum seinen Ursprung. Das optische Zentrum entspricht einer infinitesimal kleinen Lochblende, welche alle in die Kamera einfallenden Sichtstrahlen passieren. Daraus folgt, dass jeder Objektpunkt \mathbf{x}_W im Weltkoordinatensystem eindeutig auf einen Bildpunkt \mathbf{p} abgebildet wird. Der Bildhauptpunkt $\mathbf{p}_{pp} = (c_u, c_v)^T$ (engl.: principal point) beschreibt den Schnittpunkt der Bildebene mit dem Sichtstrahl, der senkrecht auf der Bildebene steht. Dieser Sichtstrahl entspricht der z_C -Achse des Kamerakoordinatensystems und wird als optische Achse bezeichnet. Unter Berücksichtigung des Kamerahauptpunkts und der Brennweite F kommt man zu folgender Abbildungseigenschaft:

$$\begin{pmatrix} u \\ v \end{pmatrix} = \frac{F}{z} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} c_u \\ c_v \end{pmatrix} \quad (2.4)$$

Diese Abbildung ist nicht umkehrbar und daher keine bijektive Abbildung.

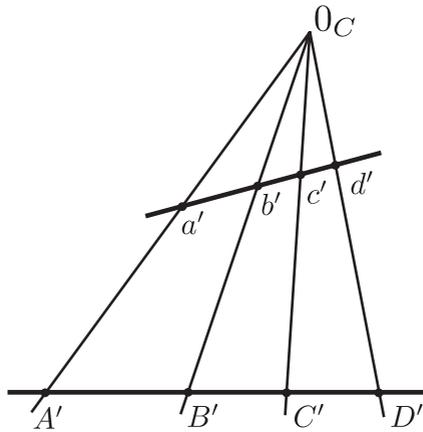


Bild 2.3: Invarianz der Doppelverhältnisse zwischen Bild- und Weltkoordinaten

Die zentrale Projektion des Lochkameramodells weist folgende Eigenschaften auf, wie in [61] beschrieben:

- Jeder Punkt des Raums entspricht genau einem Bildpunkt.
- Jedem Bildpunkt können unendlich viele Raumpunkte zugeordnet werden, die alle auf der Geraden des Sichtstrahls liegen, der durch das optische Zentrum und den Bildpunkt geht.
- Parallele Geraden in Weltkoordinaten bilden eine Geradenschar in der Bildebene, die sich in einem Fluchtpunkt schneiden. Der Fluchtpunkt entspricht einem unendlich entfernten Punkt.
- Jede Bildgerade spannt mit dem Projektionszentrum eine Ebene auf.
- Es gilt die Invarianz der Doppelverhältnisse der Strecken zwischen vier beliebigen Geradenpunkten A', B', C', D' in Weltkoordinaten und den projizierten Geradenpunkten a', b', c', d' auf der Bildebene (siehe Bild 2.3), wie z.B.

$$\overline{A'B'} : \overline{C'D'} = \overline{a'b'} : \overline{c'd'} \quad (2.5)$$

Das Verhältnis der Teilverhältnisse bleibt damit erhalten.

2.1.2 Kalibrierung

Jedes Kamerasystem weist eigene Abbildungseigenschaften der 3d-Umgebung auf das Kamerabild auf. Dies ist durch Ungenauigkeiten in der Herstellungskette aller Einzelkomponenten eines optischen Systems begründet. Die intrinsische Kalibrierung bestimmt dieses Abbildungsverhalten eines optischen Systems in der Regel durch ein parametrisches Modell. Bei Multi-Kamera-Systemen wird durch die extrinsische Kalibrierung die relative Lage der Kameras zueinander bestimmt. Die in dieser Arbeit betrachteten perspektivischen Kameras verwenden als Projektionsmodell das Lochkameramodell. Das Verzeichnungsmo- dell beschreibt die Abweichung zu einem idealen Projektionsmodell durch radiale und tangentiale Verzeichnungen [37]. Die Bildpaare binokularer Kamerasysteme werden oft durch eine Rektifizierung so entzerrt, dass dieselben Tiefenpunkte in den Bildpaaren in derselben Bildreihe wiedergefunden werden können (siehe Abschnitt 2.2). Dazu werden zwei virtuelle Kamerabildebenen mit gleicher Orientierung gewählt, welche die Epipolargeometrie [22] erfüllen. Bei binokularen Kamerasystemen wird die Basisbreite B bestimmt, die maßgeblich für die Tiefenrekonstruktion ist.

Während viele Kalibrierungen ebene Schachbretter mit bekannter Größe verwenden (Offline-Kalibrierung), gibt es auch Verfahren, welche die Kalibrierparameter iterativ zur Laufzeit bestimmen (Online-Kalibrierung). Da sich Kamerasysteme unter dynamischen Bedingungen (z.B. durch Temperaturschwankungen) verändern können, bietet die Online-Kalibrierung eine adaptive Schätzung der Kalibrierparameter zur Laufzeit. Eine Übersicht der verschiedenen Kalibrierverfahren findet sich in [62].

Nach der Entzerrung des Kamerabilds, beinhaltet die (3×3) -Kalibriermatrix \mathbf{K} die intrinsischen Kameraparameter.

$$\mathbf{K} = \begin{pmatrix} F_u & \zeta & c_u \\ 0 & F_v & c_v \\ 0 & 0 & 1 \end{pmatrix} \quad (2.6)$$

Es entsprechen sich die Brennweiten $F_u = F_v = F$ in Pixeln, wenn der Kamerachip aus quadratischen Pixeln aufgebaut ist. Die übrigen Parameter sind der Bildhauptpunkt $\mathbf{p}_{pp} = (c_u, c_v)^T$ und der Scherungsparameter ζ , der die Verkipfung zwischen der optischen Achse des Objektivs und dem Bildsensor beschreibt. Aus einem Weltkoordinatenpunkt $\tilde{\mathbf{x}}_h = (x, y, z, 1)$ lassen sich mithilfe der (3×4) -Projektionsmatrix $\mathbf{P} = \mathbf{K}[\mathbf{R}, \mathbf{t}]$ die homogenen Koordinaten des Bildpunkts $\mathbf{p}_h = (u, v, w)$ bestimmen.

$$\mathbf{p}_h = \mathbf{P}\tilde{\mathbf{x}}_h \quad (2.7)$$

2.2 Binokulare Kamerasysteme

Binokulare Kamerasysteme bestehen aus zwei Kameras. Aus praktischen Gründen überlappen die Sichtbereiche der Kameras oft, um eine Tiefenrekonstruktion durchführen zu können. Ausgewählte Sichtstrahlen der Bildreihe eines binokularen Kamerasystems sind in Bild 2.4 skizziert. Ersichtlich ist, dass der Verlauf der Sichtstrahlen nur im Nahbereich genaue Tiefenmessungen zulässt und die Tiefenunsicherheit quadratisch mit der Entfernung zunimmt.

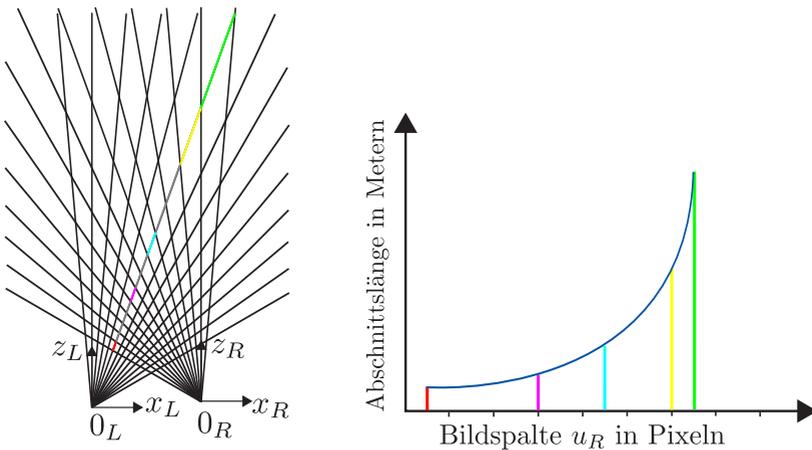


Bild 2.4: Links: Binokulares Kamerasystem mit skizzierten Sichtstrahlen gleicher Winkeldifferenz (5°). Rechts: Zunahme der Abschnittslänge (skaliert, farbig dargestellt), die auf einem Sichtstrahl der linken Kamera liegen und mit den Sichtstrahlen der rechten Kamera geschnitten werden.

Die geometrische Beziehung korrespondierender Bildpunkte in beiden Kamerabildern wird durch die Epipolareometrie beschrieben. Die Epipolarebene wird von den beiden optischen Zentren der Kameras und dem Objektpunkt im dreidimensionalen Raum aufgespannt. Die Schnittgeraden der Epipolarebene mit den Bildebenen werden als Epipolarlinien bezeichnet. Die Fundamentalmatrix \mathbf{F} stellt die algebraische Repräsentation der Epipolareometrie dar und überführt einen Bildpunkt einer Kamera \mathbf{p} in eine Epipolarlinie l der andere Kamera: $l = \mathbf{F}\mathbf{p}$. Die essentielle Matrix ist ein Spezialfall der Fundamentalmatrix, wenn generalisierte (homogene) Koordinaten verwendet werden. Bei kalibrierten binokularen Kamerasystemen kann die essentielle Matrix verwendet werden.

Nach dem Kalibrieren der Kameras werden die Epipolarlinien in der Regel horizontal in der Bildebene ausgerichtet, damit die Korrespondenzsuche auf ein rechenzeitoptimales, eindimensionales Suchproblem reduziert werden kann. Im Folgenden werden diese rektifizierten Bilder als I_{rect} bezeichnet. Der Abstand der optischen Zentren der Kameras entspricht der Basisbreite B .

2.2.1 Tiefenrekonstruktion

Ein Stereomatcher erzeugt mittels Korrespondenzsuche aus rektifizierten Bildpaaren ein Disparitätsbild, das die Verschiebung vom rechten zum linken Bild in Pixeln repräsentiert. Die dafür benötigten Bildpunkt-korrespondenzen werden auf den Epipolarlinien gesucht, die bei rektifizierten Bildern I_{rect} entlang der Bildzeilen ausgerichtet sind. Frontoparallele Flächen können von klassischen Block Matchern gut rekonstruiert werden, wie beispielsweise beim Semi-Global Block Matcher [24]. Anderweitig orientierte Flächen können durch Strecken, Stauchen und Scheren der rektifizierten Grauwertbilder in Kombination mit einem Blockmatcher genauer rekonstruiert werden [48]. Bei derzeitigen Stereomatchern wird eine qualitative Genauigkeit der Disparitätswerte von ungefähr $\frac{1}{8}$ Pixel erreicht.

Dicht besetzte Disparitätsbilder werden durch Stereomatches rektifizierter Bildpaare von binokularen Kamerasystemen erzeugt, so dass eine Disparität d zu jeder Pixelposition $(u, v)^T$ des linken Kamerabilds zugewiesen wird. Folglich werden Oberflächen zunächst als Menge von Punkten $\mathbf{p} = (u, v, d)^T$ im Disparitätsraum repräsentiert. Für jeden dieser Punkte kann die 3d Position $\mathbf{x} = (x, y, z)^T$ im Kamera-Weltkoordinatensystem durch die Funktion $r : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ rekonstruiert werden:

$$\mathbf{x} = r(\mathbf{p}) = \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \frac{B}{d} \begin{pmatrix} u - c_u \\ v - c_v \\ F \end{pmatrix}, \quad (2.8)$$

Die Brennweite F und der Bildhauptpunkt $\mathbf{p}_{pp} = (c_u, c_v)^T$ sind die kalibrierten Lochkamera-Parameter der beiden identischen Einzelkameras. B stellt die Basisbreite des binokularen Kamerasystems dar. Die Projektion eines beliebigen Punkts $\mathbf{x} = (x, y, z)^T$ in den Disparitätsraum kann direkt aus Formel 2.8 bestimmt werden:

$$\mathbf{p} = s(\mathbf{x}) = \begin{pmatrix} u \\ v \\ d \end{pmatrix} = \frac{F}{z} \begin{pmatrix} x \\ y \\ B \end{pmatrix} + \begin{pmatrix} c_u \\ c_v \\ 0 \end{pmatrix}, \quad (2.9)$$

Die Sensitivität der Punktmessungen in Weltkoordinaten gegenüber unsicherheitsbehafteter Disparitätsmessungen sind an der Jacobi-Matrix (siehe Anhang A.2) sichtbar:

$$\mathbf{J}_{r,\mathbf{p}} = \begin{pmatrix} \frac{\partial x}{\partial u} & \frac{\partial x}{\partial v} & \frac{\partial x}{\partial d} \\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial v} & \frac{\partial y}{\partial d} \\ \frac{\partial z}{\partial u} & \frac{\partial z}{\partial v} & \frac{\partial z}{\partial d} \end{pmatrix} = \begin{pmatrix} \frac{B}{d} & 0 & -\frac{B(u-c_u)}{d^2} \\ 0 & \frac{B}{d} & -\frac{B(v-c_v)}{d^2} \\ 0 & 0 & -\frac{BF}{d^2} \end{pmatrix} \quad (2.10)$$

Unsicherheiten bezüglich der Bildkoordinaten u und v resultieren in linearen Abweichungen. Unsicherheiten in der Tiefe wachsen jedoch quadratisch mit konstant zunehmendem Abstand und bedürfen deshalb einer besonderen Berücksichtigung. Ist nur die Disparität unsicherheitsbehaftet, ist die Verteilung der rekonstruierten Punkte nur auf den Sichtstrahlen der Kamera zu erwarten. In Bild 2.4 sind die Sichtstrahlen eines binokularen Kamerasystems skizziert. Unsicherheiten in der Disparität bewirken aus der geometrischen Betrachtung eine quadratische Zunahme der Tiefenunsicherheit entlang eines Sichtstrahls. Dieser Effekt in der Tiefenunsicherheit stimmt mit der Interpretation der Jacobimatrix überein. Wird das Tiefenrauschen als Gauß-normalverteilt im Disparitätsraum angenommen, entspricht die Verteilung des Tiefenrauschens in Weltkoordinaten einer schiefen Verteilungsfunktion. Dies ist auf die nicht-lineare Abbildung aus Gleichung 2.8 zwischen Welt- und Disparitätskoordinaten zurückzuführen.

Um spezifische Anforderungen an die Genauigkeit der Tiefenrekonstruktion zu erfüllen, können Auflösung, Basisbreite und Brennweite des binokularen Kamerasystems gewählt werden.

2.2.2 Transformation von Normalenvektoren

Aufgrund der nicht-linearen Abbildung zwischen Disparitätsraum und Weltkoordinaten gibt es keine winkeltreue (konforme) Abbildungen zwischen den beiden euklidischen Räumen. Oberflächennormalen werden aufgrund der mittelwertfreien Normalverteilung der Disparitäten im Disparitätsraum geschätzt, wie in [5] empfohlen wird. Um die geschätzten Oberflächenorientierungen in Weltkoordinaten auswerten zu können, ist eine Transformation von Normalenvektoren an einem bestimmten Oberflächenpunkt zwischen Disparitätsraum und Weltkoordinaten erforderlich. Diese Transformation wird im Folgenden gezeigt.

Eine wichtige und weithin bekannte Eigenschaft ist, dass die Funktion r (Gleichung 2.8) Ebenen aus dem Disparitätsraum auch als Ebenen im Weltkoordinatensystem abbildet. D.h. die vier Welt-Punkte $r(\mathbf{p}_0)$, $r(\mathbf{p}_1)$, $r(\mathbf{p}_2)$ und $r(\mathbf{p}_3)$ sind genau dann koplanar, wenn auch \mathbf{p}_0 , \mathbf{p}_1 , \mathbf{p}_2 und \mathbf{p}_3 im Disparitätsraum

koplanar sind [14]. Aufgrund dieser Eigenschaft kann die Oberflächennormale zuerst im Disparitätsraum bestimmt und anschließend in Weltkoordinaten transformiert werden. Dies ist aus zwei Gründen vorteilhaft: Erstens ist es notwendig zur Berechnung des Normalenvektors an jedem Oberflächenpunkt die direkt benachbarten Punkte zu bestimmen. Diese Punkte werden verwendet um lokal eine Ebene zu schätzen. Der damit lokal geschätzte Normalenvektor approximiert den Normalenvektor der betrachteten Oberfläche. Im Disparitätsraum kann die Nachbarschaft eines Oberflächenpunktes direkt aus den benachbarten Pixeln im Disparitätsbild bestimmt werden. Zweitens wird der Oberflächen-Normalenvektor genauer bestimmt, wenn die lokale Ebenenschätzung im Disparitätsraum anstatt in Weltkoordinaten durchgeführt wird. Wie in [58] beschrieben, kann die Verteilung von Disparitätsmessungen als normalverteilt angenommen werden. Die Transformation zwischen Disparität- und Weltkoordinaten ist nicht-linear. Daraus resultiert eine Tiefenverteilung der unsicherheitsbehafteten Beobachtungen in Weltkoordinaten, die nicht normalverteilt ist.

Für die weitere Verwendung wird die Funktion, die im Disparitätsraum einen beliebigen Oberflächenpunkt $\mathbf{p} = (u, v, d)^\top$ mit dem dazugehörigen Normalenvektor $\mathbf{n}_{\text{disp}} = (n_u, n_v, n_d)^\top$ in einen Normalenvektor $\mathbf{n}_{\text{world}}$ in Weltkoordinaten transformiert als $f : \mathbb{R}^3 \times \mathbb{R}^3 \rightarrow \mathbb{R}^3$ bezeichnet.

$$\mathbf{n}_{\text{world}} = f(\mathbf{p}, \mathbf{n}_{\text{disp}}) = \begin{pmatrix} Fn_u \\ Fn_v \\ (c_u - u)n_u + (c_v - v)n_v - dn_d \end{pmatrix}. \quad (2.11)$$

Umgekehrt gilt für einen beliebigen Oberflächenpunkt $\mathbf{x} = (x, y, z)^\top$ mit dem dazugehörigen Normalenvektor $\mathbf{n}_{\text{world}} = (n_x, n_y, n_z)^\top$ in Weltkoordinaten die Funktion g , um den Normalenvektor \mathbf{n}_{disp} im Disparitätsraum zu bestimmen:

$$\mathbf{n}_{\text{disp}} = g(\mathbf{x}, \mathbf{n}_{\text{world}}) = \begin{pmatrix} Bn_x \\ Bn_y \\ -xn_x - yn_y - zn_z \end{pmatrix} \quad (2.12)$$

Details zur Transformation von Normalenvektoren zwischen dem Disparitätsraum und Weltkoordinaten können Anhang A.1 entnommen werden.

2.3 Metrologie

Metrologie (von griechisch: $\mu\epsilon\tau\rho\omega$ = messen) bezeichnet die Wissenschaft des Messens. Um die Qualität einer Messung beurteilen zu können, werden Messunsicherheiten verwendet. Eine Messunsicherheit ist laut DIN V ENV 13005 als dem Meßergebnis zugeordneter Parameter definiert, der die Streuung der Werte kennzeichnet, der vernünftigerweise der Meßgröße zugeordnet werden könnte. Einen wesentlichen Anteil an der Messunsicherheit von Signalen hat das Rauschen, welches im folgenden Abschnitt näher definiert wird.

2.3.1 Rauschen

Als Rauschen wird in der Physik eine Störgröße mit breitem unspezifischen Frequenzspektrum bezeichnet. Rauschen kann also als eine Überlagerung vieler harmonischer Schwingungen mit unterschiedlichen Amplituden und Frequenzen ausgedrückt werden. In technischen Systemen tritt Rauschen bei der Signalübertragung auf und kann überall verursacht werden, vom Sender, Empfänger und dem Übertragungsweg. Die Qualität eines Signals wird über das Signal-Rausch-Verhältnis bestimmt. Neben den physikalischen Ursachen kann Rauschen auch nach messtechnischen Größen oder Parametern stochastischer Prozesse klassifiziert werden. Die spektrale Leistungsdichte (Leistung pro infinitesimal kleiner Bandbreite) ist eine messtechnische Größe, mit der man Rauschprozesse weiter klassifizieren kann. Rauschprozesse mit konstanter spektraler Rauschleistungsdichte bezeichnet man als Weißes Rauschen. Rauschprozesse, deren spektrale Rauschleistungsdichte in einem bestimmten Frequenzband konstant sind, werden ebenfalls als Weißes Rauschen oder quasi-weißes Rauschen bezeichnet, da in der Realität keine Rauschprozesse mit unendlich großer Leistung existieren können. Farbige Rauschen bezeichnet Rauschprozesse, deren spektrale Leistungsdichte keinen konstanten Wert aufweist, wie z.B. wenn die Rauschleistungsdichte umgekehrt proportional zur Frequenz ($1/f$ -Rauschen, bzw. Rosa Rauschen) oder dem Quadrat der Frequenz ($1/f^2$ -Rauschen, bzw. Rotes Rauschen) abfällt.

Bildrauschen [66] bei Digitalkameras ist eine zufällige Variation der Helligkeits- oder Farbinformation der Pixel in Bildern. Ursachen für das Bildrauschen ist hauptsächlich das *Dunkelrauschen*, das ohne Lichteinfall auf den Sensor entsteht und auf den Dunkelstrom der einzelnen Pixel und das Rauschen des Ausleseverstärkers zurückzuführen ist. Ungenauigkeiten in der Fertigung und Defekte im Bildsensor können das Dunkelrauschen einzelner Pixel verstärken. Zudem haben die Größe der einzelnen Pixel und der Abstand der Pixel auf dem Sensor einen direkten Einfluß auf das Bildrauschen. Je kleiner die einzelnen Pixel

sind und je geringer der Abstand zwischen den Pixeln ist, desto größer ist das Bildrauschen pro Pixel. Auch bei steigender Temperatur des Sensors nimmt das Bildrauschen zu, was beispielsweise bei Langzeitbelichtungen auftritt. Einen geringeren Einfluß haben das *Schrot- und Quantisierungsrauschen*. Das Schrotrauschen hängt von der zufälligen Anzahl der Photonen ab, die von jedem Pixel detektiert werden. Quantisierungsrauschen tritt bei digitalen Kameras auf, wenn die kontinuierlichen Helligkeitswerte des Bildsensors in diskrete Werte umgewandelt werden. Neben der Sensorgröße beeinflusst die ISO-Empfindlichkeit, die direkt die Verstärkung des Sensorsignals beeinflusst, sowie die analoge Signalverarbeitung und Analog-Digital-Wandlung das Bildrauschen und damit die Qualität des Digitalbilds.

2.3.2 Verteilungsdichtefunktion

In der Messtechnik beschreibt eine kontinuierliche Verteilungsdichtefunktion (auch: Dichtefunktion) eine Wahrscheinlichkeitsverteilung. Das Integral der Dichtefunktion $p_X(x)$ der Zufallsvariablen X in einem bestimmten Intervall $[a, b]$ entspricht der Wahrscheinlichkeit $P_X([a, b])$, dass X einen Wert x in diesem Intervall annimmt.

$$P_X([a, b]) = \int_a^b p_X(x) dx \quad (2.13)$$

Die Verteilungsfunktion $P_X(x)$ beschreibt die Wahrscheinlichkeit, dass ein Wert kleiner gleich x auftritt. Daraus folgt, dass sich die Dichtefunktion $p_X(x)$ als Ableitung der Verteilungsfunktion $P_X(x)$ darstellen lässt [60].

$$p_X(x) = \frac{d}{dx} P_X(x) \quad (2.14)$$

Definitionsgemäß sind die Werte der Dichtefunktion $p_X(x)$ nicht negativ

$$p_X(x) \geq 0 \quad (2.15)$$

und die Fläche der Dichtefunktion ist normiert.

$$\int_{-\infty}^{+\infty} p_X(x) dx = 1 \quad (2.16)$$

Ebenso ist die Verteilungsfunktion $P_X(x)$ als monoton steigend und stetig im Wertebereich $P : \mathbb{R} \rightarrow [0, 1]$ definiert.

Soll aus einem erstellten Histogramm $h(x)$ mit n Bins der Binbreite b_h eine Verteilungsdichte $p_X(x)$ angenähert werden, kann dies durch die Normalisierung des Histogramms durch die Histogrammgesamtfläche $A(h(x))$ erreicht werden:

$$p_X(x) = \frac{h(x)}{A(h(x))}, \text{ mit } A(h(x)) = \sum_{i=0}^n b_h h_i(x) \quad (2.17)$$

Beliebige Verteilungsdichtefunktionen lassen sich anhand der Momente n -ter Ordnung der Zufallsvariablen X beschreiben, die den Erwartungswerten $E\{X^n\}$ entsprechen. Der arithmetische Mittelwert $\mu_X = E\{X\}$ ist das Moment erster Ordnung, das Moment zweiter Ordnung entspricht der Leistung $P = E\{X^2\}$. Die Varianz $\sigma_X^2 = E\{(X - \mu_X)^2\}$ lässt sich als mittelwertfreies Moment zweiter Ordnung beschreiben, aus welchem sich wiederum die Standardabweichung $\sigma_X = \sqrt{\sigma_X^2}$ direkt ableitet.

$$\begin{aligned} \mu_X &= E\{X\} = \frac{1}{n} \sum_{i=1}^n x_i \\ \sigma_X^2 &= E\{(X - \mu_X)^2\} = \frac{1}{n} \sum_{i=1}^n (x_i - \mu_X)^2 \end{aligned} \quad (2.18)$$

Die Schiefe $s(X)$ der Zufallsvariablen X ist das zentrale Moment 3. Ordnung normiert auf die Standardabweichung σ_X und beschreibt die Asymmetrie der Verteilung. Unterschieden wird zwischen negativer Schiefe ($s(X) < 0$, linkschiefe oder rechtssteile Verteilung) und positiver Schiefe ($s(X) > 0$, rechtschiefe oder linkssteile Verteilung). Eine bekannte rechtsschiefe Verteilung ist die Pareto-Verteilung, siehe Bild 2.5a.

Die Schiefe kann aus der empirischen Häufigkeitsverteilung berechnet werden:

$$s = \frac{1}{n} \sum_{i=1}^n \left(\frac{x_i - \mu}{\sigma} \right)^3 \quad (2.19)$$

Die Wölbung w (auch: Kurtosis) ist ein Maß für die Steilheit einer Verteilungsfunktion und entspricht dem 4. zentralen Moment. Eine geringe Wölbung beschreibt eine gleichmäßige Verteilung. Die Wölbung kann empirisch ermittelt werden:

$$w = \frac{1}{n} \sum_{i=1}^n \left(\frac{x_i - \mu}{\sigma} \right)^4 \quad (2.20)$$

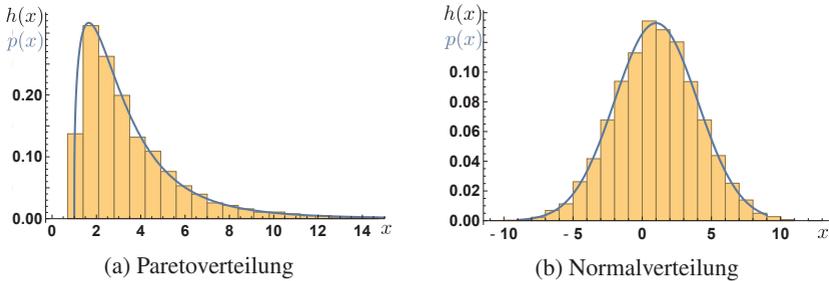


Bild 2.5: Darstellung der Pareto- und Normalverteilung als normiertes Histogramm $h(x)$ (gelb) und Dichtefunktion $p(x)$

Eine häufig auftretende Verteilungsdichtefunktion ist die Gaußverteilung $p_G(x)$ (auch: Normalverteilung), siehe Bild 2.5b. Die Normalverteilung lässt sich direkt mit dem Mittelwert μ und der Standardabweichung σ vollständig beschreiben:

$$p_G(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \quad (2.21)$$

Einen Spezialfall stellt die Standardnormalverteilung $\varphi(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}$ dar, für die der Mittelwert $\mu = 0$ und die Varianz $\sigma^2 = 1$ gilt.

Die Breite der Normalverteilung wird durch die Standardabweichung σ beschrieben. Allgemein gilt für Dichtefunktionen, dass im σ -Intervall K relativ zum Mittelwert μ betrachtet P % aller Beobachtungen liegen. Umgekehrt gilt, dass P % aller Messwerte die Abweichung K vom Mittelwert μ besitzen. Erwartet man eine Normalverteilung, dann gilt der Zusammenhang aus Tabelle 2.1. Z.B. entsprechen sich das 95%-Konfidenzintervall γ und $K = 1,960\sigma$.

K	$\pm\sigma$	$\pm 2\sigma$	$\pm 3\sigma$	$0,675\sigma$	$1,645\sigma$	$1,960\sigma$
P [%]	68, 27	95, 45	99, 73	50	90	95

Tabelle 2.1: Prozentuale Abweichungen P für verschiedene σ -Intervalle K einer Normalverteilung

In der Statistik werden Verteilungstests verwendet, um zu prüfen, ob die Zielverteilung einer bekannten Verteilung entspricht.

Ob die untersuchte Zielverteilung einer Normalverteilung entspricht, kann durch den Jarque-Bera-Test ermittelt werden. Der Jarque-Bera-Test erzeugt aus der Schiefe s , der Wölbung w und der Anzahl der Beobachtungen n einer be-

liebigen Zielverteilung ein Maß, welches die Ähnlichkeit zur Normalverteilung ausdrückt.

$$D_{JB} = \frac{n}{6} \left(s^2 + \frac{(w-3)^2}{4} \right) \quad (2.22)$$

Wenn die Zielverteilung normalverteilt ist, entspricht die resultierende Teststatistik D_{JB} einer asymptotischen Chi-Quadrat Verteilung mit zwei Freiheitsgraden ($D_{JB} \propto \chi_2^2$). Durch die Vorgabe eines Signifikanzniveaus α kann ein Schwellwert für D_{JB} ermittelt werden, der entscheidet, ob die Zielverteilung einer Normalverteilung entspricht.

Um zwei beliebige Verteilungsdichtefunktionen $p_X(x)$ und $p_Y(y)$ zu vergleichen, können verschiedene Vergleichstests verwendet werden. In diesem Abschnitt werden Vergleichstests betrachtet, die diskrete Histogrammverteilungen h_X und h_Y verwenden. Um Instabilitäten vorzubeugen, wird eine Histogrammkorrektur vorgenommen, indem allen Bins der Wert 1 hinzugefügt wird. Der normierte Wert v_i des i -ten Bins des Histogramms h mit n_b Bins der Binbreite 1 ergibt sich zu:

$$v_i = \frac{h_i + 1}{n_b + \sum_{j=1}^{n_b} h_j} \quad (2.23)$$

Im Folgenden wird eine Auswahl von verbreiteten Vergleichstests vorgestellt, welche alle normierte und korrigierte Histogrammverteilungen voraussetzen. Für alle Vergleichstests wird ein Skalar berechnet, der die Ähnlichkeit der zwei Verteilungen h_X und h_Y ausdrückt:

- Kullback-Leibler-Divergenz

$$D_{KL}(h_X, h_Y) = \sum_{i=1}^{n_b} v_{X,i} \log\left(\frac{v_{X,i}}{v_{Y,i}}\right) \quad (2.24)$$

- Jensen-Shannon-Divergenz

$$D_{JS}(h_X, h_Y) = \frac{D_{KL}(h_X, m) + D_{KL}(h_Y, m)}{2} \quad (2.25)$$

Die Jensen-Shannon-Divergenz verwendet $m = \frac{1}{2}(h_X + h_Y)$. Je kleiner das Skalar des jeweiligen Vergleichstests ist, desto wahrscheinlicher liegen für die Zufallsvariablen X und Y die gleichen Verteilungen vor.

Die Kullback-Leibler-Divergenz (auch: Information Gain oder relative Entropie) ist ein nicht symmetrisches Maß, da im Allgemeinen gilt:

$$D_{KL}(h_X, h_Y) \neq D_{KL}(h_Y, h_X) \quad (2.26)$$

Um ein symmetrisches Divergenzmaß zu erhalten, wird die Jensen-Shannon-Divergenz D_{JS} gewählt. Wenn sich die Verteilungen von X und Y gleichen, gilt $D_{JS} = 0$. Bei $X \neq Y$ nimmt D_{JS} positive Werte an, die bei wachsender Divergenz von X und Y größer werden.

