Analyse der Position, Orientierung und Bewegung von rigiden und artikulierten Objekten aus Stereobildsequenzen

Björn Barrois

Mai 2010

Dissertation zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften (Dr.-Ing.)

Abdruck der genehmigten Dissertation zur Erlangung des akademischen Grades *Doktor der Ingenieurwissenschaften (Dr.-Ing.)* an der Technischen Fakultät der Universität Bielefeld.

Dipl.-Ing. (FH) Björn Barrois
email: b.barrois@gmx.de

Gutachter:
Prof. Dr. Franz Kummert, Universität Bielefeld
Prof. Dr. Christian Wöhler, Universität Dortmund

Prüfungsausschuß:
Prof. Dr. Barbara Hammer, Universität Bielefeld
Prof. Dr. Franz Kummert, Universität Bielefeld
Prof. Dr. Christian Wöhler, Universität Dortmund
Dr. Robert Haschke, Universität Bielefeld

Für meine Tochter Johanna Marie

Danksagung

Meinem Doktorvater Prof. Franz Kummert danke ich für seine Unterstützung im Rahmen der Dissertation während der letzten Jahre.

Prof. Christian Wöhler gilt mein besonderer Dank. Er hat mir während der Arbeit stets mit Rat und Tat zur Seite gestanden und mich vor und während der Promotion ermutigt und motiviert.

Ich möchte den Kollegen vom Forschungszentrum der Daimler AG in Ulm, wo diese Arbeit entstanden ist, danken. Für seine Unterstützung danke ich Dr. Ulrich Kreßel. Als Teamleiter war er für die hervorragende Arbeitsumgebung mit verantwortlich. Den Teamkollegen Dr. Lars Krüger, Christoph Hermes und Markus Hahn danke ich für die hervorragende Atmosphäre und die vielen fachlichen Diskussionen. Bei Markus Hahn möchte ich mich auch für die gemeinsame Arbeit an dem in dieser Dissertation erwähnten Fusionsansatz ausdrücklich bedanken.

Ebenso bin ich für die Ratschläge von Prof. Rainer Ott und seine aufwändigen Korrekturen sehr dankbar.

Des Weiteren möchte ich Marcus Konrad, Stella Hristova und Marc Quirin für die sehr gute Zusammenarbeit danken.

Abschließend möchte ich auch Prof. Dietmar Brück danken, da er mich bei meiner Entscheidung zum Beginn der Promotion ermutigt hat und somit den Grundstein dieser Arbeit mit gelegt hat.

Zuletzt gilt meine besondere Wertschätzung meiner liebenswerten Frau Martina. Sie hat über ungewöhnliche Arbeitszeiten stets hinweg gesehen und mich in der Endphase durch zahlreiche Korrekturen unterstützt.

Kurzfassung

Diese Arbeit beschäftigt sich mit der Analyse der Position, Orientierung und Bewegung von rigiden und artikulierten Objekten aus Stereobildsequenzen. Grundlage für diese Analyse ist eine raum-zeitliche Szenenrekonstruktion auf Basis der bi- oder trinokularen Bildsequenzen unter Verwendung von Stereobildverarbeitung und Berechnung des optischen Flusses. Darauffolgend wird eine Vorsegmentierung der Szene durchgeführt, um die einzelnen Objekte darin zu separieren.

Verschiedene neuentwickelte Verfahren zur Pose-Estimation kommen zum Einsatz, darunter problemspezifische Varianten des Iterative Closest Point-Algorithmus, mit deren Hilfe sich aus 3D-Punktewolken unter Zuhilfenahme von Modellwissen die Pose eines Objekts (Position und Orientierung des Objekts) ermitteln lässt. Alternativ dazu werden in dieser Arbeit auch neuartige Ansätze zur modellbasierten Stereobildverarbeitung und zur Fusion von kontur- und punktbasierter Poseschätzung untersucht.

Anschließend werden Methoden zur Bewegungsanalyse benutzt, um die zeitliche Ableitung der Pose, die Bewegung, bestmöglich zu ermitteln. Bei diesen neu entwickelten Ansätzen wird auf Basis des optischen Flusses, konturbasiert oder unter Verwendung eines neuartigen modellbasierten Szenenfluss-Verfahrens, die vollständige Objektbewegung ermittelt.