2.3.3 Fortpflanzung von Messunsicherheiten

Das Rauschen von real existierenden Messgrößen resultiert in ungenauen Abbildungen der idealen Messgrößen. Nicht-lineare Systeme treten in der Realität oft auf. Sowohl das Systemrauschen als auch nicht-lineare Abbildungsfunktionen machen die Abschätzung der zu erwartenden Ausgangssignale zu einer Herausforderung. Hierfür bieten sich zwei Lösungsansätze an. Zum einen kann eine Linearisierung der Systemgleichung an einem Arbeitspunkt verwendet werden, zum anderen kann eine Monte-Carlo-Simulation zur Hilfe genommen werden.

Im Allgemeinen stellt die Fortpflanzung von Messunsicherheiten eine Möglichkeit für die Approximation der zu erwartenden Wahrscheinlichkeitsdichte des Ausgangssignals dar. In der Messtechnik wird die Messunsicherheit e im Allgemeinen als die Differenz zwischen der unsicherheitsbehafteten Messgröße \hat{x} und der wahren Größe x bezeichnet: $e = \hat{x} - x$.

2.3.3.1 Linearisierung

Die Linearisierung bietet eine Abschätzung des zu erwartenden Systemverhaltens an einem bestimmten Arbeitspunkt eines nicht-linearen Systems an [59]. Der Arbeitspunkt wird zumeist so gewählt, dass er in einer Ruhelage liegt.

Die Taylorreihenentwicklung approximiert das Systemverhalten $y(x)$ einer unsicherheitsbehafteten Variablen \hat{x} am Arbeitspunkt x_0 , indem die Abbildungsfunktion partiell abgeleitet wird:

$$y(x) = y(x_0) + \sum_{i=1}^n \frac{d^i y}{dx^i}(x_0)(x - x_0)^i \quad (2.27)$$

Die Taylorreihenentwicklung berücksichtigt meist nur den Term der ersten Ableitung, wodurch sich das linearisierte System mit $\Delta x = x - x_0$ und $\Delta y = y - y(x_0)$ approximieren lässt.

$$\Delta y \approx \frac{dy}{dx}(x_0)\Delta x \quad (2.28)$$

Die Gauß'sche Fehlerfortpflanzung wird bei einer nicht-linearen Abbildung einer Eingangsfunktion mit mehreren unkorrelierten, unsicherheitsbehafteten Variablen \hat{q}_i der Zufallsvektoren Q_i mit $i \in \mathbb{N}^+$ verwendet. Die Ausgangsfunktion ergibt sich durch das Superpositionsprinzip:

$$\Delta y \approx \frac{dy}{dq_1}(x_0)\Delta q_1 + \frac{dy}{dq_2}(x_0)\Delta q_2 + \dots \quad (2.29)$$

Der erwartete Mittelwert $\hat{\mu}_y$ von Δy ergibt sich durch Einsetzen aller Mittelwerte μ_i der Zufallsvektoren Q_i zu

$$\hat{\mu}_y = y(\hat{\mu}_1, \hat{\mu}_2, \dots) \quad (2.30)$$

Die erwartete Standardabweichung $\hat{\sigma}_y$ entspricht

$$\hat{\sigma}_y = \sqrt{\sum_{i=0}^N \left(\frac{dy}{dq_i} \sigma_{q_i} \right)^2} \quad (2.31)$$

Sind die Verteilungen Q_i der unsicherheitsbehafteten Variablen \hat{q}_i normalverteilt, kann die Ausgangsfunktion auch durch eine Normalverteilung $y(\hat{q}) = \mathcal{N}(\hat{\mu}_y, \hat{\sigma}_y)$ angenähert werden.

2.3.3.2 Monte-Carlo-Simulation

Die Monte-Carlo-Simulation ist ein Verfahren aus der Stochastik, um mathematische Problemstellungen numerisch zu lösen. Bei einer Monte-Carlo-Simulation wird durch eine große Anzahl von gleichartig durchgeführten Zufallsexperimenten eine Lösung eines mathematischen Problems approximiert, das analytisch schwer oder gar nicht lösbar ist. In rechnergestützten Monte-Carlo-Simulationen werden Zufallszahlen für die einzelnen Experimente generiert. Dadurch können sehr komplexe Systeme simulativ untersucht werden. Dieses ist vor allem durch die enorm hohe Rechenleistung von heutigen Computersystemen möglich. Da die Zufallsexperimente einer Monte-Carlo-Simulation unabhängig voneinander ausgeführt werden können, können sie auf Mehrkern-Prozessorchips und Graphikkarten parallelisiert werden.

Die Fortpflanzung von Messunsicherheiten von nicht-linearen Abbildungen kann mithilfe einer Monte-Carlo Simulation untersucht werden. Monte-Carlo Methoden sind Inferenzmethoden, die auf numerischem Sampling basieren. Hierbei wird eine Zufallsvariable X mit der Verteilungsdichte p_X einer unsicherheitsbehafteten Beobachtung betrachtet. Ziel ist es, die Verteilungsdichte zu bestimm-

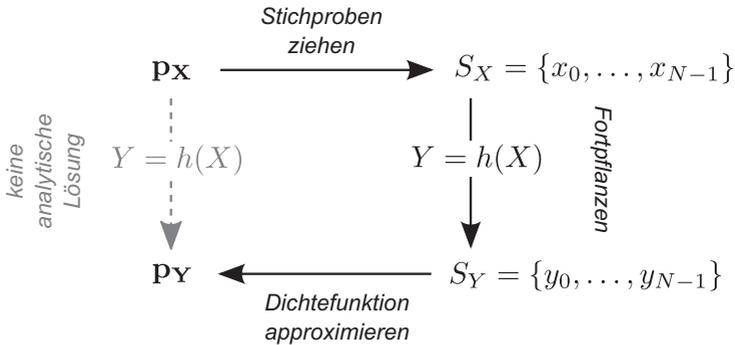


Bild 2.6: Abschätzen einer Zielverteilung p_Y aus einer Startverteilung p_X mithilfe einer Monte-Carlo-Simulation

men, die durch Fortpflanzung der Beobachtungen einer beliebigen Funktion h bestimmt wird und durch die Zufallsvariable $Y = h(X)$ gegeben ist, siehe Bild 2.6. Zunächst werden N Proben $S_X = \{x_0, \dots, x_{N-1}\}$ aus der Verteilung X gezogen, $x_k \sim p_X$. Durch das Fortpflanzen der Proben mittels h wird die Probenmenge $S_Y = \{y_0, \dots, y_{N-1}\}$ mit $y_k = h(x_k)$ bestimmt. Die Probenmenge S_Y schätzt die Verteilung von Y ab. Aus der resultierenden Verteilungsdichte p_Y lassen sich statistische Messgrößen wie der Mittelwert (Mean) μ_Y , die Standardabweichung σ_Y oder das 95%-Konfidenzintervall γ_Y bestimmen.

2.4 Optimalfilter

In der Signalverarbeitung bezeichnet das Optimalfilter (*engl.*: matched filter) ein Filter, welches die Lage einer bekannten Signalform (*engl.*: template) in einem unbekanntem Nutzsignal optimal detektiert, welches additives, stochastisches Rauschen aufweist. Das Optimalfilter ist eine Demodulationstechnik für lineare, zeitinvariante Filter, um das Signal-Rausch-Verhältnis zu maximieren.

Die Korrelation zweier zeit- und wertekontinuierlicher reeller Signale $x(t)$ und $y(t)$ mit der Zeitverschiebung τ zwischen den Signalen wird durch die Kreuzkorrelationsfunktion

$$R_{xy}(\tau) = \int_{-\infty}^{\infty} x^*(t)y(t + \tau)dt \quad (2.32)$$

beschrieben, $x^*(t)$ stellt die komplex konjugierte von $x(t)$ dar. Die zeitdiskrete Darstellung bzgl. des Laufparameters k und der Verschiebung l wird wie folgt dargestellt:

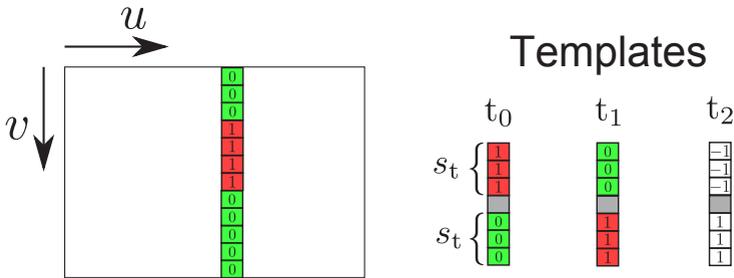


Bild 2.7: Links: Beispielwerte einer Bildspalte. Rechts: Muster t_i die zur Kantenfilterung der Bildspalte verwendet werden.

$$R_{xy}(l) = \sum_{k=-\infty}^{\infty} x^*(k)y(k+l) \tag{2.33}$$

In der Bildverarbeitung werden Korrelationsfilter verwendet, um bekannte Muster wiederzufinden. Die Zeitabhängigkeit der klassischen Signalverarbeitung wird dabei durch eine weitere Ortsvariable ersetzt. Dies ermöglicht die Detektion bekannter Muster in Bildern mit den Bildkoordinaten u und v .

Ein Beispiel für ein Optimalfilter ist in Bild 2.7 dargestellt. Die Muster t_0 und t_1 werden verwendet, um die entsprechenden Kanten in einer Bildspalte zu finden. Das Ergebnis des Optimalfilters sind die Kostenfunktionen C_0 und C_1 . Das Minimum der jeweiligen Kostenfunktion entspricht der Bildzeile, in der das Muster am besten mit dem Werteverlauf im Bild übereinstimmt.

Die Korrelationskosten werden normiert, indem mithilfe der Größe des Musters t_i der Normierungsfaktor des Templates $n_{t,i} = \sum_{l=-s_t}^{s_t} |t_i(l, 1)|$ berücksichtigt wird:

$$C_i(u, v) = \frac{1}{n_{t,i}} \sum_{l=-s_t}^{s_t} I(u+l, v)t_i(l, 1) \quad , \quad i \in \{0, 1, 2\} \tag{2.34}$$

Die Korrelationskosten C_0 und C_1 können auch bestimmt werden, wenn nur eine Korrelationsoperation ausgeführt wird. Dies wird in dem Beispiel erreicht, indem der Wertebereich des Ausgangsbilds I auf -1 bis 1 skaliert wird und die Templatemaske t_2 gewählt wird. Aus den Korrelationskosten C_2 können die Kosten C_0 und C_1 abgeleitet werden:

$$C_0(u, v) = 1 - \max(C_2(u, v), 0) \tag{2.35}$$

$$C_1(u, v) = 1 + \min(C_2(u, v), 0) \quad (2.36)$$

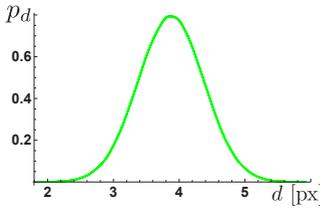
Durch die effizientere Berechnung der Kostenbilder kann der Rechenaufwand des Arbeitsschritts nahezu halbiert werden.

3 Genauigkeitsuntersuchung von binokularen Normalenvektoren

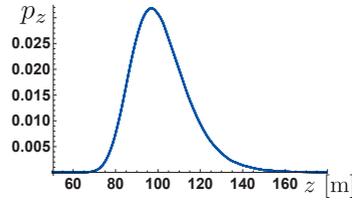
In der Umwelterfassung stellt das Schätzen von Normalenvektoren eine wichtige Aufgabe dar, z.B. für die Rekonstruktion von Oberflächen, zur Registrierung von Punktwolken oder für die Objektdetektion. In den letzten Jahren bieten viele Sensoren die Möglichkeit zur Erfassung von Tiefeninformationen, wie Lidare oder RGB-D Kameras. Binokulare Kamerasysteme bieten hochaufgelöste, dichte Tiefeninformationen in Echtzeit in nahen und mittelweit entfernten Szenen und lassen sich zudem in verschiedenen Umgebungen innerhalb und außerhalb von Gebäuden einsetzen.

Jedoch hat das Systemrauschen von binokularen Kamerasystemen die Eigenschaft, dass die Unsicherheit der Tiefeninformation in Weltkoordinaten mit dem Abstand zum Kamerasystem quadratisch anwächst. Diese Eigenschaft kann direkt aus der Jacobimatrix abgelesen werden (siehe Gleichung 2.10) und wird zudem in Bild 3.1c und 3.1d deutlich. Die Bilder zeigen die Verteilungsfunktion der Standardabweichung in Tiefenrichtung im Disparitätsraum und in Weltkoordinaten. Während die Standardabweichung der gemessenen Position im Disparitätsraum für alle Tiefen annähernd konstant bleibt, wächst die Standardabweichung in Weltkoordinaten quadratisch mit zunehmendem Abstand. Zusätzlich ist die Abbildungsfunktion zwischen Disparitätskoordinaten und Weltkoordinaten nicht linear. Dies führt bei der Annahme von normalverteilten Tiefenrauschen im Disparitätsraum [29] zu einer schiefen Verteilung des Tiefenrauschens in Weltkoordinaten (siehe Bild 3.1a und 3.1b). Diese Eigenschaften führen dazu, dass das Schätzen von Normalenvektoren aus binokularen Kamerasystemen (im Folgenden: *binokulare Normalenvektoren*) einer bestimmten Genauigkeitsgüte eine besondere Herausforderung darstellt.

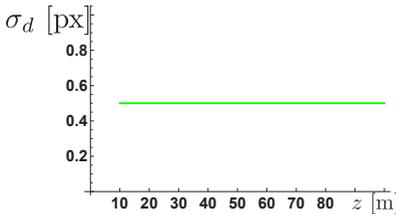
In diesem Kapitel wird zunächst das Verhalten von binokularen Normalenvektoren simulativ untersucht (Kapitel 3.2). Dazu wird der komplette Rekonstruktionsprozess der binokularen Normalenvektoren betrachtet, vom Schätzen des Normalenvektors aus der Umgebung eines Tiefenpunktes im Disparitätsraum bis hin zur Rekonstruktion des Normalenvektors in Weltkoordinaten mit der Funktion f aus Gleichung 2.11. Dabei wird die Orientierungsempfindlichkeit der geschätzten Normalenvektoren zunächst durch eine Monte-Carlo-Methode



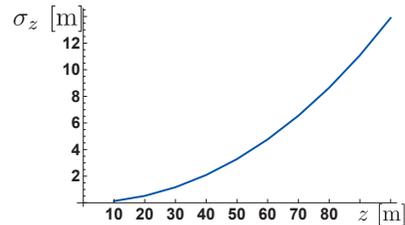
(a) **Tiefenrauschen im Disparitätsraum:** Normalverteilte Dichtefunktion



(b) **Tiefenrauschen in Weltkoordinaten:** Schiefe Verteilungsdichte



(c) **Standardabweichung im Disparitätsraum** über Distanz.



(d) **Standardabweichung in Weltkoordinaten** über Distanz.

Bild 3.1: Annahme zur Simulation des Tiefenrauschens: Verteilungsdichten und Standardabweichungen des Tiefenrauschens im Disparitätsraum (grün) und in Weltkoordinaten (blau)

bestimmt und mit einer Linearisierung durch Gauß'sche Fehlerfortpflanzung approximiert und validiert. Die Erkenntnisse aus dem untersuchten Verhalten werden in Kapitel 3.3 verwendet, um eine Orientierungsgenauigkeit von binokularen Normalenvektoren abschätzen zu können. Die Ergebnisse aus der Simulation werden abschließend mit realen Experimenten validiert.

3.1 Vorarbeiten

Das Schätzen von Oberflächennormalenvektoren aus Tiefendaten [5] wird in vielen Anwendungen genutzt, wie beispielsweise bei der Oberflächenrekonstruktion [35, 30, 3], der Registrierung von Punktwolken [52] und der Objektdetektion [28]. Zudem werden Oberflächenorientierungen in Fahrerassistenzfunktionen verwendet, um zwischen dem Freibereich vor dem Fahrzeug und begrenzenden Objekten unterscheiden zu können [38]. Das Verwenden von binokularen Normalenvektoren ist besonders herausfordernd, da die Tiefenunsicherheit der rekonstruierten Punkte in Weltkoordinaten quadratisch mit zunehmender Distanz anwächst.

Existierende Arbeiten mit binokularen Kamerasystemen verwenden zur Abschätzung des Tiefenfehlers eine Taylorreihenentwicklung erster Ordnung [14] [39]. Der Einfluss unsicherheitsbehafteter Sensorparameter auf den Quantisierungsfehler des binokularen Kamerasystems wird in [34] untersucht. Ein Ansatz zur Untersuchung der 3d- Oberflächenorientierung und deren Unstetigkeiten wurde in [68] vorgestellt. Wolff et al. [69] untersuchen die Fortpflanzung von Unsicherheiten, um Linien und ebene Oberflächenorientierungen zu schätzen. Dabei werden Methoden verglichen, die auf Linienmerkmalen und der absoluten Korrespondenz der Punkte beruhen.

In dieser Arbeit wird die erwartete Winkelfehlerverteilung durch die Fortpflanzung von Messunsicherheiten analysiert. Der Hauptbeitrag dieses Kapitels ist die Bewertung der Winkelfehlerverteilungen binokularer Normalenvektoren bezüglich der Bildposition, der Entfernung und der Orientierung. Darüber hinaus wird die Transformation der Normalenvektoren zwischen Disparitätsraum und Weltkoordinaten hergeleitet. Es werden zwei Methoden entwickelt, die es mithilfe der analysierten Eigenschaften ermöglichen, einen Normalenvektor mit einer geforderten Orientierungsgenauigkeit zu bestimmen. Dafür wird für den Oberflächenpunkt des Normalenvektors eine quadratische Nachbarschaft in Bildkoordinaten ermittelt. Die Richtungsabweichung des Normalenvektors wird über ein Vertrauensintervall definiert.

3.2 Simulationen

In diesem Kapitel wird die Genauigkeit der Richtungsschätzung binokularer Normalenvektoren aus Disparitätsbildern untersucht. Der Messfehler der Disparitätswerte aller Bildpositionen (u, v) wird als unkorreliert angenommen. Das Disparitätsrauschen wird als mittelwertefreie Normalverteilung $\mathcal{N}(0, \sigma_d)$ modelliert, welches eine konstante Standardabweichung σ_d besitzt. Die Standardabweichung σ_d ist insbesondere unabhängig von der Entfernung der Kamera zum betrachteten Oberflächenpunkt. Die Bildkoordinaten (u, v) werden als exakt bekannt angenommen und sind nicht Bestandteil der betrachteten Fortpflanzung der Messunsicherheiten.

Ziel dieses Kapitels ist eine Abschätzung der Richtungsgenauigkeit eines Normalenvektors, welcher direkt aus dem Disparitätsbild bestimmt werden soll. Die zur Berechnung des Normalenvektors verwendeten Punkte sollen direkt aus einer quadratischen Bildnachbarschaft (dem sogenannten Patch) des Oberflächenpunkts \mathbf{p}_0 bestimmt werden. Diese Punkte werden im Folgenden als Patchpunkte bezeichnet.

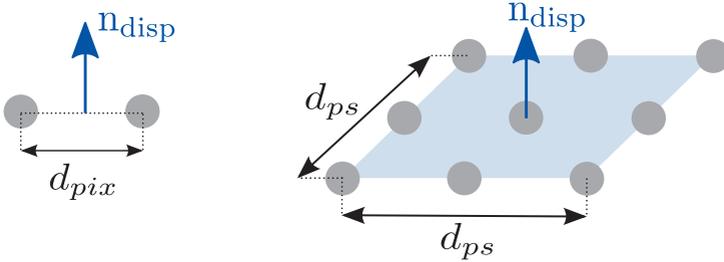


Bild 3.2: Patches mit zugehörigen Normalen \mathbf{n}_{disp} , die für 2d- (links) und 3d-Simulationen (rechts) in dieser Arbeit verwendet werden.

Für die Experimente in diesem Kapitel werden die idealen Normalenvektorrichtungen in Weltkoordinaten und die idealen Oberflächenpunkte in Bildkoordinaten (u, v) in einer vorgegebenen Entfernung bestimmt. An dem Oberflächenpunkt $\mathbf{p}_0 = s(\mathbf{x}_0)$ (Gleichung 2.9) werden mithilfe des Normalenvektors die m Patchpunkte $\mathbf{p}_i = (u_i, v_i, d_i)^\top$ bestimmt ($i \in \mathbb{N}^+$). Die Bildkoordinaten (u_i, v_i) der Patchpunkte \mathbf{p}_i werden durch die Abstände d_{pix} bzw. d_{ps} in Pixeln für die 2d/3d-Simulationen bestimmt (siehe Bild 3.2). Die entsprechenden Disparitätswerte d_i der Patchpunkte werden durch die Annahme bestimmt, dass $\mathbf{n}_{\text{world}} \cdot (\mathbf{x}_i - \mathbf{x}_0) = 0$ gilt. Es folgt durch Einsetzen von Gleichung 2.8

$$d_i = \frac{d_0 ((u_i - c_u)n_x + (v_i - c_v)n_y + F n_z)}{((u_0 - c_u)n_x + (v_0 - c_v)n_y + F n_z)}. \quad (3.1)$$

Für alle Simulationen werden realistische Kameraparameter angenommen, die dem weit verbreiteten KITTI-Datensatz [17] entsprechen.

Die im Folgenden durchgeführten Simulationen untersuchen die Korrelation charakteristischer Parameter bei der Variation der Orientierung des Normalenvektors und des Oberflächenpunkts bzgl. der Bildposition und Entfernung zur Kamera.

3.2.1 2d-Simulationen

Zunächst werden Simulationen im zweidimensionalen Raum (2d) betrachtet, bei dem ein Normalenvektor $\mathbf{n}_{\text{world}}$ aus den zwei Punkten $\mathbf{x}_1 = (x_1, 0, z_1)^\top$ und $\mathbf{x}_2 = (x_2, 0, z_2)^\top$ geschätzt wird:

$$\mathbf{n}_{\text{world}} = \begin{pmatrix} n_x \\ n_y \\ n_z \end{pmatrix} = \begin{pmatrix} z_2 - z_1 \\ 0 \\ x_1 - x_2 \end{pmatrix} \quad (3.2)$$

Zur Evaluierung der Richtungsgenauigkeit der unsicherheitsbehafteten Normalenvektoren wird ein Differenzwinkel δ eingeführt. δ beschreibt den Winkel zwischen dem idealen Normalenvektor $\mathbf{n}_{\text{world}} = (n_x, 0, n_z)^\top$ und dem geschätzten Normalenvektor $\hat{\mathbf{n}}_{\text{world}} = (\hat{n}_x, 0, \hat{n}_z)^\top$ aus den unsicherheitsbehafteten Punktmessungen $\hat{\mathbf{x}}_1$ und $\hat{\mathbf{x}}_2$. Für den 2d-Fall wird δ wie folgt berechnet:

$$\delta = f_\delta(\mathbf{n}_{\text{world}}, \hat{\mathbf{n}}_{\text{world}}) = \alpha - \hat{\alpha} = \arctan\left(\frac{n_z}{n_x}\right) - \arctan\left(\frac{\hat{n}_z}{\hat{n}_x}\right) \quad (3.3)$$

Der Winkel $\delta = \alpha - \hat{\alpha}$ beschreibt die Winkeldifferenz zwischen der x-Achse und den Normalenvektoren $\mathbf{n}_{\text{world}}$ und $\hat{\mathbf{n}}_{\text{world}}$. Gleichung 2.8 in 3.2 eingesetzt und weiter in Gleichung 3.3 ergibt

$$\begin{aligned} \delta &= f_\delta(\mathbf{n}_{\text{world}}, \hat{\mathbf{n}}_{\text{world}}) \\ &= \arctan\left(\frac{c_u(d_1 - d_2) - d_1 u_2 + d_2 u_1}{F(d_1 - d_2)}\right) \\ &\quad - \arctan\left(\frac{c_u(\hat{d}_1 - \hat{d}_2) - \hat{d}_1 u_2 + \hat{d}_2 u_1}{F(\hat{d}_1 - \hat{d}_2)}\right) \end{aligned} \quad (3.4)$$

Es gilt $\delta \in [-90^\circ, 90^\circ]$, da bei den 2d-Experimenten die Richtung des Winkels δ berücksichtigt wird.

Zunächst wird durch 2d-Experimente simulativ untersucht, ob mit einer analytischen Lösung durch Linearisierung eine vergleichbare Genauigkeit zur Lösung der Monte-Carlo-Simulation erreicht wird.

3.2.1.1 Linearisierung

Eine analytische Lösung der nichtlinearen Funktion f wird durch eine Gauß'sche Fehlerfortpflanzung beschrieben, wie in Kapitel 2.3.3.1 eingeführt. Der Messfehler der unsicherheitsbehafteten Disparitäten \hat{d}_1 und \hat{d}_2 wird als unkorreliert angenommen. Die resultierende Verteilungsdichte p_δ^{LIN} der Gauß'schen Fehlerfortpflanzung entspricht einer Normalverteilung $\mathcal{N}_\delta(\mu_\delta^{LIN}, \sigma_\delta^{LIN})$, welche im Folgenden für das betrachtete Beispiel der 2d-Simulation hergeleitet wird.

Für den Mittelwert μ_δ^{LIN} gilt bei der Gauß'schen Fehlerfortpflanzung des angeführten Beispiels

$$\mu_\delta^{LIN} = 0. \quad (3.5)$$

Die mittelwertfreie Ausgangsverteilung p_δ^{LIN} resultiert daraus, dass die Mittelwerte den idealen Disparitätswerten entsprechen:

$$\mu_{\hat{d}_1} = d_1, \quad \mu_{\hat{d}_2} = d_2 \quad (3.6)$$

Dadurch gilt $\hat{\alpha} = \alpha$ in Formel 3.3 und resultiert in $\mu_\delta^{LIN} = 0$. Die Standardabweichung σ_δ^{LIN} der Ausgangsverteilung wird durch

$$\sigma_\delta^{LIN} = \sqrt{\left(\frac{\partial f_\delta}{\partial \hat{d}_1} \sigma_{\hat{d}_1}^2\right)^2 + \left(\frac{\partial f_\delta}{\partial \hat{d}_2} \sigma_{\hat{d}_2}^2\right)^2} = \sigma_{\hat{d}} \sqrt{\left(\frac{\partial f_\delta}{\partial \hat{d}_1}\right)^2 + \left(\frac{\partial f_\delta}{\partial \hat{d}_2}\right)^2} \quad (3.7)$$

bestimmt.

Die partiellen Ableitungen nach den unsicherheitsbehafteten Variablen \hat{d}_1 und \hat{d}_2 ergeben sich zu

$$\frac{\partial f_\delta}{\partial \hat{d}_1} = \frac{\hat{d}_2 F(u_1 - u_2)}{c_u^2 (\hat{d}_1 - \hat{d}_2)^2 - 2c_u (\hat{d}_1 - \hat{d}_2) (\hat{d}_1 u_2 - \hat{d}_2 u_1)} \dots$$

$$\dots \frac{\dots}{+\hat{d}_1^2 (F^2 + u_2^2) - 2\hat{d}_1 \hat{d}_2 (F^2 + u_1 u_2) + \hat{d}_2^2 (F^2 + u_1^2)} \quad (3.8)$$

und

$$\frac{\partial f_\delta}{\partial \hat{d}_2} = -\frac{\hat{d}_1 F(u_1 - u_2)}{c_u^2 (\hat{d}_1 - \hat{d}_2)^2 - 2c_u (\hat{d}_1 - \hat{d}_2) (\hat{d}_1 u_2 - \hat{d}_2 u_1)} \dots$$

$$\dots \frac{\dots}{+\hat{d}_1^2 (F^2 + u_2^2) - 2\hat{d}_1 \hat{d}_2 (F^2 + u_1 u_2) + \hat{d}_2^2 (F^2 + u_1^2)}. \quad (3.9)$$

Die Abschätzung der Winkelverteilung $p_\delta^{LIN} = \mathcal{N}_\delta(\mu_\delta^{LIN}, \sigma_\delta^{LIN})$ ist für beliebige Normalenvektororientierungen und -positionen im Raum möglich. Diese Linearisierung wird in Kapitel 3.2.1.3 wieder aufgegriffen und gegen die numerische Lösung der im Folgenden eingeführten Monte-Carlo-Simulation bewertet.

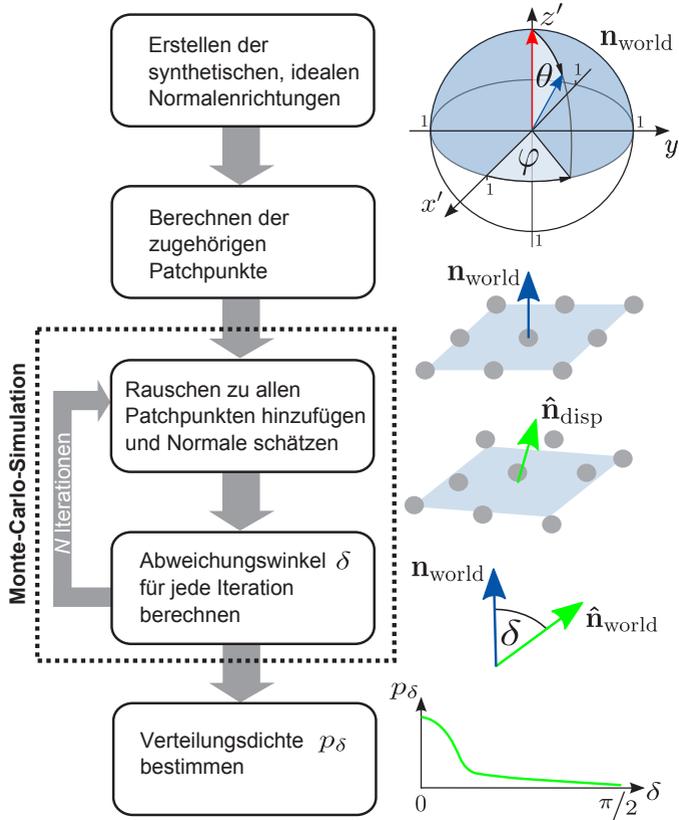


Bild 3.3: Hauptschritte der Monte-Carlo-Simulation zur Fortpflanzung der Unsicherheiten aller Oberflächenpunkte

3.2.1.2 Monte-Carlo-Simulation

Die Monte-Carlo-Simulation wird in diesem Kapitel genutzt, um die Verteilungsdichtefunktion p_δ^{MC} einer nicht-linearen Abbildung zu bestimmen. Im Gegensatz zur Gauß'schen Fehlerfortpflanzung muss die resultierende Dichtefunktion p_x nicht einer Normalverteilung entsprechen, sondern kann eine beliebige Verteilungsform annehmen. Die Präzision der Abbildung von p_x nimmt zu mit steigender Anzahl an Iterationen.

Die Monte-Carlo-Simulation in dieser Arbeit verwendet die in Bild 3.3 gezeigten Hauptschritte. Es werden der ideale Normalenvektor \mathbf{n}_{disp} mit den zugehörigen Patchpunkten $\mathbf{p}_i = (u_i, v_i, d_i)^\top$ bestimmt, wie bereits in Kapitel 3.2 beschrieben.

Während jeder Iteration wird allen idealen m Patchpunkten \mathbf{p}_i Disparitätsrauschen hinzugefügt. Das Disparitätsrauschen wird von einem Quasi-Zufallsgenerator erzeugt, der aus der Normalverteilung $\mathcal{N}(0, \sigma_d)$ mit $\sigma_d = 0, 1$ Pixel für jeden Punkt \mathbf{p}_i unabhängig einen Rauschwert ξ_i zieht. Der unsicherheitsbehaftete Disparitätswert \hat{d}_i jedes Patchpunkts ergibt sich zu $\hat{d}_i = d_i + \xi_i$. Nach jeder Iteration wird $\hat{\mathbf{n}}_{\text{disp}}$ im Disparitätsraum geschätzt und in Weltkoordinaten transformiert: $\hat{\mathbf{n}}_{\text{world}} = f(\bar{\mathbf{p}}, \hat{\mathbf{n}}_{\text{disp}})$ mit $\bar{\mathbf{p}} = (u_0, v_0, \mu_{\hat{d}_i})^\top$. Während jeder Iteration wird der Abweichungswinkel $\delta = f_\delta(\mathbf{n}_{\text{world}}, \hat{\mathbf{n}}_{\text{world}})$ bestimmt (siehe Gleichung 3.4) und in einem Histogramm h_δ vermerkt. Das Histogramm h_δ wird nach der letzten Iteration normalisiert und ergibt die Verteilungsdichte p_δ .

3.2.1.3 Vergleich von Linearisierung und Monte-Carlo-Methode

Im Folgenden werden 2d-Simulationen (Experiment ① bis ④) durchgeführt, die das Verhalten der Richtungsgenauigkeit binokularer Normalenvektoren von den folgenden Parametern zeigen soll:

- der *Entfernung* z [m] zwischen Oberflächenpunkt x_0 und der Kamera
- dem *Abstand der Patchpunkte* d_{pix} [px]
- der *Bildposition* u [px] des Oberflächenpunkts \mathbf{p}_0
- der *Orientierung* θ [°] des Normalenvektors $\mathbf{n}_{\text{world}}$

Für jedes Experiment werden die Parametersätze aus Tabelle 3.1 verwendet. Als Ergebnis werden nach jedem Experiment die Verteilungsdichten p_δ ausgewertet. Für jedes Experiment sind zwei Verteilungsdichten p_δ^{MC} der Monte-

Experiment	z [m]	d_{pix} [px]	u [px]	θ [°]
1	10 20	15	c_u	0
2	50	10 30	c_u	0
3	50	15	0 0, 75 c_u	0
4	50	15	c_u	40 80

Tabelle 3.1: Paramter der 2d-Simulationen ① bis ④

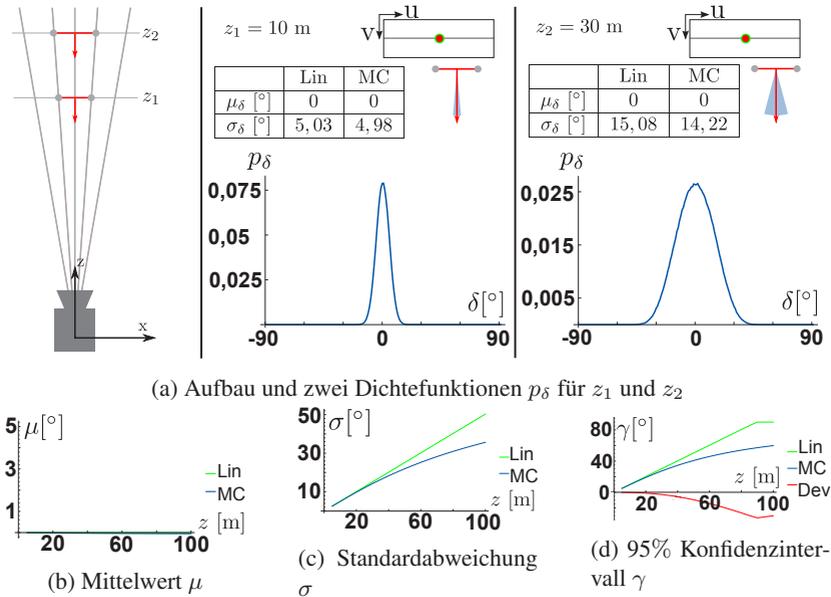


Bild 3.4: Experiment ①: Analyse der Verteilungsdichte p_δ für verschiedene Entfernungen z

Carlo-Simulation visualisiert, die zwei Werte des variierten Parameters exemplarisch darstellen. Zusätzlich werden der Mittelwert μ_δ und die Standardabweichung σ_δ der Linearisierung (Lin) und der Monte-Carlo-Simulation (MC) über den variierten Parameter analysiert. Zum einen wird die Ähnlichkeit der Monte-Carlo-Simulation zur Approximation der Gauß'schen Fehlerfortpflanzung verglichen. Daraus lässt sich schließen, ob die Linearisierung die nicht-lineare Abbildung ausreichend genau beschreiben kann. Ausschlaggebend ist die Form von p_δ der Monte-Carlo-Simulation im Vergleich zur Gauß'schen Normalverteilung der Linearisierung. Zudem wird die Ähnlichkeit der Mittelwerte μ_δ und der Standardabweichungen σ_δ analysiert. Zum anderen wird der Einfluß der in den Simulationen variierten Parameter bezüglich μ_δ und σ_δ analysiert und bewertet. Zusätzlich wird das 95%-Konfidenzintervall γ_δ der Dichtefunktion p_δ bestimmt und die Abweichung (Dev) zwischen Linearisierung und Monte-Carlo-Simulation in Rot dargestellt.

In **Experiment ①** aus Bild 3.4 werden zwei Normalenvektoren in unterschiedlichen Tiefen geschätzt ($d_{pix} = const$, $u = const$ und $\theta = const$). Für alle Entfernungen z entspricht die Bildposition des Oberflächenpunkts exakt dem Bildhauptpunkt. Alle idealen Normalenvektoren \mathbf{n}_{world} sind entlang der Kamera

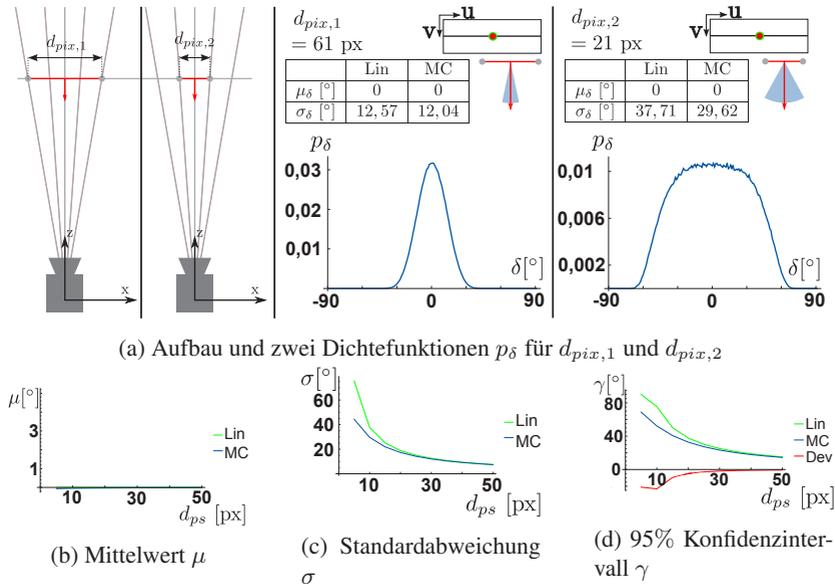


Bild 3.5: Experiment ②: Analyse der Verteilungsdichte p_δ für verschiedene Abstände der Patchpunkte d_{pix}

z -Achse ausgerichtet, die dem Sichtstrahl des Oberflächenpunkts entspricht. Die Patchpunkte aller untersuchten Normalenvektoren haben denselben Abstand d_{pix} in der Bildebene. In Weltkoordinaten wächst der Abstand zwischen den Patchpunkten mit zunehmender Entfernung an.

Die Verläufe von p_δ der Monte-Carlo-Simulation entsprechen bei den Entfernungen z_1 und z_2 mittelwertfreien Normalverteilungen mit unterschiedlichen Standardabweichungen. Die höhere Standardabweichung des weiter entfernten Normalvektors bedeutet, dass sich die Orientierung des Normalenvektors nur mit einer geringeren Genauigkeit schätzen lässt. Diese Eigenschaften lassen sich für alle Entfernungen beim Verlauf der Mittelwerte der Monte-Carlo-Simulation und der Linearisierung validieren, bei dem $\mu_\delta = \mu_f = 0^\circ$ ist. Im Nahbereich entsprechen sich die Standardabweichungen der Linearisierung und Monte-Carlo-Simulation nahezu, während sie mit zunehmender Entfernung z immer stärker divergieren.

Generell nimmt die Orientierungsgenauigkeit bei der Normalenvektorschätzung mit zunehmender Entfernung des Normalenvektors ab. Zudem überschätzt die Linearisierung die Standardabweichung mit zunehmender Entfernung z im-

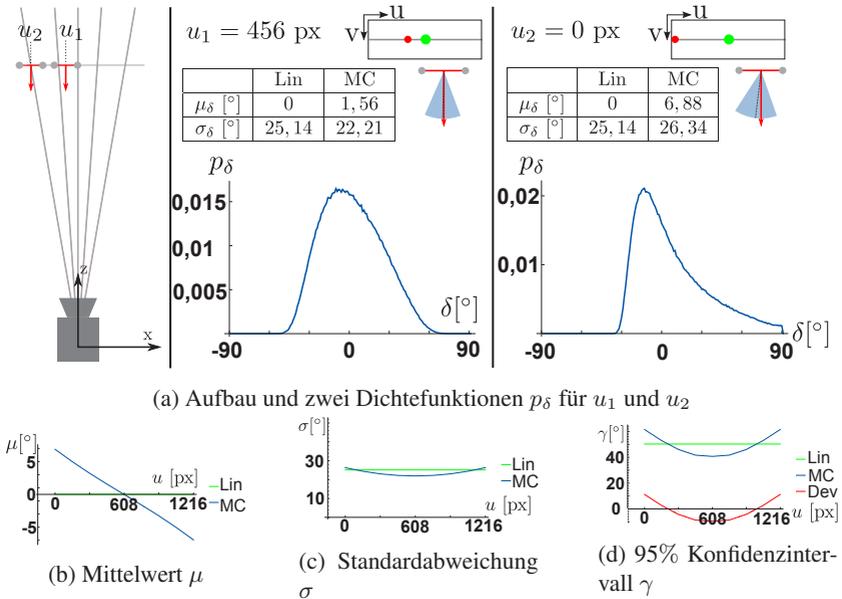


Bild 3.6: Experiment ③: Analyse der Verteilungsdichte p_δ für verschiedene Bildpositionen u

mer stärker, während die Mittelwerte der Linearisierung mit denen der Monte-Carlo-Simulationen übereinstimmen.

Experiment ② untersucht Normalenvektoren, deren Patchpunkte einen unterschiedlichen Abstand d_{pix} in der Bildebene besitzen ($z = const$, $u = const$ und $\theta = const$). Alle untersuchten Normalenvektoren sind auf die Kamera ausgerichtet, während deren Oberflächenpunkte die gleiche Entfernung zur Kamera haben und exakt im Kamerahauptpunkt liegen.

Beide Dichtefunktionen p_δ der Monte-Carlo-Simulation sind mittelwertfrei und entsprechen dahingehend der Linearisierung. Die Orientierung des Normalenvektors mit größerem Patchpunktabstand $d_{pix,1}$ lässt sich besser bestimmen als mit geringerem Patchpunktabstand $d_{pix,2}$. Dieser Effekt bestätigt sich bei der Variation der Patchpunktabstände. Während p_δ für alle beobachteten d_{ps} mittelwertfrei ist, nimmt die Standardabweichung mit zunehmendem Patchpunktabstand ab. Dies entspricht einer Zunahme der Genauigkeit der Orientierungsschätzung.

Die Linearisierung kann den Verlauf der Standardabweichung nur bei großen Patchpunktabständen d_{pix} abbilden. Bei kleiner werdenden d_{pix} divergiert die

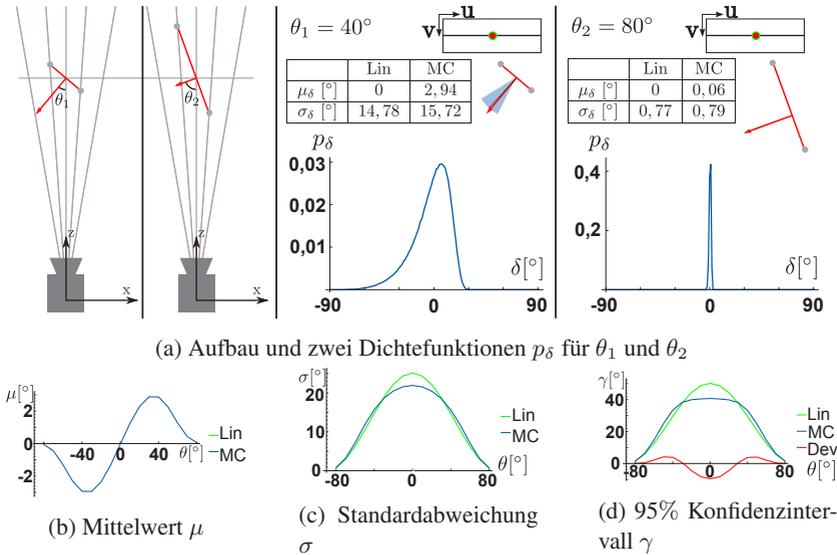


Bild 3.7: Experiment ④: Analyse der Verteilungsdichte p_δ für Normalenvektororientierungen θ

Standardabweichung der Linearisierung immer stärker von der Standardabweichung der Monte-Carlo-Simulation. Dies kann hauptsächlich mit der Verteilungsform von p_δ begründet werden, die mit kleiner werdenden d_{pix} immer stärker von der Form einer Normalverteilung abweicht. Diese Abweichung kann schon beim Verlauf von p_δ mit $d_{pix,2}$ beobachtet werden.

Zusammenfassend steigt die Orientierungsgenauigkeit der Normalenvektorschätzung an einem Oberflächenpunkt, wenn die Abstände der Patchpunkte erhöht wird. Die Dichtefunktionen p_δ der Linearisierung und der Monte-Carlo-Simulation sind mittelwertfrei, wenn d_{pix} variiert wird. Allerdings überschätzt die Linearisierung die Standardabweichung mit abnehmendem Abstand der Patchpunkte d_{pix} .

Experiment ③ schätzt Normalenvektoren an verschiedenen Bildpositionen u ($z = const$, $d_{pix} = const$ und $\theta = const$). Alle Normalenvektoren sind in einer konstanten Entfernung z parallel zur Kamera z-Achse ausgerichtet und haben dieselben Abstände d_{pix} zwischen den Patchpunkten.

Der rote Punkt in der Bildebene u_1 liegt leicht versetzt vom Bildhauptpunkt (siehe grüner Punkt in der Bildebene), während u_2 am Bildrand platziert ist. Die Verteilungsdichte p_δ ist bei u_1 leicht schief und bei u_2 eindeutig als schiefe Verteilung erkennbar. Die schiefen Verteilungen resultieren in Mittelwerten $\mu_\delta \neq 0^\circ$.

Sowohl die Mittelwerte als auch die Standardabweichung nimmt von u_1 zu u_2 zu.

Bei der Auswertung der Verteilungsdichten p_δ über die gesamte Bildbreite lässt sich die Zunahme der Orientierungsunsicherheit der Normalenvektoren zu den Bildrändern validieren. Relativ zum Bildhauptpunkt ergibt sich ein symmetrischer Verlauf der Orientierungsunsicherheit.

Dieses Verhalten von p_δ über die Bildbreite kann die Linearisierung nicht abbilden. Die Linearisierung nimmt einen konstanten Verlauf der Verteilungsdichte p_δ über die Bildbreite an, was zu erheblichen Abweichungen zum wahren Verlauf der Monte-Carlo-Simulation führt.

Grundsätzlich resultiert eine Verschiebung der u Koordinate vom Bildhauptpunkt in einer schiefen Verteilung von p_δ (man beachte die Randbedingungen der Orientierung des Normalenvektors). Dies kann die Linearisierung nicht abbilden. Vom Bildhauptpunkt zu den Bildrändern nimmt die Orientierungsunsicherheit symmetrisch zu.

In **Experiment ④** wird der Einfluss der Normalenvektororientierung θ untersucht ($z = const$, $d_{pix} = const$ und $u = const$). In diesem Experiment ist θ als Winkel des Normalenvektors zur x -Achse definiert. Es werden Normalenvektoren in konstanter Entfernung untersucht, die im Bildhauptpunkt liegen und deren Patchpunkte gleich weit entfernt sind.

Die Form der Verteilungsdichte p_δ gleicht bei θ_1 einer schiefen Verteilung. Bei θ_2 entspricht p_δ nahezu einer Normalverteilung, mit geringem Mittelwert μ_{delta} und kleiner Standardabweichung σ_δ .

Bei der Orientierung θ_1 ergibt sich eine schiefe Verteilungsfunktion p_δ , während sich bei θ_2 eine Verteilungsfunktion p_δ mit deutlich kleinerer Standardabweichung ergibt.

Bei der Variation der Orientierung θ weist die Monte-Carlo-Simulation einen symmetrischen Verlauf der Mittelwerte μ_δ und der Standardabweichung σ_θ relativ zur Orientierung $\theta = 0^\circ$ auf (Punkt- und Achsensymmetrie). Die zunächst erfolgende Zunahme des Mittelwertverlaufs von $\theta = 0^\circ$ zu $\theta = 80^\circ$ kann mit dem dominanten Einfluß der schiefen Verteilungsdichte begründet werden, während die folgende Abnahme des Mittelwerts mit der zunehmenden Ausrichtung des Patches mit den Sichtstrahlen der Kamera begründet werden kann. Der Verlauf der Mittelwerte über die Orientierungen kann von der Linearisierung nicht wiedergegeben werden. Zudem weicht die Standardabweichung der Linearisierung vom Verlauf der Monte-Carlo-Simulation ab (besonders im Bereich um $\theta = 0^\circ$).

Generell bewirkt die Orientierungsänderung im gezeigten Experiment eine steigende Genauigkeit der Normalenvektoren bei zunehmendem θ . Das Disparitätsrauschen bewirkt eine Verteilung der Patchpunkte entlang der Kamer-

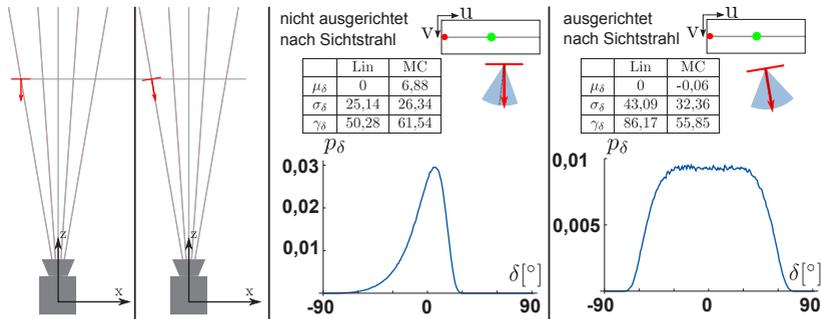


Bild 3.8: Aufbau von Experiment ⑤: Analyse der Verteilungsdichte p_δ für Normalenvektoren am Bildrand, die in Richtung der z-Achse oder des Kamerasichtstrahls ausgerichtet sind.

asichtstrahlen. Deshalb kann die steigende Genauigkeit bei zunehmendem θ mit der sich angleichenden Ausrichtung der Patchpunkte mit den Kamerasichtstrahlen erklärt werden. Die Form der Verteilungsdichte p_δ entspricht der einer schiefen Verteilung für $\theta \neq 0^\circ$.

Die bisherigen 2d-Simulationen haben gezeigt, dass die Linearisierung den Verlauf der Verteilungsdichte p_δ der Monte-Carlo-Simulation nicht ausreichend genau wiedergeben kann. Deshalb wird bei den weiteren 2d-Simulationen nur noch die Monte-Carlo-Simulation berücksichtigt. Die Variation der Entfernung z und der Abstände der Patchpunkte d_{ps} haben in mittelwertfreien Verteilungsdichten p_δ resultiert.

In Experiment ⑤ wird weiter untersucht, ob bei der Variation der Bildposition u und der Ausrichtung θ auch eine mittelwertfreie Verteilung p_δ erreicht werden kann. Zudem wird untersucht, ob es eine maximale Orientierungsungenauigkeit der Normalenvektoren gibt, wenn u und θ variiert werden.

Das Experiment ist in drei Szenarien S1, S2 und S3 unterteilt (Bild 3.9). Die Szenarien unterscheiden sich in der Definition der Entfernung und der Referenzachse für den Winkel θ . Als Entfernungsmaß wird zusätzlich zur Entfernung entlang der Kameraachse z die Euklidische Distanz d_{Euc} verwendet (S3). Als Referenzachse für θ wird neben der Kameraachse z der Sichtstrahl der Kamera durch den betrachteten Oberflächenpunkt genommen (S2 und S3).

In Experiment ⑤ wird der Einfluss der Normalenvektororientierung θ für Bildpositionen u außerhalb des Bildhauptpunkts untersucht ($z = const$ und $d_{pix} = const$). Wie in Bild 3.8 gezeigt, wird zunächst derselbe Oberflächenpunkt am linken Bildrand betrachtet, bei dem der Normalenvektor in Richtung der Kameraachse z und des Kamerasichtstrahls durch den Oberflächenpunkt orientiert

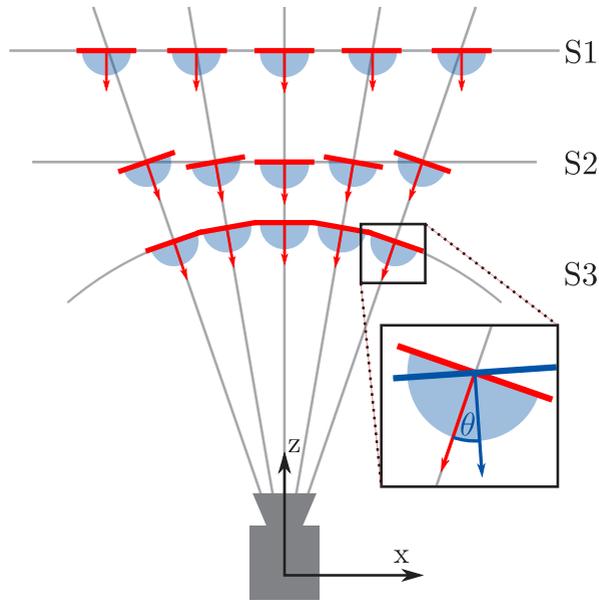


Bild 3.9: Skizze der **Simulationsszenarien S1, S2 und S3** welche in Experiment ⑤ verwendet werden: dargestellt sind die synthetisch erstellten, idealen Flächennormalen $\mathbf{n}_{\text{world}}$ und entsprechenden Patches. In rot ist der zentrale Normalenvektor mit zugehörigem Patch markiert. Die blau hinterlegte Fläche repräsentiert die Richtung aller weiteren, synthetisch erzeugten Normalenrichtungen.

ist. Die Orientierung des Normalenvektors entlang des Sichtstrahls vergrößert die Orientierungsunsicherheit, resultiert dafür aber in einer nahezu mittelwertfreien Verteilung von p_δ .

Die Szenarien S1, S2 und S3 ermöglichen die Analyse der Orientierungsgenauigkeit bei variierenden θ und u bei verschiedenen Entfernungsmaßen.

Die Auswertung der Mittelwerte μ_δ der Verteilungsfunktion p_δ für die unterschiedlich definierten Normalenvektororientierungen $\theta = 0^\circ$ ist in Bild 3.10a gezeigt. Die Orientierung des Normalenvektors entlang der Kamerasichtstrahlen (S2 und S3) ergibt über die Bildbreite betrachtet eine nahezu mittelwertfreie Verteilung. Die dazugehörige Standardabweichung σ_δ aller Szenarien nimmt zu den Bildrändern hin zu (siehe Bild 3.11a), wie schon in Experiment ③ beobachtet.

Bei Betrachtung der Mittelwerte μ_δ und Standardabweichungen σ_δ der Verteilungsfunktion p_δ am linken Bildrand für verschiedene Orientierungen θ ergibt sich wiederum eine symmetrische Verteilung für die Szenarien S2 und S3 in Bild

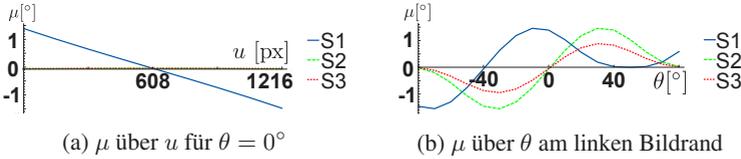


Bild 3.10: μ über u und θ

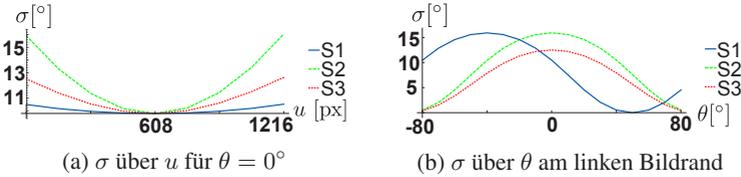


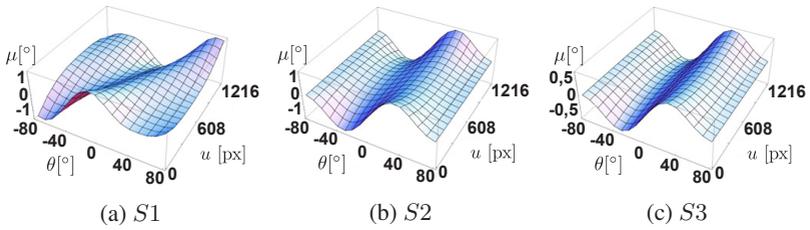
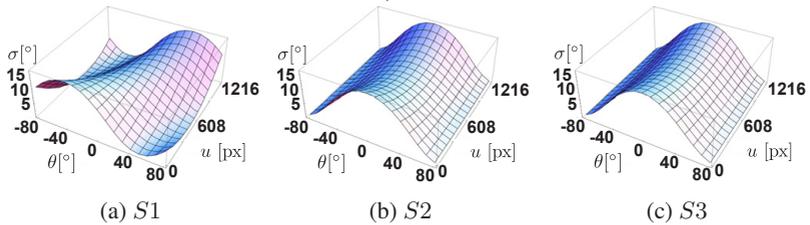
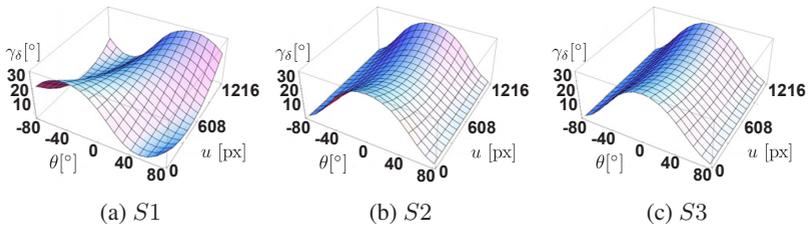
Bild 3.11: σ über u und θ

3.10b und 3.11b. Im Gegensatz dazu weist der μ_δ - und σ_δ -Verlauf von S1 keine Symmetrie auf.

In Bild 3.12 bis 3.14c sind die Verläufe von μ_δ , σ_δ und γ_δ über alle Bildpositionen u und Normalenvektororientierungen θ aufgetragen. Bei genauerer Betrachtung von γ_δ ist festzustellen, dass die maximale Orientierungsunsicherheit von S2 und S3 für alle Bildpositionen u bei $\theta = 0^\circ$ liegt. Daraus folgt, dass die mittelwertbehafteten Verteilungen von p_δ für verschiedene Orientierungen θ und Bildpositionen u keinen Einfluß auf die maximale Orientierungsunsicherheit von γ_δ hat. Betrachtet man den Verlauf von γ_δ für $\theta = 0^\circ$ der Szenarien S2 und S3, nimmt die Orientierungsungenauigkeit bei beiden Szenarien zu den Bildrändern zu. Der Euklidische Abstand d_{Euc} von Szenario S3 bewirkt also eine Abschwächung der Zunahme der Orientierungsungenauigkeit gegenüber Szenario S2.

Erweiterung der Simulationen ① und ②

In Bild 3.15 wird untersucht, ob die beobachtete Abnahme der Orientierungs-genauigkeit der Normalenvektorschätzung mit zunehmender Entfernung z und zunehmendem Abstand zum Bildhauptpunkt d_{pp} durch die Vergrößerung der Patchpunktabstände d_{pix} kompensiert werden kann. Dazu werden zunächst die konstanten Verteilungsdichten $p_\delta^{z_1}$ in der Entfernung z_1 (aus Experiment ①) und $p_\delta^{d_{pp,1}}$ mit $d_{pp,1}$ (aus Experiment ③) als Referenz bestimmt. Die Verteilungsdichten $p_\delta^{z_2}$ in der Entfernung z_2 und $p_\delta^{d_{pp,2}}$ mit Abstand $d_{pp,2}$ zum Bildhauptpunkt weisen eine größere Orientierungsunsicherheit als $p_\delta^{z_1}$ und $p_\delta^{d_{pp,1}}$ auf. Durch die Variation von d_{pix} werden zwei Mengen an Dichtefunktionen $M(p_\delta^{z_2})$ und $M(p_\delta^{d_{pp,2}})$ erstellt. Die Kompensation der Orientierungsunsicherheit wird

Bild 3.12: μ über u und θ Bild 3.13: σ über u und θ Bild 3.14: γ_δ über u und θ

zwischen den Referenzverteilungsdichten und den entsprechenden Dichtefunktionen aus $M(p_\delta^{z^2})$ und $M(p_\delta^{d_{pp}})$ getrennt betrachtet. Als Vergleichsmaß zwischen den Verteilungsdichten wird die Jensen-Shannon-Divergenz D_{JS} (Bild 3.15a und 3.15b) betrachtet, wie in Kapitel 2.3.2 definiert. Zusätzlich wird das 95%-Konfidenzintervall γ_δ der Verteilungsdichten als alternatives Vergleichsmaß herangezogen.

Zu erkennen ist sowohl an der Jensen-Shannon-Divergenz als auch an dem 95%-Konfidenzintervall, dass eine komplette Kompensation der Orientierungsunsicherheit bezüglich z und d_{pp} durch einen größeren Patchpunkteabstand d_{pix} erreicht werden kann. Dies ist an den Punkten mit minimaler Divergenz D_{JS} der Fall, als auch an den Schnittpunkten der γ_δ Verläufe. Es werden durch beide Vergleichsmaße dieselben Werte für d_{ps} erreicht.

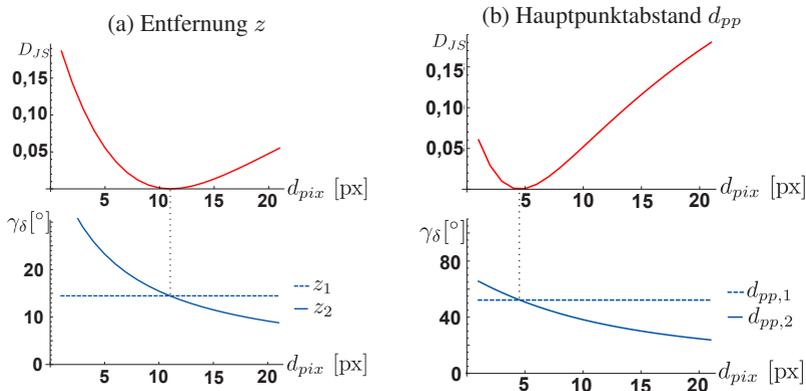


Bild 3.15: Erweiterung der Simulationen ① und ②: Vergleich verschiedener Verteilungsdichten p_δ mit der Jensen-Shannon-Divergenz D_{JS} und dem 95%-Konfidenzintervall γ_δ . Die Kompensation der höheren Orientierungsunsicherheit bei zunehmender Entfernung z wird in (a) und bei zunehmendem Abstand d_{pp} wird in (b) gezeigt.

Die Orientierungsunsicherheit des Normalenvektors in der Entfernung z_2 kann durch den Patchpunkteabstand $d_{pix} = 11$ Pixel der Orientierungsunsicherheit des Normalenvektors in der Entfernung z_1 angeglichen werden. Ebenso kann die gleiche Orientierungsunsicherheit beim größeren Abstand $d_{pp,2}$ durch den Patchpunkteabstand $d_{pix} = 5$ Pixel erreicht werden.