Es werden verschiedene Systemausprägungen vorgestellt, die anwendungsspezifisch in der Lage sind, robust die Pose und die Bewegung eines rigiden oder artikulierten Objekts mit einer hohen Genauigkeit zu schätzen. Um zu zeigen, dass der entwickelte Ansatz für sehr unterschiedliche Szenarien einsetzbar ist, wird zum einen das Produktionsszenario in der Fabrik und zum anderen das Straßenverkehrsszenario im Kreuzungsbereich untersucht. Im Produktionsszenario geht es um die Interaktion von Mensch und Roboter. Dabei wird die Hand-Unterarm-Region des Arbeiters als artikuliertes Objekt untersucht. Im Straßenverkehrsszenario geht es hingegen um die Realisierung von Fahrerassistenzsystemen. Hier werden ganze Fahrzeuge als rigide Objekte betrachtet.

Eine Evaluierung auf Basis realistischer Sequenzen belegt unter Verwendung einer genauen, mit einem anderen Verfahren ermittelten Ground-Truth diese hohen Genauigkeiten, sowohl bei der neuartigen Schätzung von Position und Orientierung, als auch bei der neuentwickelten Bewegungsanalyse. Es wird außerdem gezeigt, dass eine Fusion bei kleinen Objekten im Produktionsszenario sinnvoll ist, da eine Kombination von punkt- und konturbasierter Poseschätzung eine wesentliche Erhöhung der Systemrobustheit mit sich bringt. Im Straßenverkehrsszenario zeigt die ausschließliche Verwendung von 3D-Punkten sehr gute Ergebnisse, wobei die Verwendung eines modellbasierten Stereoansatzes eine weitere Steigerung bewirkt. Zur Schätzung der vollständigen Bewegung der Fahrzeuge zeigt ein modellbasierter Szenenflussansatz im Straßenverkehr gute Ergebnisse, wohingegen in der Produktion, bei bekannter Objektkontur ein konturbasierter

Ansatz oder eine erweiterte Methode zur Analyse der Verschiebungsvektorfelder zum Einsatz kommt.

Ein Hauptaugenmerk bei dem neuentwickelten System liegt in der Einbringung von Rückkopplungen in die Verarbeitungskette. Dadurch wird die Robustheit des Gesamtsystems nochmals erhöht, da hierdurch beispielsweise Fehlkorrespondenzen im Stereoalgorithmus verhindert werden können, oder auch artikulierte Objekte klassifiziert und somit die Objektverfolgung verifiziert werden kann. Eine weitere Eigenschaft des Systems ist eine instantane Bestimmung der Bewegung durch die Verwendung von lediglich zwei bzw. drei Zeitschritten, ohne eine zeitliche Filterung zu benutzten, die zu einer verzögerten Reaktion führen würde.

Abstract

This thesis regards the analysis of position, orientation und motion of rigid and articulated objects from stereo image sequences. In a first step, a spatio-temporal scene reconstruction based on stereo image analysis and optical flow computation on the binocular or trinocular image sequences is performed. Subsequently a segmentation stage is used to seperate the object from each other and from the backgorund.

Different newly developed pose-estimation techniques are utilized, e.g. problem-specific versions of the iterative closest point algorithm which is used to determine the object pose (position and orientation) based on the 3D point cloud using model information. Alternatively, a novel model-based stereo analysis or a novel fusion of contour-based and point-based pose estimation is applied.

Several new methods for motion analysis can be utilized to determine the temporal derivative of the pose, i.e. the object motion, instantaneously. Here the complete three-dimensional motion is determined based on optical flow data, contour-based or by using a model-based scene-flow technique.

Several system setups are presented which robustly estimate the pose and the motion of rigid or articulated objets with high accuracy. To demonstrate the usability of the approach for different szenarios the system is evaluated in the industrial production scenario and in the traffic scenario at road intersections. In the production scenario the human hand-forearm-limb is regarded as an articulated object, whereas in the traffic scenario vehicles are regarded as rigid objects.