Das Wichtigste in Kürze

Die Orientierungsgenauigkeit binokularer Normalenvektoren nimmt ab

- mit zunehmender Entfernung z in Weltkoordinaten zwischen dem betrachteten Oberflächenpunkt und der Kamera.
- mit kleiner werdendem Abstand d_{pix} in der Bildebene zwischen den Patchpunkten.
- mit steigender Distanz d_{pp} in der Bildebene zwischen Oberflächenpunkt und Bildhauptpunkt.
- mit sich angleichender Orientierung des Normalenvektors mit dem Kamerasichtstrahl durch den betrachteten Oberflächenpunkt in Weltkoordinaten.

Der Effekt der abnehmenden Orientierungsgenauigkeit bei zunehmender Entfernung z und steigendem Abstand d_{pp} kann durch das Vergrößern des Patchpunkteabstands d_{pix} kompensiert werden.

Als Vergleichsmaß zwischen verschiedenen Verteilungsdichten p_δ eignet sich das 95%–Konfidenzintervall γ_δ .

Die höchste Orientierungsgenauigkeit wird bei der Ausrichtung des Normalenvektors entlang des Sichtstrahls erreicht.

Eine Linearisierung kann die Ergebnisse der Monte-Carlo-Simulation nicht ausreichend genau wiedergeben und wird im Folgenden nicht mehr berücksichtigt.

3.2.2 3d-Simulationen

Die 3d-Simulationen verwenden alle die in Kapitel 2.3.3.2 eingeführte Monte-Carlo-Simulation. Um die bereits beobachteten Symmetrieeffekte aus den 2d-Simulationen weiter validieren zu können, wird Simulationsszenario S3 weiter betrachtet. Ziel ist es, die maximale Orientierungsunsicherheit für $\theta = 0^\circ$ auch für Oberflächenpunkte nachzuweisen, die nicht auf den Achsen der Bildhauptpunkts in der Bildebene liegen. Zudem soll die annähernd kreisförmige Verteilung gleicher Orientierungsunsicherheiten bzgl. des Bildhauptpunktes in der Bildebene nachgewiesen werden.

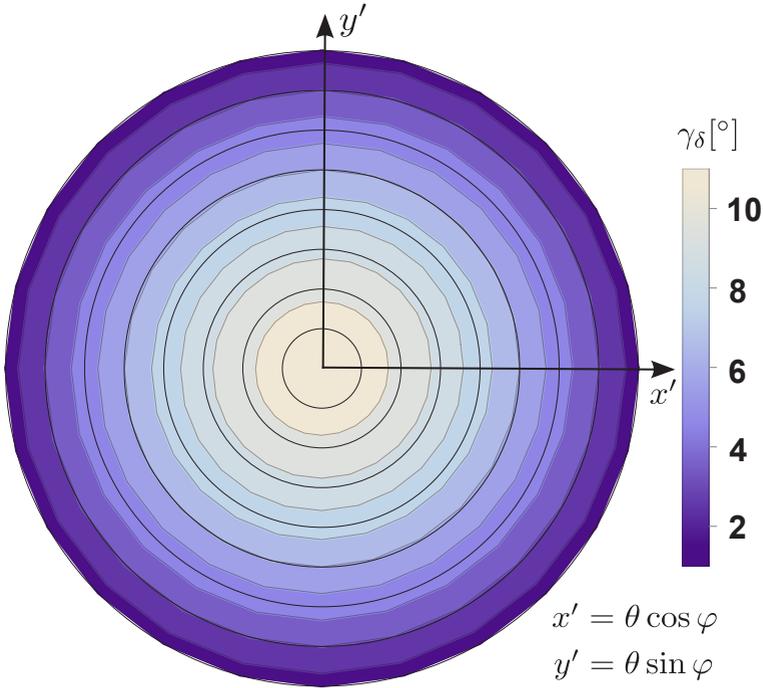


Bild 3.16: Polarkoordinatendarstellung der Orientierungsunsicherheitsverteilung für alle synthetisch erzeugten, idealen Normalenvektororientierungen für S_3 . Der 3d-Oberflächenpunkt liegt in der Entfernung $d_{Euc} = 10$ Metern an Bildposition $\mathbf{p}_E = (732, 51)$ Pixel (siehe Bild 3.17). Der zentrale Normalenvektor ist entlang des Kamerasichtstrahls ausgerichtet. Die schwarzen Kreise repräsentieren Schritte für θ von 10° vom inneren Kreis ($\theta = 10^\circ$) bis zum äußeren Kreis ($\theta = 80^\circ$).

Im dreidimensionalen Fall der Monte-Carlo-Simulation werden die idealen Normalenvektorrichtungen mit den Polarkoordinaten θ und φ bestimmt. Zur Normalenvektorschätzung werden quadratische Patches in der Bildebene mit neun Punkten verwendet. Die Seitenlänge der Patches wird als d_{ps} bezeichnet, siehe Bild 3.2. Zur Schätzung des Normalenvektors \mathbf{n}_{disp} aus den unsicherheitsbehafteten Patchpunkten wird eine Hauptkomponentenanalyse verwendet, wie in Anhang A.3 beschrieben. Der Eigenvektor mit dem kleinsten Eigenwert wird als Oberflächennormale \mathbf{n}_{disp} verwendet. Um den Normalenvektor \mathbf{n}_{world} zu bestimmen, werden \mathbf{n}_{disp} und der Mittelpunkt \mathbf{p}_M der unsicherheitsbehafteten Patchpunkte in Formel 2.11 eingesetzt.

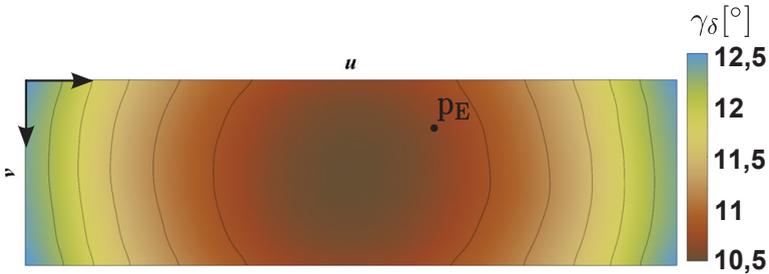


Bild 3.17: γ_δ ($\theta = 0^\circ$) für $S3$ ist in der Bildebene dargestellt mit konstanter Euklidischer Distanz $d_{Euc} = 10$ Meter. Die Bildposition p_E ist in Bild 3.16 ausgewertet.

Im ersten 3d-Experiment wird ein Oberflächenpunkt an der Bildposition p_E in 10 Metern Euklidischer Distanz betrachtet, welcher nicht auf einer Bildachse mit dem Bildhauptpunkt liegt. In Bild 3.16 ist die Verteilung des 95%-Konfidenzintervalls γ_δ in Polarkoordinatendarstellung gezeigt, welche sich für alle Normalenrichtungen der Hemisphäre ergibt. Der zentrale Normalenvektor ist entlang des Kamerasichtstrahls orientiert. Die maximale Orientierungsunsicherheit weist auch im 3d-Fall der Normalenvektor auf, der entlang des Kamerasichtstrahls ausgerichtet ist ($\theta = 0^\circ$).

Die Verteilung der maximalen Orientierungsunsicherheit der Normalenvektoren für $\theta = 0^\circ$ wird nun in der gesamten Bildebene untersucht. Dazu wird in Bild 3.17 die Verteilung von γ_δ für alle Normalenvektoren ausgewertet, die entlang der Kamerasichtstrahlen ausgerichtet sind. Es ergibt sich eine Verteilung für γ_δ in der Bildebene, die im Bildhauptpunkt minimal ist und mit zunehmendem Abstand d_{pp} zum Bildhauptpunkt zunimmt. Dieses Verhalten konnte im 2d-Fall beobachtet werden und gilt auch im 3d-Fall.

3.3 Herleitung der optimalen Patchgröße für binokulare Normalenvektoren

In diesem Abschnitt werden zwei Methoden entwickelt, welche in Echtzeitanwendungen bei der Berechnung von binokularen Normalenvektoren eingesetzt werden können. Beide Methoden verwenden die Eigenschaften der Orientierungsgenauigkeit binokularer Normalenvektoren der 2d- und 3d-Monte-Carlo-Simulationen aus den vorherigen Abschnitten.

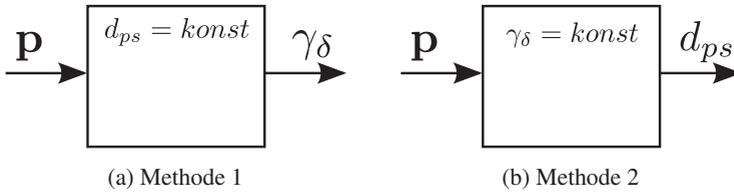


Bild 3.18: Blockschaltbilder der Methoden 1 und 2

In vielen praktischen Anwendungen, die lokale Oberflächenorientierungen von 3d-Punktwolken verwenden, ist die Abschätzung oder Einhaltung der zu erwartenden Orientierungsunsicherheit der lokalen Oberflächen sehr hilfreich. Die Orientierungsunsicherheit kann für die mathematische Formulierung der Problemstellung und für die Genauigkeitsanforderungen der Funktionssicherheit zentral sein.

Wie die 2d- und 3d-Simulationen aus Kapitel 3.2 gezeigt haben, eignet sich eine Monte-Carlo-Simulation zur Analyse der Orientierungsunsicherheit binokularer Normalenvektoren und wird im Folgenden verwendet. Als Maß der Orientierungsunsicherheit der Normalenvektoren wird das 95%-Konfidenzintervall γ_δ der Verteilungsdichte p_δ gewählt. Dieses 95%-Konfidenzintervall γ_δ ist abhängig von der verwendeten Patchgröße d_{ps} , der Orientierung des Normalenvektors und der Position \mathbf{p} des Oberflächenpunkts, welche durch die Euklidische Distanz d_{Euc} in Weltkoordinaten und dem Abstand zum Bildhauptpunkt d_{pp} in der Bildebene ausgedrückt werden kann.

Die Orientierung des Normalenvektors hat einen sehr starken Einfluß auf γ_δ . Im Folgenden soll γ_δ für jeden beliebigen Oberflächenpunkt und ohne vorheriges Wissen des Disparitätsbilds bestimmt werden. Deshalb wird für alle Oberflächenpunkte die Orientierung der Normalenvektoren mit der größtmöglichen Orientierungsunsicherheit γ_δ gewählt. Wie in Kapitel 3.2.2 simulativ gezeigt, entspricht dies der Ausrichtung der Normalenvektoren entlang des Kamerasichtstrahls. Für andere Anwendungen kann die Orientierung des Normalenvektors mit in Betracht gezogen werden, indem zunächst eine lokale Orientierung aus dem Disparitätsbild für jeden Oberflächenpunkt geschätzt wird. Ein solcher Ansatz wird in dieser Arbeit nicht weiter verfolgt, da der Rechenaufwand für praktische Anwendungen zu hoch ist.

Unter diesen Voraussetzungen werden zwei Methoden entwickelt:

- Methode 1 bestimmt die maximal zu erwartende Orientierungsunsicherheit γ_δ für einen beliebigen Oberflächenpunkt \mathbf{p} , wenn zur Schätzung der

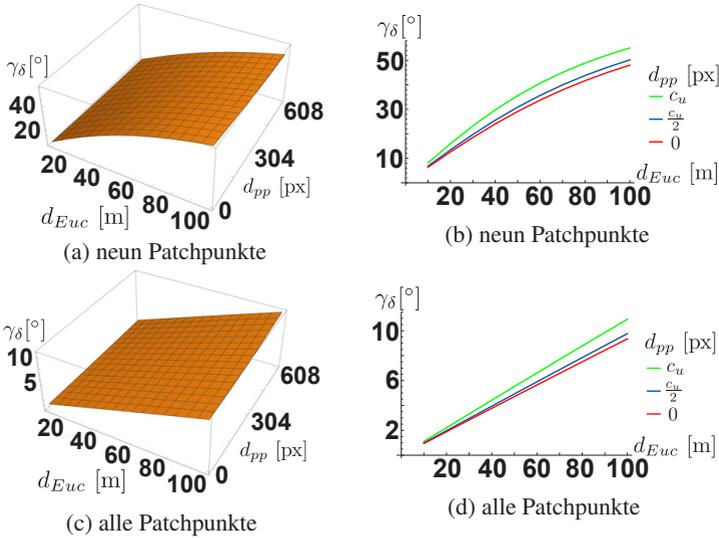


Bild 3.19: Methode 1: Beziehung zwischen dem Abstand d_{Euc} in Metern und der Entfernung zum Hauptpunkt d_{pp} in Pixeln, um eine Orientierungsgenauigkeit γ_δ zu bestimmen ($d_{ps} = konst = 15$ Pixel). (a) und (b) zeigen die Beziehung, wenn neun Patchpunkte zum Schätzen des Normalenvektors $\hat{\mathbf{n}}_{\text{world}}$ verwendet werden. (c) und (d) zeigen die Beziehung unter Verwendung aller Patchpunkte, um den Normalenvektor $\hat{\mathbf{n}}_{\text{world}}$ zu schätzen.

Normalenvektoren eine konstante Patchgröße d_{ps} im Disparitätsbild verwendet wird.

- Methode 2 bestimmt die Patchgröße d_{ps} für jeden Oberflächenpunkt \mathbf{p} , so dass eine vorgegebene, konstante Orientierungsunsicherheit γ_δ eingehalten wird.

Die zwei Methoden sind als Blockschaltbilder in Bild 3.18 abstrahiert. Beide Methoden können in Anwendungen als Look-Up-Tabellen hinterlegt werden, um in Abhängigkeit der substituierten Oberflächenpunkte $\mathbf{p} = (u, v, d)^\top$ die Orientierungsunsicherheit γ_δ bzw. die Patchgröße d_{ps} zur Laufzeit zu bestimmen.

Zur Veranschaulichung werden beide Methoden für das binokulare Kamerasystem visualisiert, welches in den Simulationen verwendet wurde. Bild 3.19a und 3.19b visualisiert die Look-Up-Tabelle für Methode 1 wenn eine Patchgröße von $d_{ps} = konst = 10$ Pixel mit neun Patchpunkten verwendet wird. Bild 3.20a und 3.20b zeigen die Look-Up-Tabellen für Methode 2 bei einer geforderten maximalen Orientierungsunsicherheit $\gamma_\delta = konst = 10^\circ$.

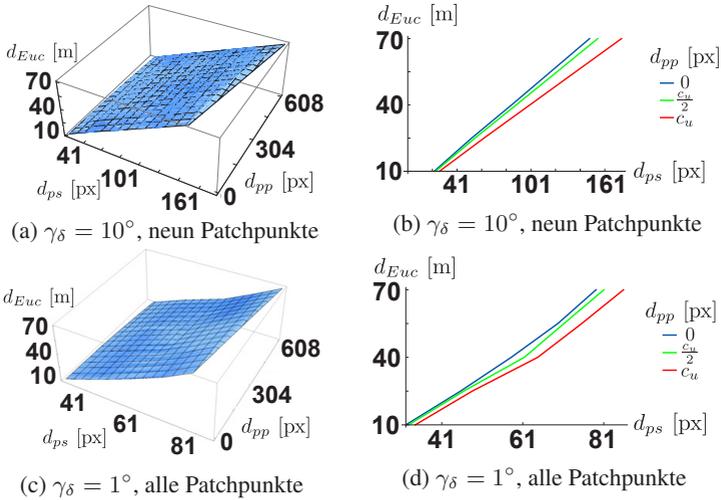


Bild 3.20: Methode 2: Beziehung zwischen dem Abstand d_{Euc} in Metern und der Entfernung zum Hauptpunkt d_{pp} in Pixeln, um eine Patchgröße d_{ps} zu bestimmen ($\gamma_\delta = konst$). (a) und (b) zeigen die Beziehung für $\gamma_\delta = 10^\circ$ unter Berücksichtigung von neun Patchpunkten zum Schätzen des Normalenvektors $\hat{\mathbf{n}}_{world}$. (c) und (d) zeigen die Beziehung für $\gamma_\delta = 1^\circ$ unter Verwendung aller Patchpunkte, um den Normalenvektor $\hat{\mathbf{n}}_{world}$ zu schätzen.

Beide Methoden werden unter den gleichen Randbedingungen in Bild 3.19c und 3.19d bzw. Bild 3.20c und 3.20d (Methode 2: $\gamma_\delta = 1^\circ$) gezeigt, wenn alle Punkte zur Normalenvektorschätzung genutzt werden, die im Patch liegen. Da deutlich mehr als neun Patchpunkte zur Normalenschätzung verwendet werden, sind die Orientierungsunsicherheiten von Methode 1 und die Patchgrößen von Methode 2 kleiner im Vergleich zur Normalenvektorschätzung mit nur neun Patchpunkten.

3.4 Experimente

In diesem Abschnitt werden die Simulationsergebnisse aus Kap. 3.2 der Genauigkeitsuntersuchungen mit Daten aus realen Experimenten verglichen. Für die realen Experimente wird ein binokulares Kamerasystem mit verschiedenen, geringen Basisbreiten verwendet (siehe Bild 3.21), mit denen zwei ebene Wandflächen beobachtet werden. Wandfläche 1 ist schräg zum Kamerasystem orientiert. Der Oberflächenpunkt hat eine Euklidische Distanz zur Kamera von $d_{Euc} = 2$ m. Wandfläche 2 wird frontoparallel beobachtet, wobei der ausgewertete Oberflä-



(a) Hardwaresetup Wandfläche 1



(b) Hardwaresetup Wandfläche 2

Bild 3.21: Für die Experimente verwendetes binokulares Kamerasystem mit verschiedenen Basisbreiten für die beiden beobachteten Wände

chenpunkt eine Entfernung von $d_{Euc} = 8$ Metern hat. Die Kalibrierparameter können Tabelle 3.2 entnommen werden.

Kamerasystem	Auflösung [px]	Basisbreite [cm]	Brennweite [px]	Hauptpunkt [px]
Wandfläche 1	640x480	3	604	$(336, 204)^T$
Wandfläche 2	640x480	11	601	$(348, 203)^T$

Tabelle 3.2: Wichtige Parameter der binokularen Kamerasysteme

Es werden kleine Basisbreiten und eine geringe Auflösung gewählt, um die simulativ beobachteten Eigenschaften des Tiefenrauschens auf die Orientierungsgenauigkeit der Normalenvektoren bereits im Nahbereich beobachten zu können. Die beobachtete Fläche deckt einen Großteil des Bilds ab und hat geringe Tiefenabweichungen, die im Vergleich zu den Tiefenabweichungen des Disparitätsrauschens vernachlässigbar sind.

Für die Experimente wird ein statischer Aufbau gewählt, bei dem das binokulare Kamerasystem dieselbe Fläche über die Zeit beobachtet. Um Redundanz zu gewährleisten, wird eine konstante Umgebungshelligkeit für alle beobachteten Zeitpunkte der Szene sichergestellt. Zur Berechnung des Disparitätsbilds wird ein Stereomatcher verwendet, der auf dem Semi-Global Block Matching (SGBM) [24] beruht. Dieser ist insbesondere ideal zur Schätzung der Disparitäten auf frontoparallelen Flächen geeignet. Bei Block-Matchern ist zu erwarten, dass die Messfehler benachbarter Disparitätswerte innerhalb der Matching-Blöcke korreliert sein werden. Die Korrelationskoeffizienten benachbarter Disparitätswerte können deshalb als positiv angenommen werden. In den folgen-

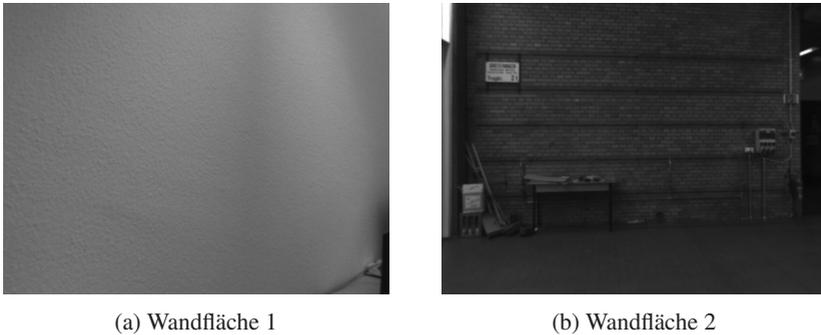


Bild 3.22: Beobachtete Wandflächen der Experimente

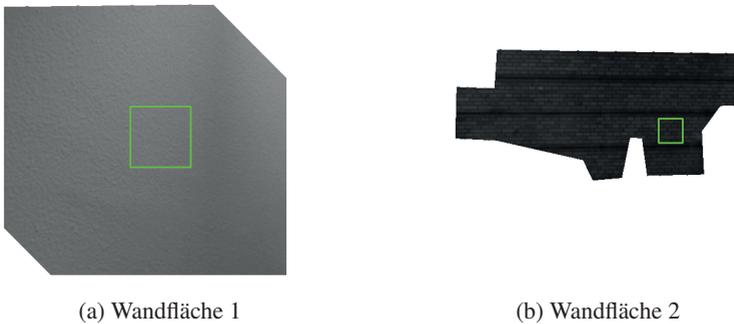


Bild 3.23: Verwendeter Ausschnitt der Wandfläche, die zur Berechnung des mittleren Normalenvektors berücksichtigt wird. In grün ist das größte, verwendete Patch zur Normalenvektorschätzung gezeigt.

den Experimenten mit Realdaten sind dadurch kleine Abweichungen der Orientierungsunsicherheit γ_δ im Vergleich zu den Ergebnissen der Monte-Carlo-Simulation mit unkorreliertem Disparitätsfehler zu erwarten, da der positive Korrelationskoeffizient die geschätzte Normalenvektororientierung der Realdaten im Idealfall kaum beeinflussen wird.

Um den mittleren Normalenvektor $\mathbf{n}_{\text{world}}$ aller n Beobachtungen über die Zeit zu bestimmen, werden zu jedem Zeitpunkt t die Beobachtungen des Disparitätsbilds \mathbf{p} der betrachteten Fläche genutzt, um $\mathbf{n}_{\text{disp},t}$ zu bestimmen. Mit der Information des Schwerpunkts $\bar{\mathbf{p}}_{\text{disp},t}$ kann der Normalenvektor $\mathbf{n}_{\text{world},t} = f(\mathbf{n}_{\text{disp},t}, \bar{\mathbf{p}}_{\text{disp},t})$ in Weltkoordinaten der Kamera bestimmt werden (siehe Formel 2.11). Der ideale Normalenvektor $\mathbf{n}_{\text{world}}$ wird durch den Mittelwert aller Normalenvektorrichtungen $\mathbf{n}_{\text{world},t}$ über die Zeit bestimmt

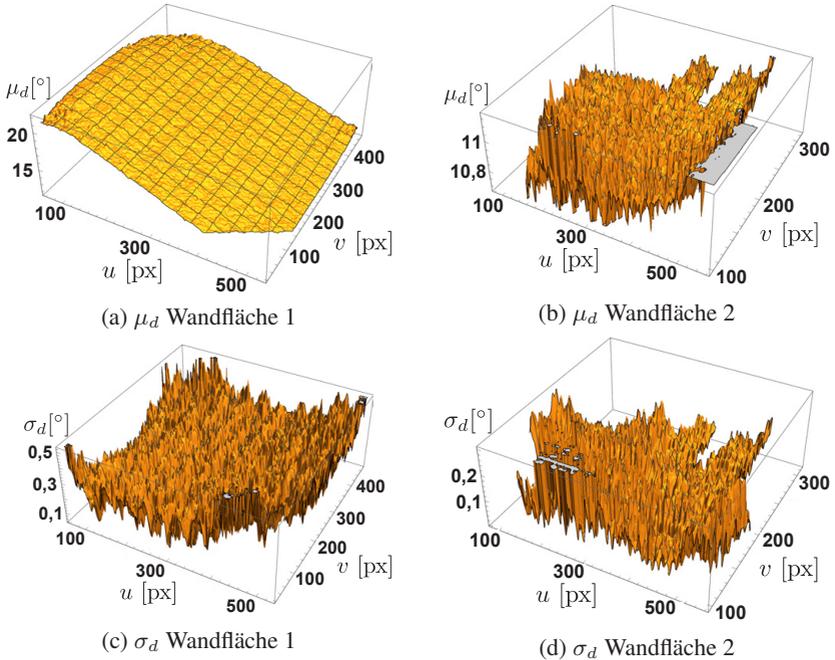


Bild 3.24: Disparitätsrauschen: Verteilung der Mittelwerte μ_d (oben) und der Standardabweichungen σ_d (unten) über die beobachtete Wandfläche

$$\mathbf{n}_{\text{world}} = \frac{1}{n} \sum_{t=1}^n \mathbf{n}_{\text{world},t}. \quad (3.10)$$

In den Experimenten ergibt sich damit eine präzise Schätzung von $\mathbf{n}_{\text{world}}$. Die Standardabweichung der Richtungsverteilung aller $\mathbf{n}_{\text{world},t}$ relativ zum mittleren Normalenvektor $\mathbf{n}_{\text{world}}$ beträgt lediglich 0.02° .

3.4.1 Untersuchung des Rauschverhaltens

Zuerst wird das Disparitätsrauschen untersucht, indem Langzeitaufnahmen der statischen Szene ausgewertet werden. Dabei wird zunächst geprüft, ob das Disparitätsrauschen einer mittelwertfreien Normalverteilung $\mathcal{N}(0, \sigma_d)$ entspricht, wie in [29] ermittelt. Dies soll insbesondere unabhängig von dem betrachteten Oberflächenpunkt gelten. Dazu wird für jedes Pixel die Verteilungsdichte der Dispa-

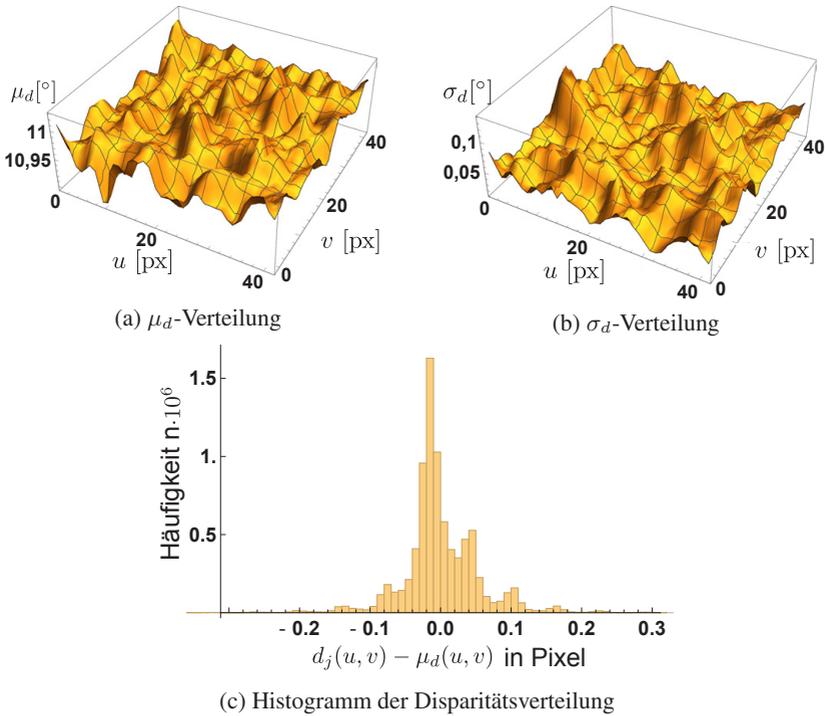


Bild 3.25: Disparitätsrauschen des betrachteten Patches auf Wandfläche 2: Verteilung der Mittelwerte μ_d (oben links) und der Standardabweichungen σ_d (oben rechts). Histogramm des Disparitätsrauschens aller Pixel relativ zum Mittelwert (unten).

ritäten ermittelt und deren Mittelwert μ_d und Standardabweichung σ_d über die Bildebene visualisiert, siehe Bild 3.24a und 3.24b.

Insbesondere sind die Disparitätsverteilungen innerhalb der betrachteten Patches interessant. Bild 3.25 zeigt die Verteilung der Mittelwerte μ_d und der Standardabweichungen σ_d des Patches auf Wandfläche 2. Neben dem Abschätzen der Wertebereiche für μ_d und σ_d kann die Annahme eines mittelwertfreien Disparitätsrauschens geprüft werden. In Bild 3.25c wird dazu ein Histogramm aus allen Disparitätswerten $d_j(u, v)$ erstellt ($j \in [0, 5000]$), von welchen der Mittelwert $\mu_d(u, v)$ für alle Patch-Pixel (u, v) subtrahiert wurde. Das Histogramm lässt auf eine mittelwertfreie Verteilung der Disparitäten zurückschließen, welche allerdings nicht perfekt normalverteilt ist.

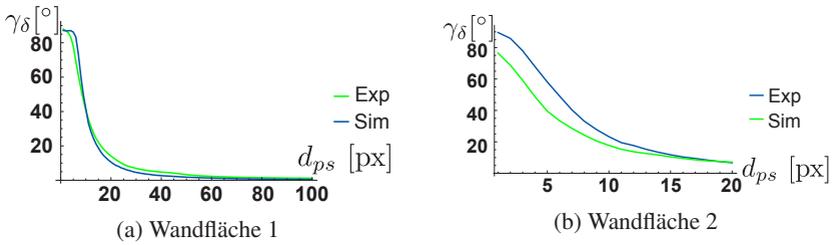


Bild 3.26: Vergleich der Orientierungsgenauigkeit γ_δ der Simulation (Sim) und der realen Daten aus den Experimenten (Exp) für verschiedene Patchgrößen d_{ps}

Die Mittelwertverteilung aus Bild 3.25a stellt eine nahezu ebene Fläche dar mit einem Mittelwert von $\mu_d = 10,98$ Pixel. Die Verteilung der Standardabweichungen σ_d liegt in einem Wertebereich von 0,01 bis 0,14 Pixel. Für alle betrachteten Patches der Wandfläche werden geeignete Standardabweichungen σ_d gewählt. Die Monte-Carlo-Simulation von Wandfläche 1 verwendet einen Mittelwert $\mu_d = 16,05$ Pixel.

3.4.2 Vergleich der Simulationsergebnisse mit Realdaten

Zum Vergleich zwischen den Realdaten und der Monte-Carlo-Simulation werden die Dichtefunktionen p_δ mit dem 95%–Konfidenzintervall γ_δ verglichen. Die Orientierungsgenauigkeit γ_δ wird an einem Oberflächenpunkt für verschiedene Patchgrößen d_{ps} bestimmt. Die Verteilungsdichte p_δ wird aus 5000 Bildern der Realdaten, bzw. aus 5000 Iterationen der Monte-Carlo-Simulation bestimmt. Bei einem ähnlichen Verlauf von γ_δ entspricht die verwendete Monte-Carlo-Simulation dem realen Verhalten der erwarteten Orientierungsunsicherheit der binokularen Normalenvektoren.

In Bild 3.26 sind die Verläufe der Orientierungsgenauigkeit γ_δ über die Patchgröße d_{ps} der beiden Wandflächen gezeigt. In blau sind die Ergebnisse der Monte-Carlo-Simulation (Sim) zu sehen, in grün ist der Verlauf der Experimente aus den Realdaten (Exp) dargestellt. Der reale Verlauf der γ_δ -Verteilung kann von der Monte-Carlo-Simulation generell wiedergegeben werden. Die Abweichungen aus den Verläufen ist im Wesentlichen auf die abweichenden Disparitätsverteilungen der Wandflächen zurückzuführen. Die Verteilung der Mittelwerte μ_d wird in der Monte-Carlo-Simulation so angenommen, dass sich eine perfekte Ebenenfläche ergibt. Diese Voraussetzung wird gemeinsam mit der Annahme

der konstanten Standardabweichung der Disparitäten verletzt, wie in Bild 3.24 sichtbar ist. Daraus folgen die leicht abweichenden Verläufe von γ_δ in Bild 3.26.

Im Rahmen der Messungenauigkeit der durchgeführten Experimente, der starken Vereinfachung des vorliegenden Disparitätsrauschens und der relativ geringen Anzahl an Beobachtungen kann das Ergebnis der Monte-Carlo-Simulation als ausreichend genau angenommen werden. Der zu erwartende Verlauf der Winkelabweichungen aus den Realdaten kann durch die Monte-Carlo-Simulation wiedergegeben werden.

3.5 Zusammenfassung

In diesem Kapitel wurde die Orientierungsgenauigkeit binokularer Normalenvektoren untersucht. Die Genauigkeitsuntersuchung hat die wesentlichen Parameter identifiziert, welche für eine Modellierung dieser Orientierungsschätzung notwendig sind. Neben der Orientierung des Normalenvektors, werden die Position des Oberflächenpunkts und die Patchgröße der lokalen Umgebung um den Oberflächenpunkt berücksichtigt, um die Orientierungsgenauigkeit der Normalenvektoren zu modellieren.

Die Ergebnisse der Experimente zeigen, dass eine Monte-Carlo-Simulation zur Abschätzung der Orientierungsgenauigkeit verwendet werden kann. Eine analytische Abschätzung der Gauß'schen Fehlerfortpflanzung kann den genauen Verlauf der Monte-Carlo-Simulation nicht wiedergeben. Mithilfe der Monte-Carlo-Simulation sind verschiedene Methoden entwickelt worden, welche es ermöglichen, die Orientierungsunsicherheit für eine bestimmte Patchgröße und umgekehrt abzuschätzen. Abschließend wurden die Ergebnisse der Monte-Carlo-Simulation anhand von realen Experimenten validiert.

Das vorgestellte Verfahren zur Modellierung der Orientierungsschätzung binokularer Normalenvektoren liefert bereits vielversprechende Ergebnisse. Mit der zusätzlichen Berücksichtigung weiterer Einflüsse in der Monte-Carlo-Simulation, wie der Strukturiertheit oder den Beleuchtungsbedingungen der betrachteten Oberfläche, besteht noch weiteres Verbesserungspotenzial für eine genauere Schätzung der zu erwartenden Normalenvektororientierung.

4 Anwendungsbeispiele mit binokularen Normalenvektoren

In diesem Kapitel werden zwei Beispiele für die Anwendung von binokularen Normalenvektoren in der Umfeldwahrnehmung vorgestellt. Die Untersuchung der binokularen Normalenvektoren aus Kapitel 3 ermöglicht eine Abschätzung der Orientierungsgenauigkeit für diese Anwendungen.

Auf dem Weg zu hochautomatisiert fahrenden Automobilen ist eine hochpräzise Umfeldwahrnehmung notwendig, um genaue on-board Lokalisierung [33], Trajektorienplanung [71] oder die Schätzung des eigenen Fahrstreifens [6] realisieren zu können. Bereits in heutigen Serienfahrzeugen werden für fortgeschrittene Fahrerassistenzsysteme (Advanced Driver Assistance Systems - ADAS) Freibereichsschätzungen für Notbremssysteme oder automatische Ausweichassistenten verwendet. Eine Umsetzung vieler Funktionen wird mittels Kamerasystemen realisiert. Diese werden heutzutage in vielen Automobilen serienmäßig eingebaut, da sie neben der flexiblen Einsetzbarkeit auch kostengünstig zur Verfügung stehen.

Speziell in städtischen Umgebungen ist die Detektion von Freibereichen herausfordernd, da oft Straßenmarkierungen und begrenzende Infrastruktur wie beispielsweise Leitplanken fehlen. In diesen Umgebungen stellen Bordsteine ein wertvolles Merkmal für die Begrenzung des Freibereichs dar. Kamerasysteme bieten die Möglichkeit, Bordsteine aus Intensitätskanten des Grauwertbilds zu detektieren. Dieses Vorgehen ist sehr fehleranfällig bei schlechten Lichtverhältnissen, irreführenden Straßenmarkierungen (z.B. an Straßenkreuzungen) und Schattenbereichen auf der Straße. Um diese Nachteile zu umgehen, wird die vorgestellte Freibereichsschätzung ausschließlich mithilfe von Tiefendaten realisiert.

Dreidimensionale Bildsensoren sind in den letzten Jahren zudem zur Oberflächenrekonstruktion [35], Objekterkennung [28], Gestenerkennung [50] und vielen anderen Bereichen der Robotik eingesetzt worden. Dadurch wurden wesentliche Fortschritte auf dem Gebiet der Umfeldwahrnehmung erzielt, die sich neben dem Feld der Assistenzsysteme für Automobile auch in den Bereichen der autonom navigierenden Roboter verbreitet haben.

Setup	KITTI	Helmkamarasystem
Bildauflösung [px]	1220 x 370	640 x 480
Basisbreite B [m]	0.54	0.183
Brennweite F [px]	722	360
Bildhauptpunkt \mathbf{p}_{pp} [px]	$(609, 173)^T$	$(302, 231)^T$

Tabelle 4.1: Parameter der binokularen Kamerasysteme aus dem KITTI-Datensatz bzw dem Helmkamarasystem.

Zur Zeit werden diese Fortschritte vermehrt für Blindenassistenzsysteme eingesetzt [1]. Menschen mit eingeschränkter Sehfähigkeit sind in der Regel auf einen Langstock angewiesen, der eine sehr vorsichtige Fortbewegung in unbekanntem Umgebungen oder in Umgebungen mit Stufen erfordert. Gerade in diesen Situationen ist es hilfreich, für betroffene Menschen zusätzliche Informationen über das Umfeld bereitzustellen. Stufen und Treppen stellen für Menschen mit eingeschränkter Sehfähigkeit solche gefährliche Situationen im alltäglichen Leben dar.

Beide Anwendungen, die in dieser Arbeit vorgestellt werden, sind im Bereich der Umfeldwahrnehmung für binokulare Kamerasysteme einzuordnen. Die erste Anwendung stellt eine Freibereichsschätzung für Automobile aus einem vorwärtsgerichteten binokularen Kamerasystem vor, während die zweite Anwendung eine Treppenerkennung für Blindenassistenzsysteme mittels einem auf dem Kopf getragenen binokularen Kamerasystem einführt. Beide Beispiele zeigen den Nutzen von Orientierungsinformationen aus Tiefendaten in echtzeitfähigen Anwendungen für die Umfeldwahrnehmung.

4.1 Genauigkeitsanforderungen der Normalenvektoren

Die Orientierungsgenauigkeit der Normalenvektoren ist von der verwendeten Hardware abhängig, welche im Folgenden betrachtet wird. Die vorgestellte Freibereichsschätzung für Automobile verwendet das binokulare Kamerasystem der Grauwertkameras aus dem KITTI-Datensatz [17], welches in Fahrzeuglängsrichtung auf dem Dach montiert ist (siehe Bild 4.1a). Das binokulare Kamerasystem der vorgestellten Treppendetektion für Menschen mit Sehbehinderung ist dage-

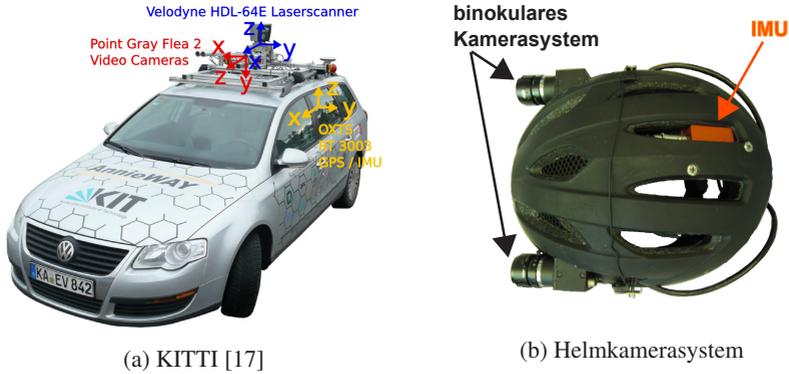


Bild 4.1: Hardware-Aufbau der verwendeten binokularen Kamerasysteme

gen in Blickrichtung in einem Helm integriert (siehe Bild 4.1b). Die Parameter der beiden binokularen Kamerasysteme sind in Tabelle 4.1 aufgelistet.

Für beide Anwendungen soll ein quadratisches Patch mit konstanter Größe d_{ps} zur Normalenvektorberechnung für alle Oberflächenpunkte genutzt werden, um die Echtzeitfähigkeit der Anwendungen zu ermöglichen. Die im weiteren Verlauf des Kapitels eingeführte Freibereichsschätzung soll mit Oberflächennormalen hohe Bordsteine bis zu einer Entfernung von $z_1 = 30$ m detektieren, während die Treppendetektion Oberflächennormalen im Nahbereich bis zu $z_2 = 5$ m in Betracht zieht.

Zunächst wird aus diesen Informationen (Kameraparameter und maximale Entfernung) eine Maximalpatchgröße bestimmt, so dass die senkrechten Flächen zumindest von einem Normalenvektor erfasst werden können. Mit Gleichung 2.8 lässt sich für 30 cm hohe Bordsteine eine Patchgröße $d_{ps,1}^{\max} = 7$ Pixel abschätzen und für Stufen von 15 cm Höhe eine Patchgröße von $d_{ps,2}^{\max} = 11$ Pixel bestimmen:

$$d_{ps}^{\max} = \frac{F \Delta y}{z} \quad (4.1)$$

Mithilfe von Methode 1 kann für eine konstante Entfernung nun die Genauigkeit abgeschätzt werden. Diese Genauigkeit ist in Abhängigkeit zum Bildhauptpunktstand d_{pp} und der Patchgröße d_{ps} in Bild 4.2 für beide Hardwaresetups dargestellt. Unter Berücksichtigung der maximalen Patchgrößen d_{ps}^{\max}

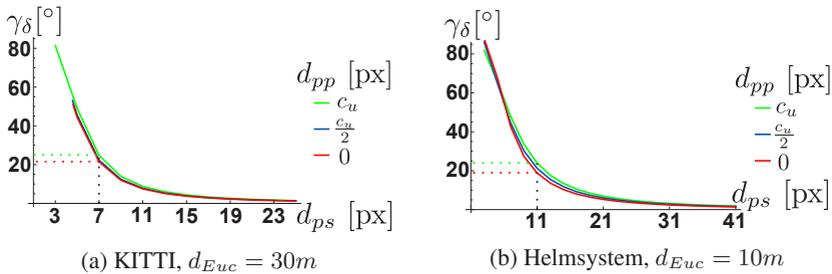


Bild 4.2: Beziehung zwischen der Orientierungsgenauigkeit der Normalenvektoren γ_δ , dem Abstand d_{pp} des Oberflächenpunkts zum Bildhauptpunkt und der Patchgröße d_{ps} für eine konstante Euklidische Distanz d_{Euc} zwischen Kamera und betrachtetem Oberflächenpunkt.

ergibt sich eine Oberflächengenauigkeit $\gamma_{\delta,1} = 25^\circ$ für die Freibereichsschätzung und $\gamma_{\delta,2} = 24^\circ$ für die Treppendetektion.

4.2 Orientierungsbild

Beide Anwendungen setzen voraus, dass ein kalibriertes binokulares Kamerasystem verwendet wird, das dichte Disparitätsbilder aus den rektifizierten Grauwertbildpaaren durch einen Stereomatcher in Echtzeit zur Verfügung stellt. Aus den Disparitätsbildern wird für jedes Pixel ein Normalenvektor $\hat{\mathbf{n}}_{disp}$ geschätzt, der zur Normalenvektorschätzung alle Pixel berücksichtigt, die innerhalb der quadratischen Patchfläche der Größe d_{ps} um das betrachtete Pixel liegen (siehe Bild 3.2). Die Normalenvektoren werden direkt in der Sensordomäne geschätzt, wie von Badino et al. in [5] empfohlen. Die geschätzten Normalenvektoren $\hat{\mathbf{n}}_{disp}$ aus dem Disparitätsbild werden für jedes Pixel in Weltkoordinaten zu $\hat{\mathbf{n}}_{world}$ transformiert, wie in Kapitel 2.2.2 beschrieben. Im Folgenden wird das Orientierungsbild I_δ definiert, das eine pixelweise Oberflächenorientierung berechnet, die invariant gegen Kamerarotationen ist. Dazu wird für jedes Pixel ein Orientierungswinkel δ berechnet, der als Winkel zwischen dem lokal geschätzten Oberflächennormalenvektor $\hat{\mathbf{n}}_{world}$ und einer globalen Hochachse \mathbf{h}_{world} in Weltkoordinaten definiert ist:

$$\delta = \arccos \frac{\mathbf{h}_{world} \cdot \hat{\mathbf{n}}_{world}}{\|\mathbf{h}_{world}\| \|\hat{\mathbf{n}}_{world}\|}. \quad (4.2)$$

δ beschreibt einen Raumwinkel, so dass $\delta \in [0^\circ, 90^\circ]$ gilt. Die globale Hochachse \mathbf{h}_{world} kann beispielsweise von den Beschleunigungs- und Magnetfeldsensoren einer inertialen Messeinheit (IMU) als unabhängige Messung vorliegen

oder durch zeitliches Filtern des Normalenvektors der Grundebene aus der Kamerasensorik geschätzt werden.

Das Orientierungsbild I_δ beinhaltet die Abweichungswinkel δ für jede Pixelposition (u, v) und wird in beiden Anwendungen normalisiert, so dass $I_\delta \in [0, 1]$.

In Beispielbild 4.11a ist exemplarisch ein Orientierungsbild gezeigt. Senkrechte Flächen sind rot dargestellt, während horizontal ausgerichtete Flächen grün gekennzeichnet sind.

4.3 Freibereichsschätzung für Automobile

Das im Folgenden vorgestellte Anwendungsbeispiel stellt eine zeitlich gefilterte Freibereichsschätzung für Automobile [20] vor, die auf dichten Disparitätsbildern aus binokularen Kameras beruht. Der Fokus dieses Beitrags richtet sich auf komplexe, innerstädtische Umgebungen, in denen der Freibereich durch flache Infrastruktur wie Bordsteine oder durch größere Hindernisse wie beispielsweise andere Verkehrsteilnehmer begrenzt wird. Freibereichsschätzungen, die auf Einzelbildauswertungen beruhen, werden stark von Sensorrauschen und Artefakten in Tiefendaten beeinflusst und führen zu starken Abweichungen vom real existierenden Freibereich.

Dieser Beitrag zur Freibereichsschätzung verwendet deshalb zeitlich gefilterte Höhen- und Orientierungsinformationen in probabilistischen Belegungsgittern. Die Belegungsgitter haben durch die zeitliche Filterung den Vorteil gegenüber existierenden, auf Einzelbildern basierenden Methoden, dass bereits erfasste Freibereichsbegrenzungen berücksichtigt werden, die zwischenzeitlich sichtbar sind und zu einem späteren Zeitpunkt außerhalb des aktuellen Sensorsichtbereichs liegen oder verdeckt werden. Der vorgestellte Ansatz wird auf verschiedenen städtischen Szenarien evaluiert und mit einer existierenden Freibereichsschätzung verglichen.

Das vorgestellte Verfahren definiert den Freibereich als den Bereich vor dem Fahrzeug, der den Sensorsichtbereich durch flache oder größere Hindernisse beschränkt. Eine Übersicht des Gesamtsystems ist in Bild 4.3 dargestellt und wird in Abschnitt 4.3.2 dieses Kapitels näher erläutert. Die verwendete Systemstruktur erlaubt die Detektion des fahrbaren Bereichs in komplexen Umfeldszenarien, wie sie oft in städtischen Umgebungen vorliegen.

Im Folgenden werden in Abschnitt 4.3.1 ein Überblick über verwandte Arbeiten im Bereich der Bordstein- und Freibereichsschätzung gegeben. Kapitel 4.3.2 beschreibt detailliert die Freibereichsschätzung, die auf zeitlich gefilterten, mehrschichtigen Belegungsgittern beruht. In Kapitel 4.4.4 werden die Ergebnis-

se der Freibereichsschätzung gezeigt, sowie die Vorteile und Herausforderungen der vorgestellten Methode diskutiert.

4.3.1 Stand der Technik

Die Übersicht der Methoden zur Bordstein- und Freibereichsschätzung ist in Verfahren mit unterschiedlicher Sensorik strukturiert, die insbesondere monokulare Kameras, Lidarsensoren und binokulare Kamerasysteme umfassen.

Auf monokularen Kameras basierende Ansätze zur Freibereichsdetektion konzentrieren sich auf die Kantenextraktion (um z.B. Straßenbegrenzungen zu bestimmen) und die Auswertung der Farbinformationen (um z.B. Fahrbahnmarkierungen oder Gebiete mit spezifischen Farben zu detektieren). Viele Ansätze [2] verwenden Segmentierungstechniken auf monokularen Kamerabildern, um den Freibereich vor Fahrzeugen zu schätzen. Diese Ansätze erweisen sich in innerstädtischen Umgebungen als fehleranfällig, da häufig auftretende Schattenwürfe und Teerflecken die Segmentierungsverfahren stören. Eine Übersicht zur Bordsteindetektion aus monokularen Kameras ist in [55] zu finden. Kuehnl et al. [32] kombinieren in Ihrem Ansatz monokulare Bildinformationen mit Tiefeninformationen, um Straßenbegrenzungen zu detektieren. Der Ansatz ist in der Lage, Straßenbegrenzungen wie Bordsteine und unbefestigte Randstreifen zu erkennen und liefert bereits vielversprechende Ergebnisse. Ebenso beeindruckende Ergebnisse liefert der Ansatz von Schomerus et al. [53], der das monokulare Kamerabild nutzt, um auf Liniensegmenten in vordefinierten Bildbereichen eine eindimensionale Signalverarbeitung vorzunehmen. Neben Bordsteinen und unbefestigten Seitenstreifen ist der Ansatz in der Lage, auch parkende Autos zu detektieren.

Zusammenfassend bieten Bildinformationen aus monokularen Kameras viel Potenzial bei der Freibereichsdetektion, sind aber sehr anfällig bei Bildrauschen durch schlechte Umgebungslichtbedingungen oder bei projizierten Mustern auf der Fahrbahnoberfläche (wie z.B. Schattenwürfen von Objekten oder Ausbesserungen des Fahrbahnbelags). Um diese Nachteile zu überwinden, kommen Sensoren, die dreidimensionale Daten erfassen, zum Einsatz.

Der Velodyne-Lidarsensor bietet in der Fahrzeugumfelderfassung eine 360°-Rundumsicht, insbesondere wenn der Sensor auf dem Fahrzeugdach montiert ist. Der Sensor erlaubt eine Auswertung der 3d-Punktwolke (10^6 Punkte/s) bis Entfernungen von 150 m. In statischen Umgebungen nutzen Moosmann et al. [38] den Velodyne-Lidar, um mit einem graphenbasierten Ansatz Objekte durch ein lokales Konvexitätskriterium zu detektieren und von der Straßenfläche zu unterscheiden. Der Ansatz von Hu et al. [27] extrahiert aus den Velodyne-Lidardaten Bereiche der Straßenebene und integriert diese in einem Straßen-Belegungsgitter. In einem zweiten Schritt wird das Belegungsgitter mit den Farbinformationen

einer Kamera kombiniert und daraus der Freibereich geschätzt. Die Velodyne-Punkt看ke wird von Shinzato et al. [57] nur teilweise genutzt, indem die Punkt看ke im Sichtbereich einer virtuellen Kamera mithilfe eines Lochkameramodells projiziert wird. Aus allen projizierten Tiefenpunkten wird durch Triangulation ein Graph erstellt. Hindernisse werden in einem weiteren Schritt klassifiziert, indem die Tiefendifferenz von benachbarten Punktpaaren berücksichtigt wird. Der Freibereich wird anschließend aus einem polaren Entfernungshistogramm bestimmt. Lidare, die nur wenige Ebenen scannen, werden ebenso zur Freibereichsschätzung verwendet. Charakteristisch ist die hohe Auflösung in horizontaler Richtung bei gleichzeitig geringer Auflösung in vertikaler Richtung. Einen solchen Lidar verwenden Kellner et al. [31], um aus dem Signalverlauf der horizontalen Scanebene Bordsteine zu detektieren.

Binokulare Kamerasysteme bieten die Möglichkeit, dichte dreidimensionale Tiefendaten in Echtzeit zu erzeugen. Die Disparitätsbilder werden mit einem Stereomatcher [23][18][48] aus den rektifizierten Bildpaaren bestimmt, welche die Epipolargeometrie erfüllen. Das Disparitätsbild wird gewöhnlich verwendet, um ein parametrisches Oberflächenmodell auf die Straßenoberfläche zu schätzen und unter Berücksichtigung dieses Oberflächenmodells einen Freibereich zu bestimmen. Einen solchen Ansatz verwenden Pfeiffer et al. [45], um den Freibereich mit dynamischer Programmierung zu bestimmen. Dabei wird in der Bildebene zu jedem Zeitpunkt separat ein Kostenbild erzeugt. Der kürzeste Pfad vom linken zum rechten Bildrand stellt die Begrenzung des Freibereichs dar, welche mit einem Graphlöser bestimmt wird. Shinzato et al. [56] haben den bereits erwähnten Ansatz [57] für Lidare auf Disparitätskarten portiert. Der Freibereich wird in der Bildebene geschätzt, indem die Begrenzungspunkte eines Strahlbündels gesucht werden, welches am Mittelpunkt der untersten Zeile zentriert ist. Die Ausrichtung der Normalenvektoren aus der 3d-Punkt看ke berücksichtigen Fernandez et al. [15] für eine Bordstein- und Fahrstreifendetektion. Die Kombination aus V-Disparität und Superpixeln aus dem Grauwertbild nutzen Vitor et al. [65] als Eingangsdaten für ein künstlich neuronales Netzwerk, welches den Freibereich schätzt. Wang et al. [67] klassifizieren zunächst die Straßenfläche einer monokularen Farbkamera und führen mit der V-Disparität eine Freibereichsschätzung durch. Enzweiler et al. [13] verwenden einen texturbasierten Bordsteinklassifikator, der ein mehrschichtiges Neuronales Netzwerk mit lokalen Merkmalen verwendet und anschließend 3d-Informationen aus dem Disparitätsbild integriert.

Belegungsgitter [12] werden oft in Anwendungen im Automobilbereich zur zeitlichen Filterung der Umgebungsinformationen verwendet. Badino et al. [4] verwenden die probabilistische Repräsentation der Belegungsgitter [64], um den

Freibereich durch dynamische Programmierung in einem polaren Belegungsgitter zu schätzen, welches aus dichten Disparitätskarten erstellt wird.

Oniga et al. [42] verwenden dichte Disparitätsbilder in einer digitalen Höhenkarte (DEM, *engl.*: Digital Elevation Map) und modellieren die Bordsteine als kubische Polynome in der DEM.

4.3.2 Methodik

Die im Folgenden eingeführte Methode zur Freibereichsschätzung definiert den Freibereich als die Fläche vor dem Fahrzeug, die durch Bordsteine oder größere Hindernisse begrenzt wird. Der vorgestellte Ansatz filtert die Bordstein- und Hindernisinformation zeitlich in einem mehrschichtigen Belegungsgitter [12]. Dabei werden keine parametrischen Modellannahmen bezüglich dem gesuchten Freibereich und des Bordsteinverlaufs getroffen. Zudem werden keine direkten Informationen aus dem Grauwertbild genutzt, da aufgrund von Schattenkanten oder Bildrauschen in schlecht belichteten Bildregionen Missklassifikationen auftreten. Im Folgenden wird der Ablauf der vorgestellten Methode erläutert. Eine abstrakte Übersicht über den Gesamtprozess ist in Bild 4.3 zu finden.

4.3.2.1 Eingangsdaten

Die vorgestellte Methode setzt ein kalibriertes binokulares Kamerasystem voraus, welche zwei synchronisierte und rektifizierte Bilder zu jedem Zeitschritt als Eingangsdaten für die weitere Verarbeitungsschritte liefert. Aus jedem Bildpaar wird ein dichtes *Disparitätsbild* mit einem Stereomatcher [48] berechnet.

Für jedes Disparitätsbild wird im Weiteren ein *Orientierungsbild* I_δ berechnet, das jedem Bildpixel einen Raumwinkel $\delta \in [0^\circ, 90^\circ]$ des lokalen Normalenvektors $\hat{\mathbf{n}}_{\text{world}}$ zu einer globalen Hochachse $\mathbf{h}_{\text{world}}$ zuweist. Details zur Berechnung von I_δ sind in Kapitel 4.2 beschrieben.

Die *Bewegung* zwischen zwei aufeinanderfolgenden Zeitschritten wird durch visuelle Odometrie kompensiert [19].

4.3.2.2 Mehrschichtiges Belegungsgitter

Zur zeitlichen Filterung der Fahrzeugumgebung wird ein mehrschichtiges Belegungsgitter genutzt [12]. Das Belegungsgitter liegt in einer virtuellen Grundebene, welche durch die extrinsische Kalibrierung fest definiert ist. Der Ursprung des Koordinatensystems des Belegungsgitters bleibt relativ zum Fahrzeug-Koordinatensystemursprung konstant. Die Bewegung des Fahrzeugs entspricht der Bewegung des Belegungsgitters, während die aktuelle Orientierung des Fahrzeugs im

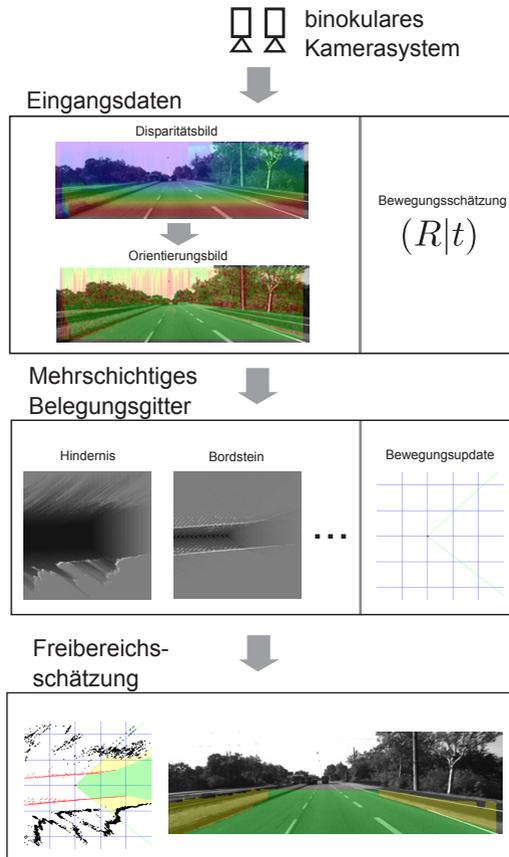


Bild 4.3: Hauptschritte der Bordstein- und Freibereichsschätzung aus Tiefendaten

Bewegungsgitter hinterlegt ist. Die Eigenbewegung des Fahrzeugs wird durch zwei Komponenten kompensiert. Durch die passende Verschiebung des Belegungsgitters (Translation des Ego-Fahrzeugs) und das Aktualisieren der Fahrzeugorientierung (Rotation des Ego-Fahrzeugs) im Belegungsgitter.

Generell können beliebig viele Schichten des Belegungsgitters erzeugt werden. In dem vorgestellten Ansatz werden zwei Schichten gewählt, die Bordstein- und die Hindernisschicht. Beide Schichten werden aus den Tiefendaten des binokularen Kamerasystems aktualisiert.

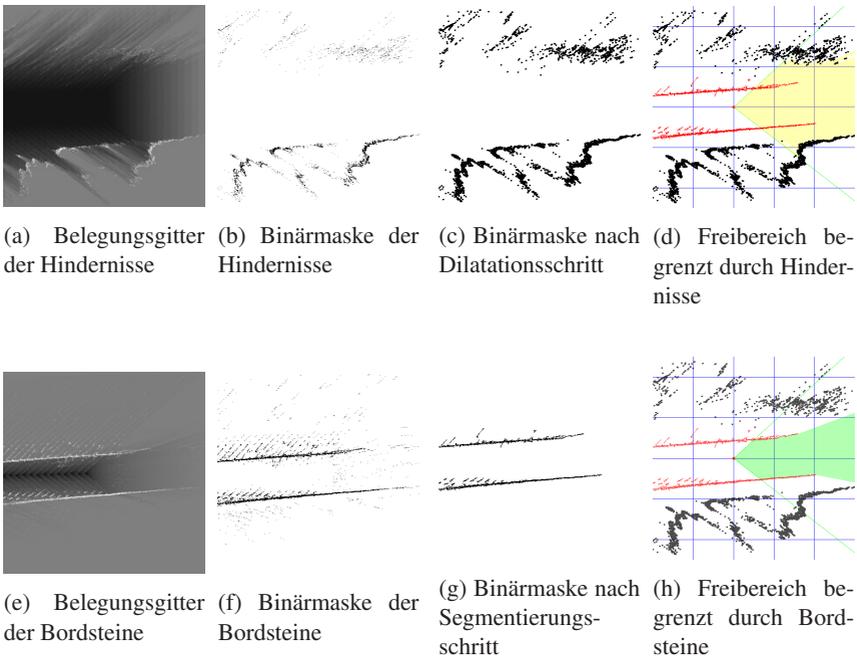


Bild 4.4: Freibereichsschätzung der Hindernisschicht (erste Reihe) und der Bordsteinschicht (zweite Reihe).

Das verwendete binokulare Kamerasystem ist näherungsweise waagrecht ausgerichtet. Senkrechte Objekte erscheinen deshalb in einer Bildspalte des Kamerabilds als Punkte gleicher Tiefe, wie beispielsweise bei Bordsteinen und größeren Hindernissen. Diese Eigenschaft wird ausgenutzt, indem die beiden Schichten des Belegungsgitters durch eine spaltenweise Bearbeitung des Disparitätsbilds bzw. Orientierungsbilds aktualisiert werden. Jede Zelle des Belegungsgitters wird als ein Bayes'sches Filter betrachtet, das die Wahrscheinlichkeit für den Zustand *belegt* schätzt. Anfänglich am nächsten Punkt des Sichtstrahls relativ zur Kamera, wird die Wahrscheinlichkeit der Zelle verringert, bis eine belegte Zelle erreicht ist. Die Wahrscheinlichkeit der ersten belegten Zelle des Sichtstrahls wird dagegen erhöht.

Die Hindernis-Schicht aktualisiert alle Zellen, bis eine Zelle mit einer Höhe größer h_{min} [m] erreicht wird. Die Bordstein-Schicht aktualisiert alle Gitterzel-



Bild 4.5: Freibereichsschätzung die eine Ebenenannahme verwendet, wie in [45] beschrieben.

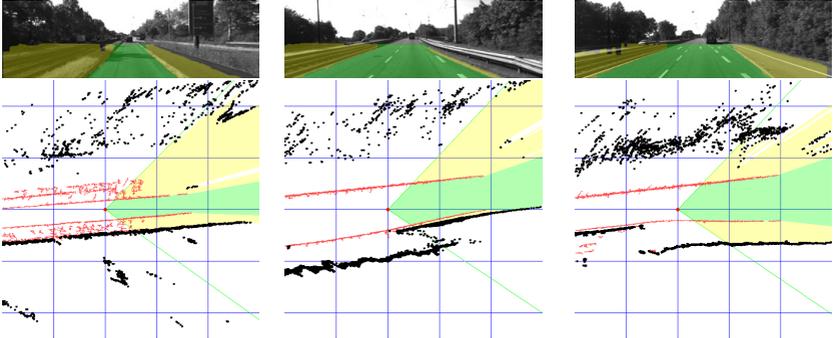


Bild 4.6: Ergebnisbilder der Freibereichsschätzung und zugehörige Belegungsgitter aus unterschiedlichen urbanen Szenarien. Der grüne Bereich stellt den extrahierten Freibereich dar, der durch Bordsteine, Hindernissen oder einem maximalen Abstand begrenzt ist. Gelbe Bereiche zeigen den Bereich hinter der erfassten Bordsteinkante. Hindernisse werden als schwarze Kreise, berücksichtigte Bordsteinkanten als rote Punkte dargestellt.

len, bis zum Erreichen der ersten Hinderniszelle oder bis eine Zelle mit einem Orientierungswert $\delta > o_{min} [^\circ]$ erreicht ist.