The evaluation is based on real-world sequences and shows high accuracies for the newly developed determination of the position and orientation as well as for the novel motion analysis using independently determined ground-truth data. Furthermore, it is shown that the developed fusion approach is useful for small objects in the production szenario, because a combination of contour-based and point-based algorithms increases the robustness of the system significantly. For the traffic szenario a use of 3D points alone shows high accuracies, while the model-based stereo technique achieves a further increase of the accuracies. For the complete motion analysis in the traffic scenraio, the model-based scene flow method reaches high accuracy, whereas in the production scenario a contour-based approach or an extented method for the analysis of optical flow fields is favourably applied.

One main focus of the system is the integration of decision feedback in the processing hierarchy, leading to an increased robustness of the system. By using decision feedback, stereo matching errors can be avoided or a classification of articulated objects can be done, which allows a verification of the object detection. Another advantage of the system is the instantaneouse determination of the object motion, where just two or three time

steps are used. A temporal filtering of the object pose is thus avoided, which would result in a delayed reaction of the system.

Inhaltsverzeichnis

Teil I.

Einleitung

1. Motivation

1.1. Kontext

Die Analyse der Position, Orientierung und Bewegung von rigiden oder artikulierten Objekten hat ein breites Anwendungsspektrum. Dazu zählen auch die beiden Anwendungsszenarien, welche dieser Arbeit zu Grunde liegen: das Produktionsszenario in der Fabrik und das Straßenverkehrsszenario im Kreuzungsbereich. Dabei ist das Ziel die Pose (Position und Orientierung) sowie deren zeitliche Ableitung, die Bewegung möglichst genau und mit einer hohen Robustheit zu ermitteln. Außerdem ist eine hohe Analysegeschwindigkeit wegen der erfolgerlichen kurzen Reaktionszeiten notwendig, weshalb von einer zeitlichen Filterung zur Ermittlung der Bewegung wegen großer Einschwingzeiten abgesehen werden soll, da dadurch stets eine gewisse Verzögerung entsteht.

1.1.1. Produktion

Im Bereich von Industrieanlagen hat die Sicherheit der dort arbeitenden Personen oberste Priorität. Gemäß der entsprechenden Gesetze (siehe Kap. 2.3) muss der Arbeitgeber verschiedenste Vorkehrungen treffen und so die Gefahren minimieren. Bis zum Jahre 2007 wurden dafür entweder eine komplette Trennung der Arbeitsbereiche von Mensch und Maschine vorgesehen, oder falls ein gemeinsamer Arbeitsbereich notwendig war, mittels Lichtgitter, Laserscanner, Trittmatten etc. ein abwechselndes Arbeiten von Mensch und Maschine ermöglicht. Seit dem Jahr 2007 ist nun ein visueller Sensor zur 3D-Objekterkennung der Firma Pilz verfügbar, der die verschiedenen, meist zahlreichen Sensoren durch ein Kamerasystem ersetzt. Dadurch ergeben sich verschiedene Vorteile: Zum einen können nun Warn- und Schutzbereiche per Software definiert werden und somit flexibler gestaltet werden. Zum anderen wird der teilweise enorme Hardwareaufwand früherer Sicherheitssysteme vermieden, was auch auf der finanziellen Seite zum Tragen kommt.

Durch eine aufwendige Prüfung durch die Berufgenossenschaft wurde das System als sicherheitstechnisch zuverlässig deklariert und wird seit diesem Zeitpunkt in verschiedenen Industrien, darunter auch in der Automobilindustrie, eingesetzt.

Dieses sogenannte SafetyEye-System arbeitet auf Basis einer trinokularen Optik und zwei unabhängig voneinander arbeitenden Stereo-Triangulationsalgorithme. Auf beide Algorithmen wird in Kap. 2.1.2 eingegangen. Durch die definierten Warn- bzw. Schutzbereiche erfolgt eine Überprüfung, ob sich berechnete 3D-Punkte in diesen Bereichen befinden. Dringt ein Objekt in diese Bereiche ein, wird die Anlage (z.B. ein Industrieroboter) abgeschaltet oder wenigstens dessen Fahrt verlangsamt.