4.3.2.3 Freibereichsschätzung

Der vorgestellte Ansatz verwendet eine strahlenbasierte Freibereichsschätzung im Belegungsgitter. Die maximale Anzahl der verwendeten Strahlen bzw. die Winkelauflösung der Strahlen definiert sich über den Öffnungswinkel der Kamera und die horizontale Auflösung der Kamera (Anzahl der Bildspalten). Die Strahlen entspringen im Koordinatenursprung des Kamerasystems und laufen unter Verwendung des Bresenham-Algorithmus [7] bis zur ersten belegten Zelle der entsprechenden Belegungsgitterschicht.

Belegte Zellen werden durch morphologische Operationen der Belegungsgitterschichten und eine anschließende Binarisierung der Schichten definiert. Dieser Prozeß ist in Bild 4.4 näher dargestellt. Die Hindernisse werden durch eine Dila-



Bild 4.7: Freibereichsschätzung die eine Ebenenannahme verwendet, wie in [45] beschrieben.

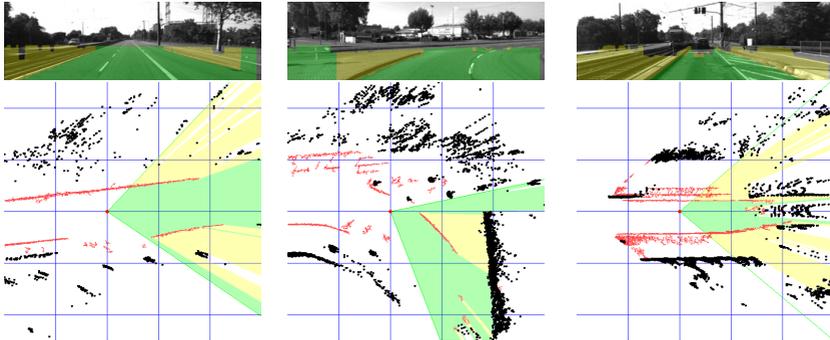


Bild 4.8: Weitere Ergebnisbilder der Freibereichsschätzung und zugehörige Belegungsgitter aus unterschiedlichen urbanen Szenarien. Beschreibung siehe Bild 4.6.

tation vergrößert, so dass Lücken innerhalb der Objekte geschlossen werden (siehe Bild 4.4a – 4.4c). Die Bordstein-Schicht beinhaltet viele kleinere Rauscharfakte. Diese werden entfernt, indem eine Segmentierung nach [63] der belegten Bordsteinzellen durchgeführt wird und anschließend kleine Segmente entfernt werden (siehe Bild 4.4e – 4.4g). Belegte Zellen werden anschließend durch eine Binarisierung bestimmt.

Der resultierende Freibereich wird als überlagerte Polygonfläche der beiden Belegungsgitterschichten dargestellt. Die Freibereichsschätzung ist im Belegungsgitter sowie als Projektion in Bildkoordinaten visualisiert, siehe Bild 4.6 und 4.8. Die grüne Fläche repräsentiert den extrahierten Freibereich, der durch Bordsteine, Hindernisse oder eine maximale Entfernung begrenzt ist. In gelb ist die Fläche visualisiert, die hinter den Bordsteinbegrenzungen detektiert wurde. Hindernisse sind als schwarze Kreise und berücksichtigte Bordsteine als rote Punkte visualisiert.

4.3.3 Ergebnisse und Auswertung

Die vorgestellte Freibereichsschätzung ist ausgelegt, um in komplexen, urbanen Umgebungen langzeitstabil zu arbeiten. Bisherige Benchmark-Tests bieten nur auf Einzelbildern eine Evaluation der Freibereichdetektion an. Deshalb wird der vorgestellte Ansatz auf einer fortlaufenden Sequenz aus den Rohdaten des KITTI-Datensatzes ausgewertet, die verschiedene, herausfordernde Freibereichssituationen zeigt. Da es keine *ground-truth* Informationen zum Freibereich gibt, wird die vorgestellte Methode qualitativ visuell evaluiert und mit einer existierenden Freibereichsschätzung [45] verglichen.

Der vorgestellte Ansatz verwendet nur Daten aus einem binokularen Kamerasystem. Die Eigenbewegung wird mit einer visuellen Odometrie [19] aus dem rektifizierten Grauwertbildpaar berechnet. Es wird ein echtzeitfähiger Stereomatcher [18] verwendet, um die dichten Disparitätsbilder zu erzeugen. Fehlende Disparitätsinformationen werden durch spaltenweise lineare Interpolation hinzugefügt.

Das mehrschichtige Belegungsgitter ist mit einer Größe von 50 x 50 m initialisiert und hat eine Zellgröße von 0.1 x 0.1 m. Der Ursprung des Kamera-Koordinatensystems liegt an der Position (20, 30) m im Belegungsgitter und wird durch einen markanten roten Punkt angedeutet (Bild 4.3). Die Kameraorientierung wird durch zwei grüne Linien dargestellt, die dem aktuellen Sichtbereich der Kamera entsprechen. Blaue Linien entsprechen Referenzlinien, die im Abstand von 10 m eingezeichnet sind.

Im Folgenden werden die Parameter gewählt, wie in Kapitel 4.3.2.2 eingeführt. Es werden die Zellhöhe $h_{min} = 0.5$ m und der Orientierungswert $o_{min} = 30^\circ$ gewählt. Die Bordstein- und Hindernisschicht des Belegungsgitters wird mit einem Schwellwert von 0.5 binarisiert. Die Hindernisschicht wird mit Kreisen des Durchmessers 0.4 m dilatiert, während die Bordsteinschicht segmentierte Bereiche mit weniger als 10 Zellen unterdrückt. Die relevanten Parameter des binokularen Kamerasystems zur dreidimensionalen Rekonstruktion der Szene können in Tabelle 4.1 gefunden werden.

Bild 4.6 und 4.8 zeigen ausgewählte Ergebnisbilder der vorgestellten Freibereichsschätzung dieser Sequenz. Der Freibereich ist im Bildbereich und im Belegungsgitter visualisiert. Bild 4.5 und 4.7 zeigen eine Vergleichsmethode zur Freibereichsschätzung, die eine Ebenenannahme verwendet und den Freibereich durch dynamische Programmierung bestimmt [45].

Im Bildbereich markieren die roten Punkte die End-Zellpositionen des Freibereichs, welcher als grüner Bereich dargestellt ist, wenn ein Bordstein oder eine maximale Entfernung den Freibereich begrenzt. Die roten Punkte treten in

der Entfernung mit höherer Dichte auf, da die Freibereichsschätzung im Belegungsgitter durchgeführt wird. Gezackte Freibereichsbegrenzungen im Bildbereich können durch Diskretisierungseffekte bei der Projektion der Belegungszellen in die Bildebene erklärt werden. Im Belegungsgitter stellen rote Zellen gefilterte Bordsteinzellen dar, schwarze Kreise stehen für detektierte Hindernisse.

Die Hindernisse aus der Belegungsgitterschicht werden robust detektiert. Die Bordsteindetektion unterdrückt kleinere Rauschartefakte, die hauptsächlich auf der Straßenoberfläche auftreten. Die Bordsteindetektion funktioniert für mittelhohe Bordsteine bis zu einer Entfernung von 30 m. Die Genauigkeit der Bordsteindetektion hängt stark von der Bordsteinhöhe und der Qualität der Tiefendaten ab.

Bei Betrachtung verschiedener Straßenszenarien in Bild 4.6 und 4.8 kann die Robustheit und Generalisierbarkeit der vorgestellten Freibereichsschätzung evaluiert werden. Verläufe von sich verengenden und weitenden Straßen können aufgrund der paramterfreien Bordstein- und Hindernisdetektion präzise geschätzt werden. Zudem werden Hindernisse wie Leitplanken oder Vegetation in den Belegungsgitterschichten kartiert und korrekt als Freibereichsbegrenzung detektiert. Im Vergleich zur Freibereichsschätzung aus [45] werden Freibereichsberandungen durch Bordsteine zuverlässig detektiert.

Schattenbereiche stellen für die vorgestellte Methode kein Problem dar, da keine Grauwertkanten berücksichtigt werden. Abgeflachte Bordsteine werden nicht als Freibereichsbegrenzung detektiert. Das ermöglicht das Überfahren von abgeflachten Bordsteinen an Einfahrten zu Gebäuden oder Parkplätzen, wie sie oft in Wohngebieten vorkommen. Ein Beispiel einer Kurve ist in Bild 4.8b dargestellt. Während die Bordsteinberandung auf der linken Bildseite detektiert wird, treten auf der rechten Bildseite Fehldetektionen von Bordsteinen aufgrund starker Tiefenartefakte auf. Bild 4.8c zeigt ein Szenario mit dynamischen Objekten (der Zug und das Fahrzeug bewegen sich geradeaus in Richtung des Ego-Fahrzeugs). Hinter den dynamischen Objekten tritt ein sogenannter Objektschweif auf, der durch die nächsten Aktualisierungsschritte der Hindernisschicht verschwindet. Bei dynamischen Objekten bietet sich eine Bewegungskompensation im Belegungsgitter an.

Herausforderungen ergeben sich in Bereichen der Disparitätsbilder, in denen zeitlich konsistente Tiefenartefakte auftreten. In diesen Bereichen, die oft in der Nähe von Straßenmarkierungen und Schattenbereichen liegen, treten in der Regel Falsch-Positiv-Detektionen auf. In diesen über- bzw. untersättigten Bildbereichen haben heutige Stereomatcher Probleme, korrekte Tiefendaten zu schätzen.

Die Modularität der mehrschichtigen Belegungskarte erlaubt die Integration von weiteren zeitlich gefilterten Merkmalschichten. Beispielsweise kann eine

Höhenschätzung für die Bordsteine integriert werden, ähnlich wie in [15] erklärt. Für Straßen- oder Fahrstreifenschätzer können in Zukunft weitere Umfeldinformationen integriert werden, wie Straßenmarkierungen und Berandungspfosten.

Die Gesamtlaufzeit des Systems liegt bei 195 ms mit vorberechneten Disparitätsbildern. Der Hauptteil der Rechenzeit auf der 2.67 Ghz dual core CPU wird von der Berechnung der visuellen Odometrie (100 ms), der Normalenvektoren und des Orientierungsbilds I_δ (80 ms), der Belegungsgitter-Aktualisierungen (6 ms) und der Freibereichsschätzung (9 ms) benötigt.

4.4 Treppendetektion für Blindenassistenzsysteme

Eine zuverlässige Umfeldwahrnehmung stellt eine zentrale Aufgabe für viele Anwendungen in der Robotik dar. Für Systeme, die zwischen verschiedenen Ebenen inner- und außerhalb von Gebäuden (Indoor und Outdoor) navigieren, ist eine robuste und präzise Treppendetektion besonders essentiell. In diesem Kapitel wird eine Treppendetektion aus Tiefendaten eines binokularen Kamerasystems vorgestellt [21]. Als Eingangsdaten werden ein dichtes Disparitätsbild und ein Orientierungsbild I_δ (siehe Kapitel 4.2) berechnet. Aus dem rotationsinvarianten Orientierungsbild werden mit einem Optimalfilter konvexe und konkave Tiefenkanten bestimmt. Liniensegmente werden in die Bereiche der Tiefenkanten geschätzt und genutzt, um zunächst Stufen und später Treppen zu detektieren. Es wird sich zeigen, dass das Verfahren gewöhnliche Treppentypen zuverlässig detektieren kann und existierende Methoden zur Treppendetektion bzgl. der Genauigkeit übertrifft. Das Verfahren eignet sich zudem als echtzeitfähiges Assistenzsystem für Menschen mit eingeschränkter Sehfähigkeit, um mithilfe einer Helmkamera zwischen Gebäudeebenen zu navigieren.

In diesem Kapitel wird eine Detektionsmethode für aufsteigende Treppen vorgestellt. Diese ermöglicht die Weitergabe nützlicher Zusatzinformationen über die Lage, Orientierung oder Abmessungen der Treppe an Blinde. Der Neuigkeitswert dieses Beitrags liegt vor allem in der direkten Detektion von dreidimensionalen konvexen und konkaven Liniensegmenten aus einer 3d-Punktwolke und der sehr hohen Detektionsgenauigkeit dieser Liniensegmente. Die Treppendetektion stellt eine mögliche Anwendung für den Nutzen solcher 3d-Tiefenkanten dar und unterstreicht den praktischen Nutzen der robusten Kantendetektion. Das vorgestellte System basiert auf der Hardware einer Helmkamera, die in Bild 4.1b gezeigt ist. Neben dem binokularen Kamerasystem ist eine inertielle Messeinheit (IMU) starr mit dem Helm verbunden, die eine globale Orientierungsreferenz liefert.

Im folgenden Abschnitt wird zunächst ein Überblick über bestehende und verwandte Treppendetektionsmethoden gegeben. Die Hauptschritte der eingeführten Treppendetektion werden detailliert in Kapitel 4.4.2 ausgeführt, bevor in Abschnitt 4.4.4 die Methode evaluiert wird.

4.4.1 Stand der Technik

In den letzten Jahrzehnten ist die Detektion von Treppen ein weit verbreitetes Forschungsthema. Dies gilt vor allem für die Bereiche der mobilen Robotik und

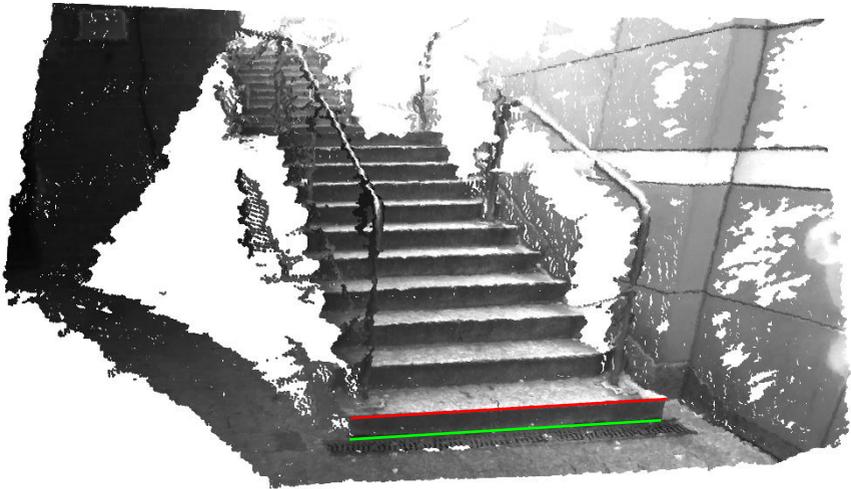


Bild 4.9: Rekonstruierte 3D-Punktwolke einer typischen Treppenszene. Die Liniensegmente der ersten Stufe sind als konkav (grün) und konvex (rot) markiert.

der Assistenz von sehbehinderten Menschen. In diesem Abschnitt wird ein Überblick über den Stand der Technik der verschiedenen Treppendetektionsmethoden gegeben. Im Wesentlichen lassen sich die verwendeten Treppenmodellierungen in zwei Ansätze unterteilen: die linien- und die ebenenbasierten Treppenmodellierungen [49].

Linienbasierte Treppendetektionen erkennen zunächst konvexe und/oder konkave Liniensegmente (siehe Bild 4.9), um diese in einem zweiten Schritt einem Treppenmodell anzupassen. Dagegen werden bei ebenenbasierten Methoden die Treppen aus einer Kombination von horizontal und senkrecht orientierten Ebenensegmenten detektiert. Die verwendete Hardware existierender Methoden reicht von monokularen Kameras bis hin zu 3d-Sensoren, wie RGB-D Kameras oder binokularen Kamerasystemen.

Die Wahl der richtigen Hardware und der passenden Methode für die Treppendetektion hängt zudem stark vom vorliegenden Anwendungsfall ab. Neben der typischerweise gewählten Hardware für Indoor- und Outdoorapplikationen ist die Position und der Blickwinkel des gewählten Sensors entscheidend für die gewählte Methode zur Treppendetektion. In dieser Arbeit wird vorausgesetzt, dass die Treppe durch konkave und konvexe Tiefenkanten beschrieben werden

kann und diese aus dem Sichtfeld der Kamera zumindest im Nahbereich sichtbar sind.

Ebenenbasierte Methoden werden im Allgemeinen mit Tiefensensoren verwendet. Oehler et al. [41] gruppieren Oberflächenelemente mithilfe einer RGB-D-Kamera. Diese Oberflächenelemente werden weiter segmentiert in größere, zusammenhängende Ebenenelementen. Pradeep et al. [46] berechnen für jeden Bildpunkt die Oberflächennormale aus einem binokularen Kamerasystem. Anschließend werden global konsistente Normalenvektoren bestimmt, welche zur Schätzung von Ebenensegmenten in der 3d-Punktwolke verwendet werden. Osswald et al. [43] extrahieren Ebenen für Tritt- und Setzstufen aus 3d-Punktwolken, die durch das Wackeln eines beweglichen, auf einem Roboterkopf montierten 2d-Lidars zusammengesetzt wird. Kürzlich wurde von Yus et al. [47] ein Verfahren für RGB-D-Kameras vorgestellt, welches ein Regionenwachstum auf 3d-Punktwolken nutzt, unter der Berücksichtigung der Normalenvektororientierung und der Oberflächenkrümmung.

Linienbasierte Treppendetektionsansätze extrahieren Kanten in der Regel aus einem monokularen Kamerabild und teilen diese in Gruppen paralleler Liniensegmente ein. Die Variation der Intensität wird verwendet, um zwischen konvexen und konkaven Liniensegmenten zu unterscheiden. Offensichtlich sind Herausforderungen in Form von Falschdetektionen zu erwarten, wenn in der Szene Schattenkanten auftreten oder aufgrund von schwierigen Lichtverhältnissen niedrige Intensitätsgradienten vorliegen.

Ein Verfahren zur Kantendetektion auf Basis von Gabor-Filtern wurde von Se und Brady [54] für monokulare Kameras vorgeschlagen. Konkave und konvexe Liniensegmente werden durch die Variation der Intensität unterschieden. Eine Kombination aus Gabor-Filtern zur Kantenerkennung und einer Gruppierungsmethode der Liniensegmente, die eine Fuzzy-Logik verwendet, wurde von Zhong et al. [70] vorgestellt. Delmerico et al. [8] verwenden eine RGB-D-Kamera, um konkave Treppenkanten aus Tiefensprüngen der 3d-Informationen zu extrahieren und kombinieren diese Beobachtungen, um ein Treppenmodell zu schätzen. Ansätze, die Informationen aus dem Grauwertbild und zusätzliche 3d-Informationen nutzen (z.B. aus einem binokularen Kamerasystem oder einer RGB-D Kamera) berücksichtigen Tiefenkanten der 3d-Punktwolke, um zwischen konvexen und konkaven Liniensegmenten zu unterscheiden. Eine häufig verwendete Methode zur Linienerfassung ist die Hough-Transformation (HT) [11]. Die Autoren aus [36] kombinieren die Tiefendiskontinuitäten mit einer Kantenerkennung eines Grauwertbilds. Diese Information wird durch eine gewichtete HT fusioniert, die die konkaven oder konvexen Kantensegmente klassifiziert. Das Auftreten von Tiefenartefakten in den Disparitätsbildern erwies sich als die größte Herausfor-

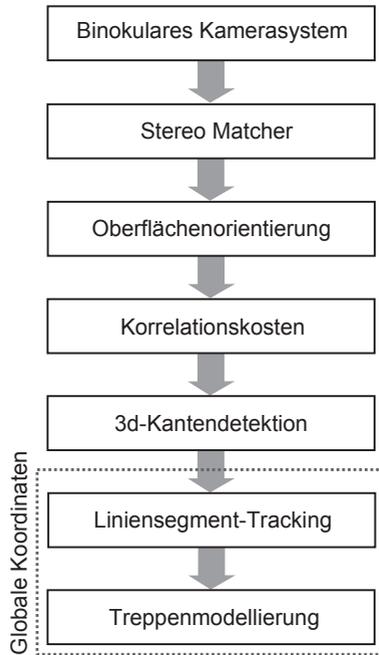
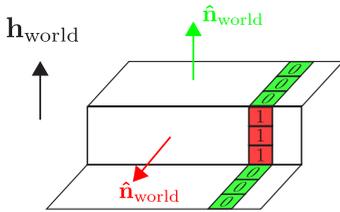
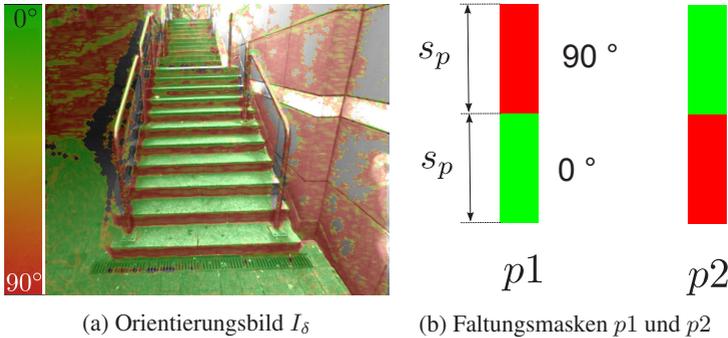


Bild 4.10: Verwendete Hauptschritte der Treppendetektion

derung bei diesem Ansatz und konnte nur teilweise durch zusätzliche Heuristiken kompensiert werden. Ansätze, die Strukturinformationen nutzen, stützen sich auf die Existenz von Intensitätsgradienten. Diese sind nicht immer vorhanden und stellen daher eine weitere Fehlerquelle dar.

Auch wenn es eine große Vielfalt unterschiedlicher linienbasierter Treppenerkennungen existieren, gibt es keinen Ansatz, der Tiefeninformationen ohne die Verwendung von Strukturinformationen zur Schätzung von konvexen und konkaven Liniensegmente nutzt. Ein solcher Ansatz hat den Vorteil, modular für verschiedene Tiefensensoren einsetzbar zu sein. Zudem können Nachteile der texturbasierten Ansätze überwunden werden, wie beispielsweise Fehldetektionen durch irreführende Schattenkanten oder nicht berücksichtigte Detektionen in Bildbereichen mit geringen Intensitätsgradienten. Der im Folgenden vorgestellte Ansatz nimmt die Herausforderung der Liniendetektion aus Tiefendaten an und wird im nächsten Abschnitt näher erläutert.



$\mathbf{h}_{\text{world}}$	$\hat{\mathbf{n}}_{\text{world}}$	δ	δ_{norm}
		0°	0
		90°	1

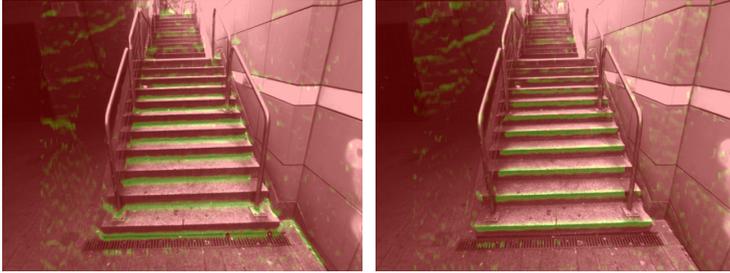
(c) Skizze der normierten Winkelabweichungen δ_{norm} einer Stufe in Weltkoordinaten.

Bild 4.11: Reales Orientierungsbild I_δ , Korrelationsmasken $p1$ und $p2$, Stufe in Weltkoordinaten

4.4.2 Methodik

In diesem Abschnitt wird ein linienbasierter Algorithmus zur Treppendetektion vorgestellt, der den Verarbeitungshauptschritten aus Bild 4.10 folgt. Das Ziel ist, eine Treppe als eine Menge von Stufen zu beschreiben, wobei jede Stufe aus einem Paar eines konvexen und konkaven Liniensegments besteht, siehe Bild 4.9. Die konkaven und konvexen Liniensegmente werden direkt aus den Tiefendaten (d.h. aus dichten Disparitätsbildern) geschätzt, um vorher genannte Nachteile anderer Verfahren auszuschließen. Die geschätzten Liniensegmente werden zeitlich assoziiert, während die 3d-Dimensionen der Liniensegmente gefiltert werden. Letztendlich werden die Liniensegmente mithilfe eines hinterlegten Treppenmodells passend zusammengefügt.

Die verwendeten Sensoren in dieser Arbeit sind ein binokulares Kamerasystem und eine IMU, die sowohl intrinsisch als auch extrinsisch kalibriert sind. Beide Sensoren sind starr an einem Helm befestigt (siehe Bild 4.1b). Weitere Informa-

(a) Kostenbild C_1 (b) Kostenbild C_2 **Bild 4.12:** Resultierende Kostenbilder

tionen zur Sensor-Hardware finden sich in Abschnitt 4.4.3. Die beiden Sensoren werden synchronisiert, so dass ein Paar rektifizierter Grauwertbilder und eine globale Orientierung des Helms für jeden Verarbeitungsschritt zur Verfügung steht. Ein Stereomatcher [24] [18] mit Subpixel-Genauigkeit wird verwendet, um dichte Disparitätsbilder zu berechnen. Aus dem dichten Disparitätsbild wird das Orientierungsbild I_δ (siehe Bild 4.11a) bestimmt, wie in Kapitel 4.2 eingeführt. Diese beiden Bilder stellen die Eingangsdaten für die im Folgenden beschriebenen Verarbeitungsschritte dar.

4.4.2.1 Korrelationskosten

Der nächste Verarbeitungsschritt erstellt zwei Kostenbilder aus I_δ , die später verwendet werden, um konkave und konvexe Liniensegmente zu extrahieren. Beide Kostenbilder bestehen aus pixelweisen Kostenwerten, die einen Übergang von flachen zu senkrechten Flächen (C_1) und umgekehrt (C_2) beschreiben. Im Folgenden werden die Kostenbilder durch C_j abgekürzt, $j \in \{1, 2\}$. Zur Berechnung von C_j werden die Korrelationsmasken p_j (siehe Bild 4.11b) auf I_δ angewendet. Dies kann durch die folgenden, spaltenweise ausgeführten Korrelationsfunktionen beschrieben werden:

$$C_1(u, v) = 1 - \frac{1}{s_p} \left(\sum_{i=v-s_p}^v I_\delta(u, i) - \sum_{i=v}^{v+s_p} I_\delta(u, i) \right) \quad (4.3)$$

$$C_2(u, v) = 1 - \frac{1}{s_p} \left(\sum_{i=v}^{v+s_p} I_\delta(u, i) - \sum_{i=v-s_p}^v I_\delta(u, i) \right) \quad (4.4)$$

Der Parameter s_p in Pixeln ist definiert als die diskrete Pixelhöhe, die für jeden betrachteten 3d-Punkt geschätzt wird. Dazu wird die Höhe h_m der Faltungsmaske in Weltkoordinaten festgelegt (z.B. $h_m = 0, 2$ Meter).

$$s_p(d) = \frac{dh_m}{B}, \quad (4.5)$$

d repräsentiert den aktuellen Disparitätswert an Pixelposition (u, v) und B die Basisbreite des binokularen Kamerasystems in Metern. Je weiter entfernt der betrachtete 3d-Punkt liegt, desto kleiner ist s_p .

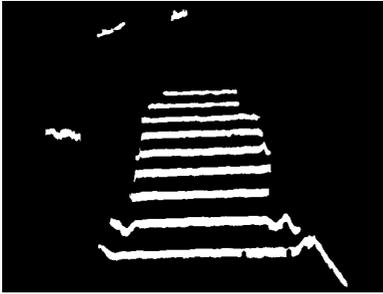
Die Korrelationsfunktionen entsprechen dem Optimalfilter [40] und können zu einer Korrelationsoperation zusammengefasst werden. Dazu kann die Kantenfilterung aus Bild 4.11c mit dem Korrelationsfilter aus Kapitel 2.4 verglichen werden.

Die beiden Kostenbilder C_j stellen pixelweise Werte zur Verfügung, die die Zugehörigkeit zu einem Übergang von flachen zu senkrechten Flächen und umkehrt beschreiben. In Bild 4.12 sind Beispielbilder für C_j dargestellt. Die Farbe Grün repräsentiert niedrige Kosten (d.h. das Muster p_j passt), während die Farbe Rot auf hohe Kosten hinweist (d.h. p_j passt nicht).

4.4.2.2 3d-Kantendetektion

In diesem Schritt werden aus den Kostenbildern C_j konkave bzw. konvexe Liniensegmente geschätzt. Beide Kostenbilder C_j weisen Histogramme mit bimodalen Verteilungen auf. Um eine Unterscheidung der Maxima in den variablen Verteilungen zu finden, wird die Intravarianz σ_w der Histogramme maximiert nach Otsu et al. [44]. Das Ergebnis ist ein Binärbild I_{bin_j} aus jedem Kostenbild C_j . Zusammenhängende Bereiche in dem Binärbild werden segmentiert mittels eines Grenzverfolgungsalgorithmus [63]. Kleine Cluster werden durch die Annahme einer Mindestclustergröße S_e unterdrückt, die eine relevante Tiefenkante in der Regel aufweist. Alle extrahierten Cluster stellen eine Interessensregion (ROI) dar, welche die relevanten Treppenkanten enthalten, siehe Bild 4.13a und 4.13b.

In den extrahierten ROIs werden in einem weiteren Schritt Liniensegmente geschätzt, welche den konkaven und konvexen Treppenkanten entsprechen sollen. Dazu wird zunächst jede ROI mit dem entsprechenden Kostenbild multipliziert, siehe Bild 4.13d. Die globale Lösung für die Tiefenkante einer beliebigen Form innerhalb jeder ROI ist der kürzeste Pfad, der z.B. durch die Verwendung des Dijkstra-Algorithmus [9] berechnet werden kann. Die Berechnung des kürzesten Pfads wird im Folgenden nicht verwendet, da sie rechenintensiv ist und einen definierten Start- und Endpunkt erfordert, der aufgrund der beliebigen Ausrichtung



(a) Binärmaske I_{bin_1} aus C_1 für konkave Kanten



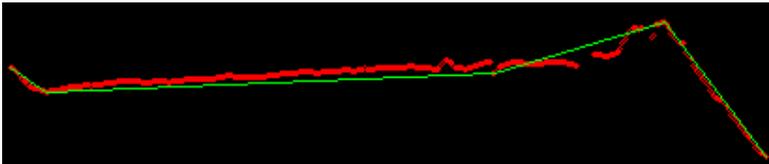
(b) Binärmaske I_{bin_2} aus C_2 für konvexe Kanten



(c) Ein Segment (ROI) der Binärmaske



(d) Multiplikation der Binärmaske mit dem Kostenbild. Schwarz stellt hohe Kosten dar, weiß niedrige Kosten.



(e) Rote Punkte repräsentieren Kostenminima, grüne Liniensegmente werden durch einen iterativen Endpunkt Algorithmus geschätzt.

Bild 4.13: Liniensegmentschätzung

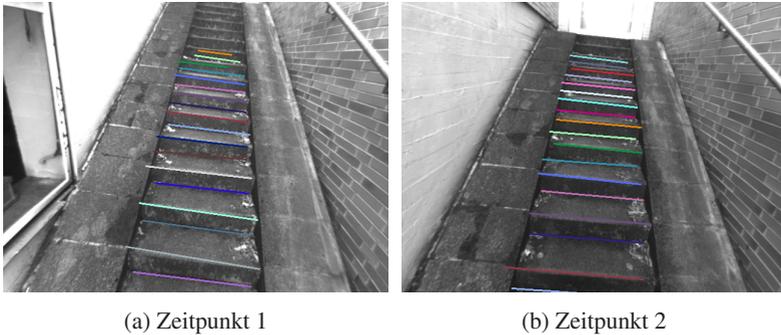


Bild 4.14: Getrackte Liniensegmente einer Treppe. Derselbe Track hat eine zeitlich konsistente Farbe.

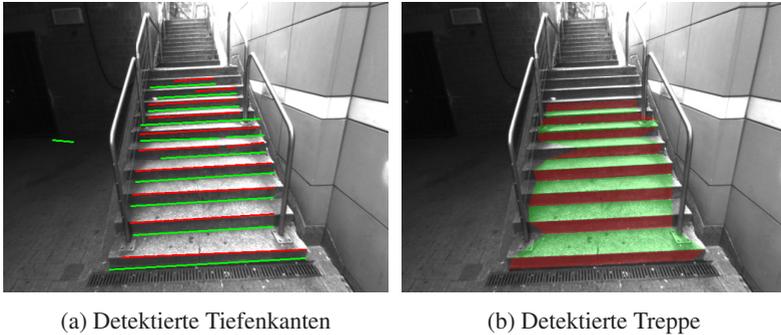
der ROI nicht bekannt ist. Zur Approximation des kürzesten Pfads wird jede ROI in der Bildebene rotiert, so dass die Hauptachse mit der u-Achse des Bildkoordinatensystems ausgerichtet ist. Dadurch wird eine annäherungsweise Ausrichtung der ROI mit der relevanten Tiefenkante erreicht. Danach wird für jede Spalte das Minimum bestimmt, wie durch die roten Punkte in Bild 4.13e visualisiert ist. Eine direkte Lösung des Liniensegments kann nicht direkt durch einen Schätzer der kleinsten Fehlerquadrate oder eines M-Estimators erreicht werden, da oft mehrere Tiefenkanten in den ROIs enthalten sind. Dies kann durch Tiefenkanten an den Stufenaußenseiten, durch Verdeckungen wie Infrastrukturobjekten oder Tiefenartefakte der Fall sein, wie in Bild 4.13d unschwer zu erkennen ist.

Die meisten Treppen sind aus Stufen aufgebaut, die sich durch Liniensegmente oder einer Menge von Liniensegmenten beschreiben lassen. Um diese relevanten Liniensegmente aus den ROIs zu filtern, wird ein iterativer Endpunkt Algorithmus von Ramer et al. [10] verwendet. Das Ergebnis sind für jede ROI die relevanten Liniensegmente, die in Bild 4.13e grün dargestellt sind und den Verlauf der roten Punkte der Kostenminima approximieren. Die grünen Liniensegmente werden abschnittsweise durch einen Schätzer der kleinsten Fehlerquadrate genauer bestimmt.

Als Ergebnis ergeben sich konkave und konvexe Liniensegmente aus den Kostenbildern C_j , siehe Bild 4.15a.

4.4.2.3 Tracken der Liniensegmente

Die konvexen und konkaven Liniensegmente werden über die Zeit verfolgt und in einer Karte mit weltfestem Koordinatensystem gespeichert. Der



(a) Detektierte Tiefenkanten

(b) Detektierte Treppe

Bild 4.15: Detektierte Tiefenkanten und das daraus geschätzte Treppenmodell

Assoziationsschritt kompensiert die Eigenbewegung zwischen zwei aufeinanderfolgenden Bildern durch visuelle Odometrie [18]. Jedes detektierte Liniensegment eines neuen Zeitschritts wird dem räumlich nächsten, existierenden Liniensegment der Karte zugewiesen. Es wird ein vernachlässigbar kleiner Fehler der Eigenbewegungsschätzung angenommen. Es werden gültige von ungültigen Liniensegmenten unterschieden, um nur zeitlich stabile Tracks der Liniensegmente für die Treppendetektion zu verwenden.

Ein Track wird als gültig betrachtet, wenn eine minimale Anzahl von Liniensegmentsdetektionen n_{min} in dem vorausgegangenen Zeitintervall T_p assoziiert wurden. Ansonsten ist der Track ungültig. Die geometrischen Parameter eines gültigen Tracks sind die Position, Länge und Orientierung. Die Parameter aller Tracks werden gefiltert, indem die Parameterwerte aller assoziierten Liniensegmente gemittelt werden.

Neue Tracks werden der Karte hinzugefügt, wenn ein detektiertes Liniensegment keinem existierenden Track assoziiert wird und einen Mindestabstand d_{min} zu allen existierenden Tracks erfüllt. Tracks werden beibehalten, wenn sie den Sichtbereich des Sensors verlassen. Dagegen werden Tracks im Sichtbereich verworfen, welche keine assoziierten Liniensegmente im vorangegangenen Zeitintervall T_p aufweisen. In Bild 4.14 und 4.17e sind getrackte konvexe und konkave Liniensegmente visualisiert. Derselbe Track wird durch eine zeitlich konsistente Farbe gekennzeichnet.

4.4.2.4 Treppenmodellierung

Das Treppenmodell wird aus allen gültigen konvexen und konkaven Tracks der Liniensegment erstellt. Im Allgemeinen können Treppen als eine Abfolge von Stufen beschrieben werden, wobei jede Stufe aus einem Paar eines konvexen und eines konkaven Liniensegments besteht. Zunächst werden einzelne Stufen aus einem Paar eines konvexen und eines konkaven Liniensegments modelliert. Bild 4.9 zeigt ein Liniensegment-Paar, das eine Stufe darstellt. Zweitens werden alle erkannten, aufeinanderfolgenden Stufen zu Treppen kombiniert. Für jeden Stufen- und Treppendetektionsschritt wird eine minimale Anzahl von Merkmalen verwendet, die die Höhe und Ausrichtung der betrachteten Liniensegmente sowie die kürzeste Entfernung zueinander beinhalten.

Im Folgenden sind die Voraussetzungen aufgelistet, damit ein Paar eines konvexen und konkaven Liniensegments als Stufe berücksichtigt wird:

- der Höhenunterschied der Liniensegmente ist kleiner als Δh_{step} ,
- die Orientierungsdifferenz der Liniensegmente ist kleiner als Δo_{step} und
- wenn ein Liniensegment mehreren anderen Liniensegmenten zugeordnet werden kann, werden die Abmessungen und die durchschnittliche Entfernung der infragekommenden Liniensegment berücksichtigt.

Zwei Stufen gehören zu derselben Treppe, wenn folgende Randbedingungen erfüllt sind:

- die Höhendifferenz des konvexen Liniensegments einer Stufe ist gegenüber dem konkaven Liniensegment der anderen Stufe kleiner als Δh_{stair} ,
- die Orientierungsdifferenz dieser zwei Liniensegmente ist kleiner als Δo_{stair} und
- wenn eine Stufe mit mehreren anderen Stufen gepaart werden kann, wird die Stufe gewählt, deren Abmessungen und die durchschnittliche Entfernung der infragekommenden Stufen berücksichtigt.

Das Ergebnis einer solchen Treppenmodellierung ist in Bild 4.15b zu sehen. Alle erkannten Stufen werden als transparente rote Polygone dargestellt. Stufen, die derselben Treppe zugeordnet werden, sind mit transparenten grünen Polygonen verbunden. Alle nicht infragekommenden Liniensegmente aus Bild 4.15a werden als Stufenkandidaten aussortiert, aufgrund von unzureichend passenden Geometrierandbedingungen zu anderen Liniensegmenten oder weil die Liniensegment-Tracks ungültig sind.

4.4.3 Systemkonfiguration

Die vorgeschlagene Treppendetektionsalgorithmus soll sehbehinderte Menschen unterstützen. Ein Prototyp des Systems wird gegenwärtig mithilfe eines Helms realisiert, an dem die Sensoren montiert sind, siehe Bild 4.2b. Ein binokulares Kamerasystem, bestehend aus zwei Point Grey Flea2 Kameras, und eine inertielle Messeinheit (IMU) MTI-300-AHRS von Xsens sind auf dem Helm montiert. Die relevanten Parameter des Kamerasystems können Tabelle 4.1 entnommen werden.

Im laufenden System wird die IMU verwendet, um eine globale Referenzhochachse bereitzustellen, wie in Abschnitt 4.2 beschrieben. Alle Berechnungen werden auf einem Notebook ausgeführt, das von der Person in einem Rucksack getragen wird, die den Helm trägt.

Die dichten Disparitätsbilder werden mithilfe eines Semi-Global Block Matchers berechnet, wie von Hirschmueller et al. eingeführt [24]. Die Normalvektoren werden im Disparitätsraum auf Basis von Integralbildern berechnet, siehe [25] und [26] für weitere Informationen. Die Transformation der Normalenvektoren in Weltkoordinaten wird wie in Formel 2.11 ausgeführt. Die Laufzeiten aller Verarbeitungsschritte kann Tabelle 4.2 entnommen werden.

Verarbeitungsschritt	Laufzeit in [ms]
Rektifizieren und Stereomatchen	38
Visuelle Odometrie	35
Normalenvektorberechnung	40
Oberflächenorientierung	5
Korrelationskosten	5.5
3d Kantendetektion	20
Liniensegment-Tracking	0.11
Treppenmodellierung	0.04

Tabelle 4.2: Laufzeiten der vorgestellten Methode

Durch Pipelining der drei ersten Verarbeitungsschritte, die die komplexesten Berechnungen durchführen, läuft das System bei ca. 10 Hz.

Das Framework ist in C++ realisiert und verwendet ein Standard-Notebook mit einer 2,67-GHz-Dual-Core-CPU, auf der alle Algorithmen berechnet werden.

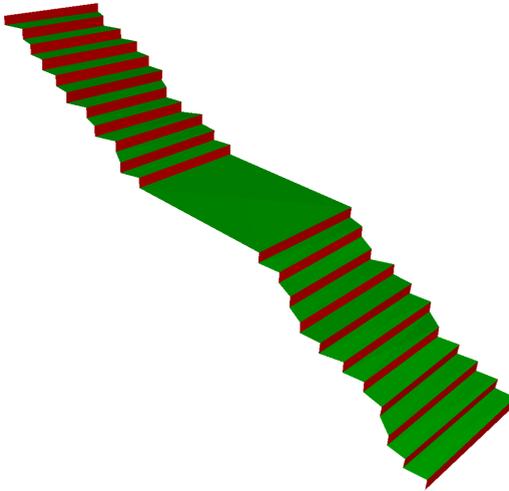


Bild 4.16: Ergebnis des geschätzten globalen 3d-Treppenmodells

4.4.4 Ergebnisse und Auswertung

Das Tracken der Liniensegmente und der Treppendetektionsschritt werden durch folgende Parameter initialisiert:

$$\begin{aligned}n_{min} &= 3 \text{ frames} \\T_p &= 15 \text{ frames} \\d_{min} &= 0.2 \text{ meters} \\ \Delta o_{\text{step}} &= \Delta o_{\text{stair}} = 10^\circ \\ \Delta h_{\text{step}} &= 0.25 \text{ meters} \\ \Delta h_{\text{stair}} &= 0.05 \text{ meters}\end{aligned}$$

Aus Gründen der Robustheit werden nur Liniensegmente für das Tracking verwendet, die einer Stufe aus der Menge der derzeit beobachteten konkaven und konvexen Liniensegmenten zugeordnet werden können.

Diese Einschränkungen sind ausreichend, um Treppen aus den erfassten Paaren von konvexen und konkaven Liniensegmenten zu modellieren. Die Parameter können für andere Treppentypen angepasst werden, um z.B. Wendeltreppen oder Treppen mit größeren Stufenhöhen detektieren zu können.

Die Evaluierung der vorgeschlagenen Methode berücksichtigt aufsteigende Treppen, denen man sich annähert und die überquert werden. Voraussetzung ist,

Abweichungsgröße	Harms [21]	RGB-D [8]	TPRS [43]	SLG [43]
Höhe [cm]	0.12 ± 0.66	1.7 ± 1.4	0.42 ± 0.33	0.68 ± 0.54
Prozent	0.8	8.9	6.0	9.7
Tiefe [cm]	0.24 ± 1.14	1.2 ± 1.6	1.17 ± 0.67	0.9 ± 0.61
Prozent	0.85	4.2	6.5	5.0
Breite [cm]	19.43 ± 10.8	17.3 ± 12.8	3.4 ± 1.95	2.25 ± 1.97
Prozent	10.7	16.5	5.7	3.8
Steigung [°]	0.09 ± 1.28	2.3 ± 1.9		
Ebenen (parallel)			2.22 ± 2.17	1.14 ± 1.13
Ebenen (90°)			4.97 ± 2.13	3.12 ± 1.47

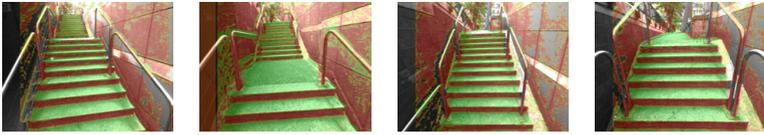
Tabelle 4.3: Fehler der Stufenmodellierung vergleichen den Mittelwert \pm die Standardabweichung der vorgestellten Treppendetektion mit einer auf einer RGB-D Kamera basierten Methode [8], und auf einem Lidar basierten Methoden der Scan-Line Grouping (SLG) und der Two-Point Random Sampling (TPRS) [43].

dass die Treppenstufen konkave und konvexe Liniensegmente aufweisen. Die Treppendimensionen werden auf dem zeitlich gefilterten, globalen Treppenmodell geschätzt. Zur Zeit existiert kein öffentlich verfügbarer Treppendetektionsbenchmark oder Datensatz, der verwendet werden kann, um Ergebnisse auf einer Sequenz zu vergleichen. Der vorgestellte Ansatz zur Treppendetektion wird deshalb an einer typischen Outdoortreppe evaluiert, indem die Höhe, Tiefe, Breite und der Steigungswinkel der Stufen durch den Mittelwert und die Standardabweichung ausgewertet werden. Tabelle 4.3 vergleicht die Ergebnisse der vorgestellten Methode, die auf einem binokularen Kamerasystem basiert, mit drei weiteren Methoden die dem Stand der Technik entsprechen. Eine Vergleichsmethode verwendet eine RGB-D Kamera [8], während die anderen beiden Vergleichsmethoden einen Lidar verwenden [43].

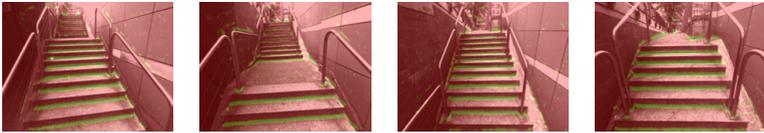
Die Schätzung der Stufenhöhe, -tiefe und der -steigung übertrifft die Genauigkeit aller Ansätze, die dem Stand der Technik entsprechen, siehe Tabelle 4.3. Die Treppenbreite wird größtenteils unterschätzt, da Verdeckungen durch das Treppengeländer und geteilte Liniensegmente bei Tiefendiskontinuitäten auftreten, siehe Bild 4.17d. Die Genauigkeit des vorgestellten Verfahrens wird durch die präzise Detektion des Treppenabsatzes zwischen den zwei Teilen der Treppen unterstrichen, welcher mit einer Tiefe von 2,22 m nur um einen Zentimeter von der wahren Tiefe (2,23 m) abweicht. Ein rekonstruiertes 3d-Modell der Treppe ist in Bild 4.16 dargestellt. Die vorgestellte Treppendetektion verwendet nur rauschanfällige Disparitätsbilder eines binokularen Kamerasystems, im Gegensatz zu anderen Ansätzen, welche genauere Tiefeninformationen aus RGB-D-Kameras [8] oder Laserscannern [43] verwenden.

Die Ergebnisbilder der evaluierten Außentreppe sind in Bild 4.17 dargestellt. Jede Spalte zeigt die gleiche Treppenszene eines Zeitschritts. Visualisiert sind die Orientierungsbilder I_δ , die Kostenbilder C_1 und C_2 , die erkannten konkaven und konvexe Liniensegmente (3d-Kanten), die gültigen konkaven und konvexen Liniensegment-Tracks und das geschätzte Treppenmodell der getrackten Liniensegmente, wie in Abschnitt 4.4.2 eingeführt. In Bild 4.17f werden erkannte Stufen als transparente, rote Polygone gekennzeichnet. Die transparenten, grünen Polygone verbinden Stufen, die zu derselben Treppe gehören. Zu beachten ist, dass alle nicht passenden 3d-Kanten aus Bild 4.17d als Stufenkandidaten verworfen werden. Das Treppenmodell aus den getrackten Liniensegmenten kann vollständig geschätzt werden, auch wenn in Einzelbildern keine Stufen oder nur Teile von Stufen erkannt werden. Insbesondere im Nahfeld werden die getrackten Liniensegmente sehr präzise zusammengeführt. Alle Missassoziationen nicht zusammengehörender Liniensegmente während dem Überqueren der Treppe resultieren in ungültigen Liniensegment-Tracks. Alle detektierten Stufen werden im Nahbereich des Kamerasystems korrekt als zusammengehörende Treppe detektiert. In Spalte zwei aus Bild 4.17d wird die detektierte Treppe korrekt durch den vorhandenen Treppenabsatz getrennt.

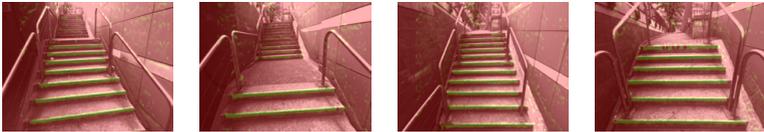
Ergebnisbilder weiterer Treppen sind in Bild 4.18 und 4.19 zu finden. Es können verschiedene Treppentypen detektiert werden. Auch leichte Verdeckungen der konkaven Stufenkanten durch Blätter bereiten der vorgestellten Treppendetektion keine Probleme, siehe zweite Spalte in Bild 4.18. Die Invarianz der vorgestellten Treppendetektion gegen Rotationen kann in den beiden rechten Spalten aus Bild 4.18 und der linken Spalte in Bild 4.19 erkannt werden. Sehr breite Treppenstufen weisen oft Tiefenartefakte entlang der Liniensegmente auf. Dies führt zur Teilung der geschätzten Liniensegmente und damit zu einer Unterschätzung der Treppenbreite (siehe beide mittleren Spalten in Bild 4.19). Dieses Problem kann durch ein zusätzliches Zusammenfügen mehrerer geschätzter Liniensegmente umgegangen werden. Durch sehr starke Rauschartefakte werden die geschätzten Orientierungen der Liniensegmente massiv gestört und bereiten der Treppenmodellierung aus mehreren Stufen Probleme (siehe rechte Spalte in Bild 4.19). Trotzdem werden alle Stufen detektiert. Dieser Fall kann durch eine variable Orientierungsabweichung bei der Treppenmodellierung, abhängig von der Stärke des Tiefenrauschens, gelöst werden.



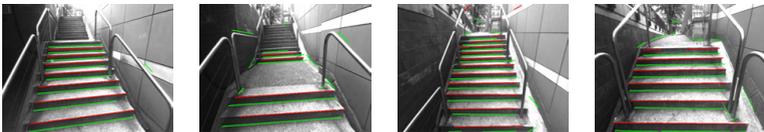
(a) Orientierungsbilder I_δ



(b) Kostenbilder C_1



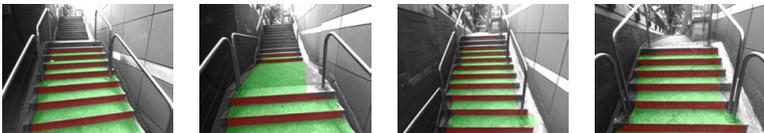
(c) Kostenbilder C_2



(d) Detektierte 3d-Kanten: konkave Kanten sind grün, konvexe Kanten rot visualisiert.



(e) Getrackte Liniensegmente, dieselbe Farbe kennzeichnet gleiche Liniensegmente.

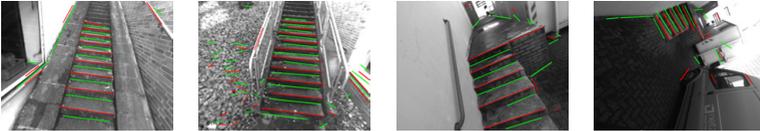


(f) Geschätztes Treppenmodell. Rote Polygone sind erkannte Stufen, grüne Polygone verbinden Stufen die zu derselben Treppe gehören.

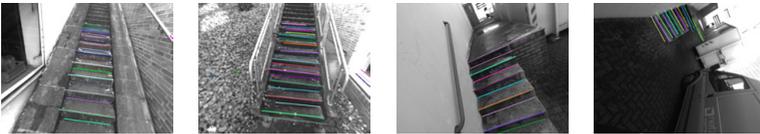
Bild 4.17: Ergebnisse der vorgestellten Treppendetektion der evaluierten Treppensequenz



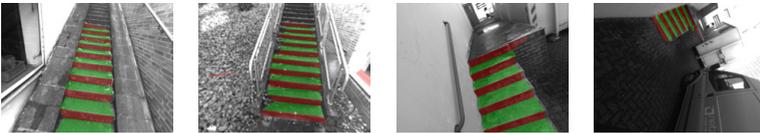
(a) Orientierungsbilder I_δ



(b) Detektierte konkave (grün) und konvexe (rot) 3d-Kanten

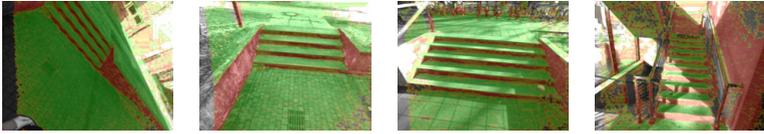


(c) Getrackte Liniensegmente

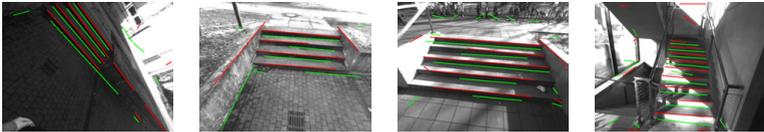


(d) Geschätztes Treppenmodell

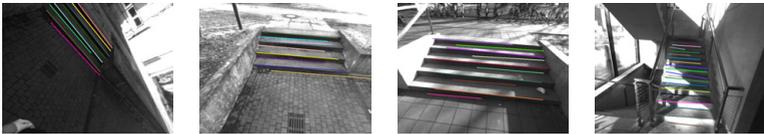
Bild 4.18: Ergebnisbilder der vorgestellten Treppendetektion weiterer Treppensequenzen



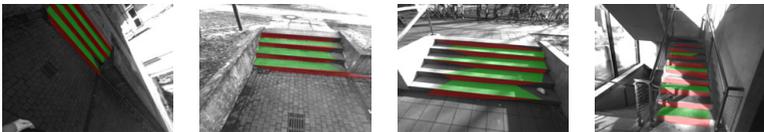
(a) Orientierungsbilder I_δ



(b) Detektierte konkave (grün) und konvexe (rot) 3d-Kanten



(c) Getrackte Liniensegmente



(d) Geschätztes Treppenmodell

Bild 4.19: Ergebnisbilder der vorgestellten Treppendetektion weiterer Treppensequenzen

5 Zusammenfassung und Ausblick

Die vorliegende Arbeit beschreibt eine Genauigkeitsuntersuchung von Normalenvektoren, die aus verrauschten Disparitätsbildern binokularer Kamerasysteme geschätzt werden. Die Analyse der binokularen Normalenvektoren liefert eine Beschreibung der Orientierungsgenauigkeit durch ein Konfidenzintervall, das lediglich vom betrachteten Oberflächenpunkt, der Orientierung des Normalenvektors sowie der Größe des Bildausschnitts abhängt. Diese Abhängigkeiten wurden genutzt, um zwei Methoden zur Auslegung der Normalenvektorberechnung zu entwickeln, welche die Größe des Bildausschnitts und die Orientierungsgenauigkeit der Normalenvektorschätzung optimal bestimmen. Die Methoden beruhen auf Monte-Carlo-Simulationen, welche im Vergleich zur Linearisierung das charakteristische Verhalten von Normalenvektoren deutlich präziser abbilden. Ein Vergleich der Orientierungsgenauigkeit aus der Monte-Carlo-Simulation mit aus Experimenten gewonnenen Realdaten zeigt, dass die Monte-Carlo-Simulation das Verhalten der Normalenvektoren mit einer guten Genauigkeit wiedergibt. Zudem wurde eine allgemeingültige Transformation von Normalenvektoren zwischen dem Disparitätsraum und Weltkoordinaten hergeleitet.

In der Arbeit wurden auf dieser Basis zwei neuartige Realisierungen zur Umfeldwahrnehmung vorgestellt, eine Freibereichsschätzung und eine Treppendetektion, die nur Disparitätsbilder und Orientierungsinformationen benötigen. Beide Anwendungen verwenden eine in dieser Arbeit entwickelte Methode, um die Orientierungsgenauigkeiten der geschätzten Normalenvektoren für eine gegebene Patchgröße einzuhalten. Dadurch kann die Funktion der Anwendungen sichergestellt werden.

Die vorgestellte Freibereichsschätzung verwendet als Freibereichsbegrenzung zeitlich gefilterte Hindernis- und Bordsteininformationen aus einem mehrschichtigen, stochastischen Belegungsgitter. Die zusätzliche Berücksichtigung der Orientierungsinformation zur Bordsteinerkennung stellt eine sinnvolle Ergänzung zur Freibereichsschätzung dar. Die Ergebnisse zeigen zudem die Vorteile einer nicht-parametrischen Schätzung der Bordstein- und Hindernisbereiche. Die Freibereichsschätzung ist ausgelegt bis zu einer Entfernung von 30 Metern, in wel-

cher auch hohe Bordsteine bei einer ausreichenden Qualität der Tiefeninformation detektiert werden. Die strahlenbasierte Freibereichsschätzung im Belegungs-gitter funktioniert robust in unterschiedlichen Umgebungen. Sie hat den Vorteil, um weitere Sensoren erweiterbar zu sein, die andere Sichtbereiche abdecken können. Da die vorgestellte Methode rein auf Tiefendaten basiert, kann das binokulare Kamerasystem durch einen anderen Tiefensensor ersetzt werden.

Die vorgestellte Treppendetektion ist invariant gegen Rotationen des Kame-rasystems und basiert wiederum nur auf Tiefendaten. Ein Optimalfilter verwendet die Oberflächenorientierungen, um konvexe und konkave Liniensegmente in der beobachteten Szene zu schätzen. Diese Liniensegmente werden über die Zeit gefiltert und zu Stufen kombiniert, die weiter zu Treppen zusammengefügt werden. Die experimentellen Ergebnisse zeigen, dass die vorgestellte Methode eine robuste Detektion von Treppen erlaubt. Wurde die Treppe überquert, liegt ein hochpräzises, geschätztes Treppenmodell vor, das die Genauigkeit aller aus der Literatur bekannten Methoden zur Treppendetektion übertrifft. Falschdetektionen an Zebrastreifen oder Schattenbereichen treten bei den Methoden auf, die Li-niensegmente aus Intensitätsbildern schätzen. Diese Falschdetektionen können mit dem in dieser Arbeit gewählten Ansatz komplett vermieden werden, da nur Tiefen- und Orientierungsdaten verwendet werden. Der vorgestellte Ansatz ist modular und ermöglicht die Verwendung einer RGB-D Kamera anstatt des bin-okularen Kamerasystems.

Aufgrund der vielversprechenden Ergebnisse der Genauigkeitsuntersuchung und der beiden Anwendungen zur Umweltwahrnehmung ist die zukünftige Integra-tion folgender Erweiterungen in die bestehenden Methoden denkbar:

- Die Genauigkeitsuntersuchung binokularer Normalenvektoren kann in zu-künftigen Versionen erweitert werden, so dass
 - unsicherheitsbehaftete Kalibrierparameter,
 - das Rauschverhalten unterschiedlicher Stereomatcher in variablen Umgebungen und
 - weitere Kameramodellemit berücksichtigt werden können.

- Zeitlich konsistente Tiefenartefakte führen bei beiden Anwendungen zu Herausforderungen, da beide Anwendungen nur Informationen nutzen, die direkt oder indirekt aus Disparitätsbildern ermittelt werden. Diese Tiefen-artefakte treten in den Disparitätsbildern heutiger Stereomatcher in unter-oder überbelichteten Bildbereichen auf, welche in den Anwendungen se-perat behandelt werden könnten.

- Die Freibereichsschätzung könnte durch zusätzliche Tiefensensoren mit überlappendem und erweitertem Sichtbereich das Potential einer robusteren Rundumschätzung des Freibereichs erschließen.
- Die Referenzhochachse der Treppendetektion des verwendeten Sensorsystems (IMU) könnte durch eine getrackte Oberflächennormale der Grundebene ersetzt werden, so dass die Methode nur einen Tiefensensor benötigt.
- Die Treppendetektion könnte zusätzlich für absteigende Treppen erweitert werden, indem das Optimalfilter für Sprünge in der Tiefe anpasst wird.

A Anhang

A.1 Transformation von Normalenvektoren zwischen Welt- und Bildkoordinaten

Der Oberflächenpunkt $\mathbf{p}_0 = (u_0, v_0, d_0)^\top$ und der zugehörige Normalenvektor $\mathbf{n}_{\text{disp}} = (n_u, n_v, n_d)^\top$ repräsentieren eine Ebene im Disparitätsraum. Im Folgenden wird der Normalenvektor $\mathbf{n}_{\text{world}} = (n_x, n_y, n_z)^\top$ am rekonstruierten Oberflächenpunkt $\mathbf{x}_0 = r(\mathbf{p}_0)$ in Weltkoordinaten bestimmt. Dafür werden die beiden Oberflächenpunkte

$$\mathbf{p}_1 = \mathbf{p}_0 + \begin{pmatrix} -n_v \\ n_u \\ 0 \end{pmatrix} \quad (\text{A.1})$$

und

$$\mathbf{p}_2 = \mathbf{p}_0 + \begin{pmatrix} 0 \\ n_d \\ -n_v \end{pmatrix}. \quad (\text{A.2})$$

ausgewählt. Die folgende Herleitung der allgemeingültigen Transformation kann auch mit anderen Ebenenpunkten gezeigt werden.

Wie man direkt sehen kann, liegen die Punkte \mathbf{p}_0 , \mathbf{p}_1 und \mathbf{p}_2 in einer Ebene im Disparitätsraum, die auch durch den Normalenvektor \mathbf{n}_{disp} an der Stelle \mathbf{p}_0 beschrieben wird.

Die Punkte \mathbf{p}_0 , \mathbf{p}_1 und \mathbf{p}_2 werden mithilfe der Funktion r aus Gleichung 2.8 in Weltkoordinaten rekonstruiert:

$$\mathbf{x}_i = r(\mathbf{p}_i), i \in \{0, 1, 2\}. \quad (\text{A.3})$$

Da Ebenen aus dem Disparitätsraum auch auf Ebenen in Weltkoordinaten abgebildet werden, liegen die Punkte \mathbf{x}_0 , \mathbf{x}_1 und \mathbf{x}_2 exakt auf der Ebene in Weltkoordinaten, die durch \mathbf{p}_0 und \mathbf{n}_{disp} im Disparitätsraum definiert ist. Der zugehörige

Normalenvektor in Weltkoordinaten kann nun durch das folgende Kreuzprodukt berechnet werden:

$$\mathbf{n}_{\text{world}} = (\mathbf{x}_1 - \mathbf{x}_0) \times (\mathbf{x}_2 - \mathbf{x}_0). \quad (\text{A.4})$$

Für die weitere Verwendung wird die Funktion, die im Disparitätsraum einen beliebigen Oberflächenpunkt $\mathbf{p} = (u, v, d)^\top$ mit dem dazugehörigen Normalenvektor $\mathbf{n}_{\text{disp}} = (n_u, n_v, n_d)^\top$ in einen Normalenvektor $\mathbf{n}_{\text{world}}$ in Weltkoordinaten transformiert als $f : \mathbb{R}^3 \times \mathbb{R}^3 \rightarrow \mathbb{R}^3$ bezeichnet.