Nachteilig bei dem System ist die fehlende Klassifikation der im Sicherheitsbereich befindlichen Objekte. Sind diese Objekte lediglich ein Schweißfunken, ein Insekt oder ähnliches, so ist die Abschaltung nicht sinnvoll, verursacht jedoch vermeidbare Kosten.

Außerdem bleibt die Bewegung der Objekte unbeachtet. Dadurch müssen die Sicherheitsbereiche größer definiert werden, um eine Kollision zwischen Mensch und Maschine unabhängig von der jeweiligen Geschwindigkeit zu verhindern. Durch eine gezielte Bewegungserkennung der arbeitenden Person im Umfeld des Industrieroboters, könnte die Person noch besser geschützt werden und gleichzeitig würden unnötige Störungen bei der Arbeit des Roboters verhindert. Durch eine Prädiktion der Position der Person kann ein Aussage darüber getroffen werden, wo sich die Person bzw. deren Körperteile in Zukunft befinden. Durch die bekannte Bewegung des Roboters kann so ein Zusammenstoß vorhergesagt werden und somit frühzeitlich gewarnt werden.

Eine weitere Anwendung, bei der die Bewegungserkennung von Körperteilen des Arbeiters sinnvoll ist, ist die Mensch-Roboter-Interaktion. Roboter sind dazu geeignet, einfache, zyklische Arbeiten zu verrichten. Jedoch gibt es immer noch komplexe und individuelle Arbeiten, bei denen der Mensch wesentlich besser geeignet ist. Daher ist eine Zusammenarbeit bzw. eine Interaktion sinnvoll und bedarf ebenso einer Analyse der Position, Orientierung und Bewegung des Arbeiters. Die Pose und die Bewegung des Roboters ist durch die Parameter aus dessen Programmierung bei den derzeit modernsten Industrierobotern auch in sicherheitstechnisch spezifizierter Form verfügbar.

1.1.2. Straßenverkehr

Im Straßenverkehr ist die Entwicklung von Fahrerassistenzsystemen in den letzten Jahren rasch vorangeschritten. Fahrerassistenzsysteme bieten eine Steigerung der Sicherheit, der Energieeffizienz oder auch des Komforts, durch die Analyse der Umwelt, der aktuellen Situation, von Verkehrsinfrastruktur oder auch der anderen Verkehrsteilnehmer. Winner et al. (2009) beschreiben detailliert aktuelle Entwicklungen in diesem Feld.

Speziell für den Kreuzungsbereich sind momentan noch keine kommerziellen Fahrerassistenzsysteme verfügbar, was maßgeblich an der Komplexität der Situation liegt. Es ist zum einen anspruchsvoll alle relevanten Verkehrsteilnehmer zu detektieren und deren Verhalten zu analysieren. Dazu gehört es auch deren Absicht zu erkennen, um daraus ein Gefahrenpotential für das eigene Verhalten ableiten zu können. Zum anderen muss die vorliegende Situation stets schnell erkannt werden, da sich die Zustände der Verkehrsteilnehmer im Kreuzungsbereich schnell ändern können. Daher ist es wichtig, zum einen die Pose der anderen Verkehrsteilnehmer möglichst genau zu kennen und zum anderen deren Bewegung ebenso zu bestimmen.

Diese Daten können direkt benutzt werden, um mithilfe der eigenen Bewegungsdaten eine Kollisionsgefahr frühzeitig zu erkennen. Dabei ist vorallem die Gefahr bei querendem Verkehr sehr hoch und kann durch aktuelle Sensoren im Fahrzeug (Radar oder monokulare Kamerasysteme) nicht ausreichend genau eingeschätzt werden. Außerdem ist auf Basis dieser Daten eine Klassifikation der Trajektorien der anderen Verkehrsteilnehmer möglich, wodurch eine Langzeitprädiktion mit einem Zeithorizont von bis zu zwei Sekunden ermöglicht wird. Arbeiten dazu wurden beispielsweise von Hermes et al.

(2009) veröffentlicht.