Dazu werden Gleichung A.1 bis A.3 in Gleichung A.4 eingesetzt und resultieren in

$$\mathbf{n}_{\text{world}} = f(\mathbf{p}, \mathbf{n}_{\text{disp}}) = \begin{pmatrix} Fn_u \\ Fn_v \\ (c_u - u)n_u + (c_v - v)n_v - dn_d \end{pmatrix}. \quad (\text{A.5})$$

Umgekehrt gilt für einen beliebigen Oberflächenpunkt $\mathbf{x} = (x, y, z)^\top$ mit dem dazugehörigen Normalenvektor $\mathbf{n}_{\text{world}} = (n_x, n_y, n_z)^\top$ in Weltkoordinaten die Funktion g um den Normalenvektor \mathbf{n}_{disp} im Disparitätsraum zu bestimmen:

$$\mathbf{n}_{\text{disp}} = g(\mathbf{x}, \mathbf{n}_{\text{world}}) = \begin{pmatrix} Bn_x \\ Bn_y \\ -xn_x - yn_y - zn_z \end{pmatrix} \quad (\text{A.6})$$

A.2 Berechnung der Jacobi-Matrix

In dieser Arbeit wird zur Abschätzung funktionaler Zusammenhänge eine Linearisierung verwendet. Die Linearisierung einer nichtlinearen, mehrdimensionalen und an einem Arbeitspunkt \mathbf{x}_0 differenzierbaren Funktion $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ wird durch die Jacobi-Matrix approximiert:

$$\mathbf{J}_{\mathbf{x}_0} = \mathbf{J}_{f, \mathbf{x}} \Big|_{\mathbf{x}_0} = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \Big|_{\mathbf{x}_0} = \left[\begin{array}{ccc} \frac{\partial f_1(\mathbf{x})}{\partial x_1} & \dots & \frac{\partial f_1(\mathbf{x})}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m(\mathbf{x})}{\partial x_1} & \dots & \frac{\partial f_m(\mathbf{x})}{\partial x_n} \end{array} \right] \Big|_{\mathbf{x}_0} \quad (\text{A.7})$$

Die lineare Näherung des Funktionsverlaufs am Arbeitspunkt \mathbf{x}_0 kann nun durch die Taylorreihe erster Ordnung dargestellt werden:

$$f(\mathbf{x}) \approx f(\mathbf{x}_0) + \mathbf{J}_{\mathbf{x}_0}(\mathbf{x} - \mathbf{x}_0) \quad (\text{A.8})$$

A.3 Hauptkomponentenanalyse

Eine Hauptkomponentenanalyse (engl.: Principal Components Analysis (PCA)) ist eine orthogonale Transformation, die die Hauptausrichtungen einer n-dimensionalen Datenmenge bestimmt. Das Ergebnis sind n Eigenvektoren und n dazugehörige Eigenwerte. Der Eigenvektor mit dem höchsten Eigenwert beschreibt die dominierende Hauptrichtung der Datenmenge.

Als Beispiel wird eine 3d-Punktwolke mit m Beobachtungen $\mathbf{x}^T = (x, y, z)$ betrachtet, deren Eigenvektoren \mathbf{v} und Eigenwerte λ bestimmt werden sollen.

Zunächst wird die Matrix \mathbf{S} bestimmt:

$$\mathbf{S} = \sum_{i=1}^m (\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^T \quad (\text{A.9})$$

mit dem Mittelwert

$$\boldsymbol{\mu} = \frac{1}{m} \sum_{i=1}^m \mathbf{x}_i \quad (\text{A.10})$$

Die Eigenwerte und Eigenvektoren können durch

$$\det(\lambda \mathbf{I} - \mathbf{S}) = 0 \quad (\text{A.11})$$

bestimmt werden.

Repräsentiert die 3d-Punktwolke eine annähernd ebene Fläche, dann spannen die Eigenvektoren mit den beiden größten Eigenwerten die ebene Fläche auf. Der Eigenvektor mit dem kleinsten Eigenwert entspricht dem Normalenvektor der betrachteten Fläche.

A.4 Bordsteine

Die in dieser Arbeit vorgestellte Freiflächenschätzung wird durch mittelhohe Bordsteine und größere Objekte begrenzt. In Deutschland gibt es keine Regelung, die Bordsteine als Straßenbegrenzung im urbanen Raum vorschreiben. Die

Höhe von Bordsteinen ist nicht vereinheitlicht und wird in den Richtlinien für Stadtstraßen 2006 in drei Kategorien unterschieden:

- niedrige Bordsteine (bis 4 cm Höhe)
- halbhohle Bordsteine (4 - 10 cm Höhe)
- hohe Bordsteine (10 - 14 cm Höhe)

Niedrigen Bordsteine sind allgemein als abgesenkte Bordsteine bekannt und werden an Stellen wie Einfahrten oder auch Fahrradwegeinmündungen verbaut. Halbhohle und hohe Bordsteine werden als Begrenzung zwischen Fahrbahn bzw. Parkstreifen und Gehweg bzw. Radweg verwendet. Hohe Bordsteine finden vor allem entlang von Hauptverkehrsstraßen Verwendung und sind häufig bis zu 30 cm hoch.

Allgemein können alle Bordsteinformen im innerstädtischen Bereich als Fahrbahnbegrenzung auftreten, da es keine baurechtlichen Regulierungen gibt. Insbesondere durch altersbedingte Absenkungen, Anhebungen oder Verkippungen der Bordsteine sind verschiedene Bordsteinhöhen zu erwarten.

A.5 Automatisierungsgrade von Fahrzeugen

Nomenklatur	Fahraufgaben des Fahrers nach Automatisierungsgrad	Automatisierungsgrad
Vollautomatisiert	<p>Das System übernimmt Quer- und Längsführung vollständig in einem definierten Anwendungsfall</p> <ul style="list-style-type: none"> • Der Fahrer muss das System dabei nicht überwachen • Vor dem Verlassen des Anwendungsfalles fordert das System den Fahrer mit ausreichender Zeitreserve zur Übernahme der Fahraufgabe auf • Erfolg dies nicht, wird in den risikominimalen Systemzustand zurückgeführt • Systemgrenzen werden alle vom System erkannt, das System ist in allen Situationen in der Lage, in den risikominimalen Systemzustand zurückzuführen 	
Hochautomatisiert	<p>Das System übernimmt Quer- und Längsführung für einen gewissen Zeitraum in spezifischen Situationen</p> <ul style="list-style-type: none"> • Der Fahrer muss das System dabei nicht überwachen • Bei Bedarf wird der Fahrer zur Übernahme der Fahraufgabe mit ausreichender Zeitreserve aufgefordert • Systemgrenzen werden alle vom System erkannt. Das System ist nicht in der Lage, aus jeder Ausgangssituation den risikominimalen Zustand herbeizuführen 	
Teilautomatisiert	<p>Das System übernimmt Quer- und Längsführung (für einen gewissen Zeitraum oder/und in spezifischen Situationen)</p> <ul style="list-style-type: none"> • Der Fahrer muss das System dauerhaft überwachen • Der Fahrer muss jederzeit zur vollständigen Übernahme der Fahrzeugführung bereit sein 	
Assistiert	<p>Fahrer führt dauerhaft entweder die Quer- oder die Längsführung aus. Die jeweils andere Fahraufgabe wird in gewissen Grenzen vom System ausgeführt</p> <ul style="list-style-type: none"> • Der Fahrer muss das System dauerhaft überwachen • Der Fahrer muss jederzeit zur vollständigen Übernahme der Fahrzeugführung bereit sein 	
Driver only	<p>Fahrer führt dauerhaft (während der gesamten Fahrt) die Längsführung (Beschleunigen/Verzögern) und die Querführung (lenken) aus</p>	

Bild A.1: Definition der Fahrzeugautomatisierungsgrade nach dem Bundesamt für Straßenwesen (BASt) [16]

Literaturverzeichnis

- [1] *ECCV 2014 Workshops – Assistive Computer Vision and Robotics*, 2014.
- [2] J. M. Alvarez, T. Gevers, Y. LeCun, and A. M. Lopez, “Road scene segmentation from a single image,” in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 376–389.
- [3] N. Amenta, M. Bern, and M. Kamvysselis, “A new voronoi-based surface reconstruction algorithm,” in *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*. ACM, 1998, pp. 415–421.
- [4] H. Badino, U. Franke, and R. Mester, “Free space computation using stochastic occupancy grids and dynamic programming,” in *Workshop on Dynamical Vision, ICCV, Rio de Janeiro, Brazil*, vol. 20. Citeseer, 2007.
- [5] H. Badino, D. Huber, Y. Park, and T. Kanade, “Fast and accurate computation of surface normals from range images,” in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 3084–3091.
- [6] J. Beck and C. Stiller, “Non-parametric lane estimation in urban environments,” in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*. IEEE, 2014, pp. 43–48.
- [7] J. E. Bresenham, “Algorithm for computer control of a digital plotter,” *IBM Systems journal*, vol. 4, no. 1, pp. 25–30, 1965.
- [8] J. A. Delmerico, D. Baran, P. David, J. Ryde, and J. J. Corso, “Ascending stairway modelling from dense depth imagery for traversability analysis,” in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 2283–2290.
- [9] E. W. Dijkstra, “A note on two problems in connexion with graphs,” *Numerische mathematik*, vol. 1, no. 1, pp. 269–271, 1959.
- [10] D. H. Douglas and T. K. Peucker, “Algorithms for the reduction of the number of points required to represent a digitized line or its caricature,”

Cartographica: The International Journal for Geographic Information and Geovisualization, vol. 10, no. 2, pp. 112–122, 1973.

- [11] R. O. Duda and P. E. Hart, “Use of the hough transformation to detect lines and curves in pictures,” *Communications of the ACM*, vol. 15, no. 1, pp. 11–15, 1972.
- [12] A. Elfes, “Using occupancy grids for mobile robot perception and navigation,” *Computer*, vol. 22, no. 6, pp. 46–57, 1989.
- [13] M.ENZWEILER, P. GREINER, C. KNOPPEL, and U. FRANKE, “Towards multi-cue urban curb recognition,” in *Intelligent Vehicles Symposium (IV), 2013 IEEE*. IEEE, 2013, pp. 902–907.
- [14] O. FAUGERAS, *Three dimensional computer vision: A geometric viewpoint*. the MIT Press, 1993.
- [15] C. FERNÁNDEZ, R. IZQUIERDO, D. LLORCA, and M. SOTELO, “Road curb and lanes detection for autonomous driving on urban scenarios,” in *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*. IEEE, 2014, pp. 1964–1969.
- [16] T. M. GASSER, C. ARZT, M. AYOUBI, A. BARTELS, L. BÜRKLE, J. EIER, F. FLEMISCH, D. HÄCKER, T. HESSE, W. HUBER *et al.*, “Rechtsfolgen zunehmender fahrzeugautomatisierung,” *Berichte der Bundesanstalt für Straßenwesen. Unterreihe Fahrzeugtechnik*, no. 83, 2012.
- [17] A. GEIGER, P. LENZ, C. STILLER, and R. URTASUN, “Vision meets robotics: The kitti dataset,” *International Journal of Robotics Research (IJRR)*, 2013.
- [18] A. GEIGER, M. ROSER, and R. URTASUN, “Efficient large-scale stereo matching,” in *Asian Conference on Computer Vision (ACCV)*, 2010.
- [19] A. GEIGER, J. ZIEGLER, and C. STILLER, “Stereoscan: Dense 3d reconstruction in real-time,” in *Intelligent Vehicles Symposium (IV)*, 2011.
- [20] H. HARMS, E. REHDER, and M. LAUER, “Grid map based free space estimation using stereo vision,” in *1st Workshop on Environment Perception for Automated On-road Vehicles, IEEE Intelligent Vehicles Symposium, 2015*. IEEE, 2015.
- [21] H. HARMS, E. REHDER, T. SCHWARZE, and M. LAUER, “Detection of ascending stairs using stereo vision,” in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 2496–2502.

-
- [22] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.
- [23] H. Hirschmuller, “Accurate and efficient stereo processing by semi-global matching and mutual information,” in *Computer Vision and Pattern Recognition (CVPR)*, vol. 2. IEEE, 2005, pp. 807–814.
- [24] H. Hirschmuller, “Stereo processing by semiglobal matching and mutual information,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 30, no. 2, pp. 328–341, 2008.
- [25] D. Holz, S. Holzer, R. B. Rusu, and S. Behnke, “Real-time plane segmentation using rgb-d cameras,” in *RoboCup 2011: Robot Soccer World Cup XV*. Springer, 2012, pp. 306–317.
- [26] S. Holzer, R. B. Rusu, M. Dixon, S. Gedikli, and N. Navab, “Adaptive neighborhood selection for real-time surface normal estimation from organized point cloud data using integral images,” in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 2684–2689.
- [27] X. Hu, S. Rodrigues, and A. Gepperth, “A multi-modal system for road detection and segmentation,” in *IEEE Symposium on Intelligent Vehicles*, 2014.
- [28] A. E. Johnson and M. Hebert, “Using spin images for efficient object recognition in cluttered 3d scenes,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 21, no. 5, pp. 433–449, 1999.
- [29] G. Kamberova and R. Bajcsy, “Sensor errors and the uncertainties in stereo reconstruction,” in *Workshop on Empirical Evaluation Methods in Computer Vision, Santa Barbara, California*. Citeseer, 1998.
- [30] M. Kazhdan, M. Bolitho, and H. Hoppe, “Poisson surface reconstruction,” in *Proceedings of the fourth Eurographics symposium on Geometry processing*, 2006.
- [31] M. Kellner, U. Hofmann, M. E. Bouzouraa, H. Kasper, and S. Neumaier, “Laserscanner based road curb feature detection and efficient mapping using local curb descriptions,” in *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*. IEEE, 2014, pp. 2602–2609.

- [32] T. Kuhn and J. Fritsch, “Visio-spatial road boundary detection for unmarked urban and rural roads,” in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*. IEEE, 2014, pp. 1251–1256.
- [33] H. Lategahn, A. Geiger, and B. Kitt, “Visual slam for autonomous ground vehicles,” in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1732–1737.
- [34] D. F. Llorca, M. A. Sotelo, I. Parra, M. Ocaña, and L. M. Bergasa, “Error analysis in a stereo vision-based pedestrian detection sensor for collision avoidance applications,” *Sensors*, vol. 10, no. 4, pp. 3741–3758, 2010.
- [35] W. E. Lorensen and H. E. Cline, “Marching cubes: A high resolution 3d surface construction algorithm,” in *ACM Siggraph Computer Graphics*, vol. 21, no. 4. ACM, 1987, pp. 163–169.
- [36] X. Lu and R. Manduchi, “Detection and localization of curbs and stairways using stereo vision,” in *IEEE International conference on Robotics and Automation*, vol. 4. Citeseer, 2005, p. 4648.
- [37] A. Luber, “Ein generisches abbildungsmodell für stereokamerasysteme,” Ph.D. dissertation, Humboldt-Universität zu Berlin, Mathematisch-Naturwissenschaftliche Fakultät II, 2015.
- [38] F. Moosmann, O. Pink, and C. Stiller, “Segmentation of 3d lidar data in non-flat urban environments using a local convexity criterion,” in *Intelligent Vehicles Symposium, 2009 IEEE*. IEEE, 2009, pp. 215–220.
- [39] D. Murray and J. J. Little, “Environment modeling with stereo vision,” in *Intelligent Robots and Systems (IROS). Proceedings.*, vol. 3. IEEE, 2004, pp. 3116–3122.
- [40] D. O. North, “An analysis of the factors which determine signal/noise discrimination in pulsed-carrier systems,” *Proceedings of the IEEE*, vol. 51, no. 7, pp. 1016–1027, 1963.
- [41] B. Oehler, J. Stueckler, J. Welle, D. Schulz, and S. Behnke, “Efficient multi-resolution plane segmentation of 3d point clouds,” in *Intelligent Robotics and Applications*. Springer, 2011, pp. 145–156.
- [42] F. Oniga and S. Nedeveschi, “Polynomial curb detection based on dense stereovision for driving assistance,” in *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*. IEEE, 2010, pp. 1110–1115.

-
- [43] S. Oßwald, J.-S. Gutmann, A. Hornung, and M. Bennewitz, “From 3d point clouds to climbing stairs: A comparison of plane segmentation approaches for humanoids,” in *Humanoid Robots, 2011 11th IEEE-RAS International Conference on*. IEEE, 2011, pp. 93–98.
- [44] N. Otsu, “A threshold selection method from gray-level histograms,” *Automatica*, vol. 11, no. 285-296, pp. 23–27, 1975.
- [45] D. Pfeiffer and U. Franke, “Efficient representation of traffic scenes by means of dynamic stixels,” in *Intelligent Vehicles Symposium (IV), 2010 IEEE*. IEEE, 2010, pp. 217–224.
- [46] V. Pradeep, G. Medioni, J. Weiland *et al.*, “Piecewise planar modeling for step detection using stereo vision,” in *Workshop on computer vision applications for the visually impaired*, 2008.
- [47] A. Pérez-Yus, G. López-Nicolás, and J. J. Guerrero, “Detection and modeling of staircases using a wearable depth sensor,” in *ECCV 2014 Workshops - Assistive Computer Vision and Robotics*, 2014.
- [48] B. Ranft and T. Strauß, “Modeling arbitrarily oriented slanted planes for efficient stereo vision based on block matching,” in *Intelligent Transportation Systems (ITSC)*. IEEE, 2014, pp. 1941–1947.
- [49] I. B. Rejeb, “Development of an algorithm for staircase detection using stereo vision,” 2014.
- [50] Z. Ren, J. Yuan, and Z. Zhang, “Robust hand gesture recognition based on finger-earth mover’s distance with a commodity depth camera,” in *Proceedings of the 19th ACM international conference on Multimedia*. ACM, 2011, pp. 1093–1096.
- [51] M. Roser, *Modellbasierte und positionsgenaue Erkennung von Regentropfen in Bildfolgen zur Verbesserung von videobasierten Fahrerassistenzfunktionen*. KIT Scientific Publishing, 2012, vol. 21.
- [52] J. Salvi, C. Matabosch, D. Fofi, and J. Forest, “A review of recent range image registration methods with accuracy evaluation,” *Image and Vision Computing*, vol. 25, no. 5, pp. 578–596, 2007.
- [53] V. Schomerus, D. Rosebrock, and F. M. Wahl, “Camera-based lane border detection in arbitrarily structured environments,” in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*. IEEE, 2014, pp. 56–63.

- [54] S. Se and M. Brady, “Vision-based detection of staircases,” in *Fourth Asian Conference on Computer Vision ACCV*, vol. 1, 2000, pp. 535–540.
- [55] A. Seibert, M. Hahnel, A. Tewes, and R. Rojas, “Camera based detection and classification of soft shoulders, curbs and guardrails,” in *Intelligent Vehicles Symposium (IV), 2013 IEEE*. IEEE, 2013, pp. 853–858.
- [56] P. Y. Shinzato, D. Gomes, and D. F. Wolf, “Road estimation with sparse 3d points from stereo data,” in *Intelligent Transportation Systems (ITSC)*. IEEE, 2014, pp. 1688–1693.
- [57] P. Y. Shinzato, D. F. Wolf, and C. Stiller, “Road terrain detection: Avoiding common obstacle detection assumptions using sensor fusion,” in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*. IEEE, 2014, pp. 687–692.
- [58] G. Sibley, L. Matthies, and G. Sukhatme, *Bias reduction and filter convergence for long range stereo*. Springer, 2007.
- [59] C. Stiller, *Vorlesungsmanuskript zu Grundlagen der Mess- und Regelungstechnik*, Institut für Mess- und Regelungstechnik (MRT), Karlsruher Institut für Technologie (KIT), 2006.
- [60] C. Stiller, *Vorlesungsmanuskript zu Messtechnik II*, Institut für Mess- und Regelungstechnik (MRT), Karlsruher Institut für Technologie (KIT), 2010.
- [61] K. Storjohann, *Perspektive Kartierungen: funktionelle Konsequenzen für die visuelle Raumwahrnehmung*, ser. Fortschritt-Berichte VDI. VDI Verlag, 1989. <https://books.google.de/books?id=uYapmwEACAAJ>
- [62] T. Strauss, *Kalibrierung von Multi-Kamera-Systemen*. KIT Scientific Publishing, 2015, vol. 33.
- [63] S. Suzuki *et al.*, “Topological structural analysis of digitized binary images by border following,” *Computer Vision, Graphics, and Image Processing*, vol. 30, no. 1, pp. 32–46, 1985.
- [64] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.
- [65] G. B. Vitor, A. C. Victorino, and J. V. Ferreira, “Comprehensive performance analysis of road detection algorithms using the common urban Kitti-road benchmark,” in *Workshop on IEEE Intelligent Vehicle Symposium (IV)*, Dearborn, Michigan, United States, Jun. 2014, pp. 19 – 24. <https://hal.archives-ouvertes.fr/hal-01089094>

- [66] F. Wahl, *Digitale Bildsignalverarbeitung: Grundlagen, Verfahren, Beispiele*, ser. Nachrichtentechnik (Springer-Verlag). Springer, 1984. <https://books.google.de/books?id=AXNgygAACAAJ>
- [67] B. Wang, V. Fremont, and S. A. Rodriguez Florez, “Color-Based Road Detection and Its Evaluation on the KITTI Road Benchmark,” in *2014 IEEE Intelligent Vehicles Symposium*, United States, Jun. 2014, p. 6 p. <https://hal.archives-ouvertes.fr/hal-01010998>
- [68] R. P. Wildes, “Direct recovery of three-dimensional scene geometry from binocular stereo disparity,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 8, pp. 761–774, 1991.
- [69] L. B. Wolff and T. E. Boult, “Using line correspondence stereo to measure surface orientation.” in *IJCAI*, 1989, pp. 1655–1660.
- [70] C. Zhong, Y. Zhuang, and W. Wang, “Stairway detection using gabor filter and fpg,” in *Soft Computing and Pattern Recognition (SoCPaR), 2011 International Conference of*. IEEE, 2011, pp. 578–582.
- [71] J. Ziegler and C. Stiller, “Spatiotemporal state lattices for fast trajectory planning in dynamic on-road driving scenarios,” in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, 2009, pp. 1879–1884.

Veröffentlichungen des Autors

Hannes Harms, Eike Rehder, Tobias Schwarze, and Martin Lauer. Detection of ascending stairs using stereo vision. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 2496–2502. IEEE, 2015.

Hannes Harms, Eike Rehder, and Martin Lauer. Grid map based free space estimation using stereo vision. In *1st Workshop on Environment Perception for Automated On-road Vehicles, IEEE Intelligent Vehicles Symposium, 2015*. IEEE, 2015.

Hannes Harms, Johannes Beck, Julius Ziegler, and Christoph Stiller. Accuracy analysis of surface normal reconstruction in stereo vision. In *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, pages 730–736. IEEE, 2014.

J Nagel, C Beck, H Harms, P Stiller, H Guth, G Bretthauer, and R Guthoff. Energie- und Speichereffiziente Berechnung des Akkommodationsbedarfs im Künstlichen Akkommodationssystem. *Klinische Monatsblätter der Augenheilkunde*, 227(12):930–934, 2010.

J Nagel, H Harms, P Stiller, H Guth, G Bretthauer, and R Guthoff. Energy efficient algorithm for the calculation of the demand of accommodation in the artificial accommodation system. *Biomedizinische Technik/Biomedical Engineering*, 55(Suppl 1):126–128, 2010.

Hannes Harms. Konzeption einer Softwarearchitektur für das Künstliche Akkommodationssystem. Diplomarbeit, Institut für Angewandte Informatik (IAI), Karlsruher Institut für Technologie (KIT), 2010.

Hannes Harms. Objektsegmentierung auf akkumulierten Lidardaten ohne objektspezifisches Vorwissen. Studienarbeit, Institut für Mess- und Regelungstechnik (MRT), Karlsruher Institut für Technologie (KIT), 2008.

Betreute Abschlussarbeiten

Sirine Friaa. Optimized command shaping for motion control of highly flexible systems, 2015.

Ines Ben Rejeb. Development of an algorithm for staircase detection using stereo vision, 2014.

Nils Hirsenkorn. Entwicklung einer Bahnplanung mit aus der Fahrdynamik erkanntem Anhänger basierend auf einem aus Flottendaten gelernten Topologiemodell, 2014.

Anca Ene. Real time segmentation for environment sensor systems, 2014.

Johannes Gräter. Rekonstruktion von rotationssymmetrischen Objekten aus 3D Punktwolken, 2013.

**Schriftenreihe
Institut für Mess- und Regelungstechnik
Karlsruher Institut für Technologie
(1613-4214)**

- Band 001** Hans, Annegret
Entwicklung eines Inline-Viskosimeters
auf Basis eines magnetisch-induktiven
Durchflussmessers. 2004
ISBN 3-937300-02-3
- Band 002** Heizmann, Michael
Auswertung von forensischen Riefenspuren
mittels automatischer Sichtprüfung. 2004
ISBN 3-937300-05-8
- Band 003** Herbst, Jürgen
Zerstörungsfreie Prüfung von Abwasserkanälen
mit Klopferschall. 2004
ISBN 3-937300-23-6
- Band 004** Kammel, Sören
Deflektometrische Untersuchung spiegelnd
reflektierender Freiformflächen. 2005
ISBN 3-937300-28-7
- Band 005** Geistler, Alexander
Bordautonome Ortung von Schienenfahrzeugen
mit Wirbelstrom-Sensoren. 2007
ISBN 978-3-86644-123-1
- Band 006** Horn, Jan
Zweidimensionale Geschwindigkeitsmessung
texturierter Oberflächen mit flächenhaften
bildgebenden Sensoren. 2007
ISBN 978-3-86644-076-0

- Band 007** Hoffmann, Christian
Fahrzeuginnenraumdetektion durch Fusion monoskopischer Videomerkmale. 2007
ISBN 978-3-86644-139-2
- Band 008** Dang, Thao
Kontinuierliche Selbstkalibrierung von Stereokameras. 2007
ISBN 978-3-86644-164-4
- Band 009** Kapp, Andreas
Ein Beitrag zur Verbesserung und Erweiterung der Lidar-Signalverarbeitung für Fahrzeuge. 2007
ISBN 978-3-86644-174-3
- Band 010** Horbach, Jan
Verfahren zur optischen 3D-Vermessung spiegelnder Oberflächen. 2008
ISBN 978-3-86644-202-3
- Band 011** Böhringer, Frank
Gleiselektive Ortung von Schienenfahrzeugen mit bordautonomer Sensorik. 2008
ISBN 978-3-86644-196-5
- Band 012** Xin, Binjian
Auswertung und Charakterisierung dreidimensionaler Messdaten technischer Oberflächen mit Riefentexturen. 2009
ISBN 978-3-86644-326-6
- Band 013** Cech, Markus
Fahrspurschätzung aus monokularen Bildfolgen für innerstädtische Fahrerassistentenanwendungen. 2009
ISBN 978-3-86644-351-8
- Band 014** Speck, Christoph
Automatisierte Auswertung forensischer Spuren auf Patronenhülsen. 2009
ISBN 978-3-86644-365-5

- Band 015** Bachmann, Alexander
Dichte Objektsegmentierung in Stereobildfolgen. 2010
ISBN 978-3-86644-541-3
- Band 016** Duchow, Christian
Videobasierte Wahrnehmung markierter Kreuzungen mit lokalem Markierungstest und Bayes'scher Modellierung. 2011
ISBN 978-3-86644-630-4
- Band 017** Pink, Oliver
Bildbasierte Selbstlokalisierung von Straßenfahrzeugen. 2011
ISBN 978-3-86644-708-0
- Band 018** Hensel, Stefan
Wirbelstromsensorbasierte Lokalisierung von Schienenfahrzeugen in topologischen Karten. 2011
ISBN 978-3-86644-749-3
- Band 019** Carsten Hasberg
Simultane Lokalisierung und Kartierung spurgeführter Systeme. 2012
ISBN 978-3-86644-831-5
- Band 020** Pitzer, Benjamin
Automatic Reconstruction of Textured 3D Models. 2012
ISBN 978-3-86644-805-6
- Band 021** Roser, Martin
Modellbasierte und positionsgenaue Erkennung von Regentropfen in Bildfolgen zur Verbesserung von videobasierten Fahrerassistenzfunktionen. 2012
ISBN 978-3-86644-926-8

- Band 022** Loose, Heidi
Dreidimensionale Straßenmodelle für Fahrerassistenzsysteme auf Landstraßen. 2013
ISBN 978-3-86644-942-8
- Band 023** Rapp, Holger
Reconstruction of Specular Reflective Surfaces using Auto-Calibrating Deflectometry. 2013
ISBN 978-3-86644-966-4
- Band 024** Moosmann, Frank
Interlacing Self-Localization, Moving Object Tracking and Mapping for 3D Range Sensors. 2013
ISBN 978-3-86644-977-0
- Band 025** Geiger, Andreas
Probabilistic Models for 3D Urban Scene Understanding from Movable Platforms. 2013
ISBN 978-3-7315-0081-0
- Band 026** Hörter, Marko
Entwicklung und vergleichende Bewertung einer bildbasierten Markierungslichtsteuerung für Kraftfahrzeuge. 2013
ISBN 978-3-7315-0091-9
- Band 027** Kitt, Bernd
Effiziente Schätzung dichter Bewegungsvektorfelder unter Berücksichtigung der Epipolargeometrie zwischen unterschiedlichen Ansichten einer Szene. 2013
ISBN 978-3-7315-0105-3
- Band 028** Lategahn, Henning
Mapping and Localization in Urban Environments Using Cameras. 2013
ISBN 978-3-7315-0135-0

- Band 029** Tischler, Karin
Informationsfusion für die kooperative
Umfeldwahrnehmung vernetzter Fahrzeuge. 2014
ISBN 978-3-7315-0166-4
- Band 030** Schmidt, Christian
Fahrstrategien zur Unfallvermeidung im
Straßenverkehr für Einzel- und
Mehrobjektszenarien. 2014
ISBN 978-3-7315-0198-5
- Band 031** Firl, Jonas
Probabilistic Maneuver Recognition
in Traffic Scenarios. 2014
ISBN 978-3-7315-0287-6
- Band 032** Schönbein, Miriam
Omnidirectional Stereo Vision
for Autonomous Vehicles. 2015
ISBN 978-3-7315-0357-6
- Band 033** Strauß, Tobias
Kalibrierung von Multi-Kamera-Systemen.
Kombinierte Schätzung von intrinsischem
Abbildungsverhalten der einzelnen Kameras
und deren relativer Lage zueinander ohne
Erfordernis sich überlappender Sichtbereiche. 2016
Nicht erschienen
- Band 034** Liebner, Martin
Fahrerabsichtserkennung und Risikobewertung für
warnende Fahrerassistenzsysteme. 2016
ISBN 978-3-7315-0508-2
- Band 035** Ziegler, Julius
Optimale Trajektorienplanung für Automobile. 2017
ISBN 978-3-7315-0553-2

Band 036 Harms, Hannes
**Genauigkeitsuntersuchung von
binokularen Normalenvektoren für
die Umfeldwahrnehmung.** 2017
ISBN 978-3-7315-0628-7

In modernen Sicherheitsassistenzsystemen spielt die präzise Umfelderkennung eine zentrale Rolle für die Funktionssicherheit und -zuverlässigkeit. Zunehmend werden binokulare Kamerasysteme in diesem Bereich eingesetzt. Insbesondere die Rekonstruktion von dichten Punktwolken aus binokularen Kamerasystemen bietet die Möglichkeit, Objekte im Umfeld dreidimensional zu erfassen. Die Orientierung der Objekte liefert für viele Methoden der Umfeldwahrnehmung eine wertvolle Zusatzinformation. Diese Orientierungsschätzung unterliegt jedoch einer Unsicherheit, die abhängig von der betrachteten Szene und dem verwendeten Sensor variiert.

In der vorliegenden Arbeit wird eine Genauigkeitsuntersuchung von Normalenvektoren aus binokularen Kamerasystemen vorgestellt, die eine Modellierung der Orientierungsunsicherheit der Normalenvektoren aus wenigen Parametern ermöglicht. Die Ergebnisse der Genauigkeitsuntersuchung werden verwendet, um zwei Methoden zur Umfelderkennung aus Orientierungsinformationen auszulegen: eine Freibereichsschätzung für Fahrzeuge sowie eine Treppendetektion für Menschen mit eingeschränkter Sehfähigkeit.

ISSN 1613-4214

ISBN 978-3-7315-0628-7

Gedruckt auf FSC-zertifiziertem Papier

ISBN 978-3-7315-0628-7



9 783731 506287 >