Ist der aktuelle Zustand der anderen Verkehrsteilnehmer bekannt, kann zunächst durch einen autonomen Brems- und/oder Lenkeingriff bei drohender Gefahr die Situation entschärft werden. Dadurch können Unfälle mit Sach- oder auch Personenschäden verhindert werden. Des Weiteren kann der Fahrer aktiv im Kreuzungsbereich unterstützt werden, um eine gefahrlose Durchfahrt durch die Kreuzung zu gewährleisten. Dadurch wird der Fahrer gerade in komplexen, innerstädtischen Situationen entlastet, wodurch verkehrsbedingter Streß abgebaut wird.

1.2. Sensorik

In der vorliegenden Arbeit kommt ausschließlich die Stereokamera als Sensor zur Erfassung von dreidimensionalen Szeneninformationen zum Einsatz. Im Folgenden soll erläutert werden, weshalb dieser Sensor für die beiden betrachteten Szenarien vorteilhaft ist und was Vor- und Nachteile anderer Sensorkonzepte sind. Dazu zeigt Tabelle 1.1 zunächst eine Übersicht über Sensoren zur 3D-Rekonstruktion.

Sensor	Vorteile	Nachteile
Radar	Entfernungs- und Geschwindigkeitsinformation, sichtunabhängig	geringe laterale Auflösung, Phantombilder, nur zwei Raumdimensionen
Laserscanner	hohe laterale Auflösung; hohe Tiefenauflösung	sichtabhängig, nur zwei Raumdimensionen, hohe Kosten
Sonar	geringe Kosten	breite Signalkeule, Störung untereinander, geringe Reichweite
Taktiler Sensor	hohe Genauigkeit	physikalischer Kontakt
PMD-Sensor	direktes 3D-Abbild der Szene, hohe Tiefenauflösung	hohe Kosten; geringe laterale Auflösung
Stereokamera	hoher Informationsgehalt, hohe laterale Auflösung, geringe Kosten	sichtabhängig, hoher Berechnungsaufwand

Tabelle 1.1.: Vergleich von Sensoren zur 3D-Szenenrekonstruktion

Aus Tabelle 1.1 ist ersichtlich, dass unter der Vorraussetzung, dass eine hohe laterale Auflösung in beiden Szenarien notwendig ist, jedoch auch der Kostenaspekt nicht außer Acht gelassen werden darf, nur die Stereokamera als Sensor in Frage kommt. Auf den genauen Aufbau des Kamerasystems wird jeweils in den Experimenten in Teil III eingegangen, wobei hier auch die Verwendung von Grauwert- oder Farbkameras untersucht wird.

Im Produktionsumfeld werden seit längerem optische Sensoren zum Schutz von Mensch und Maschine eingesetzt. Dabei sind häufig zweidimensionale Sensoren im Einsatz, wodurch zur dreidimensionalen Überwachung oft eine Vielzahl von Sensoren not-

wendig sind. Dies ist zum einen aufgrund der hohen Kosten nachteilig, aber auch die Flexibilität leidet darunter, da beispielsweise bei kleinen Änderungen der Anlage evtl. ein Umbau des gesamten Sicherheitskonzepts notwendig wird.

Der Stand der Forschung, der in Kap. 2.2 und 2.3 wiedergegeben wird, zeigt, dass bereits eine Vielzahl von Arbeiten auf dem Gebiet der Poseschätzung von Personen unter Verwendung von multiokularen Kamerasystemen existieren und vielversprechende Ergebnisse aufweisen.

In der Automobilindustrie kommen bereits heute verschiedene Sensorkonzepte zum Einsatz (Winner et al., 2009). Dazu zählt der Radarsensor, welcher meist für die weit nach vorne gerichtete Erkennung von Fahrzeugen im Längsverkehr benutzt wird. Ultraschallsensoren kommen häufig als Parksensorik zum Einsatz. Und Kameras werden als Rückfahrhilfe, zur Spur- und Verkehrszeichenerkennung oder als Infrarotausführung zur Unterstützung bei der Fahrt bei Nacht verwendet. Für die Analyse von querendem Verkehr sind auf dem Markt derzeit keine Systeme vorhanden, jedoch zeigt der Stand der Forschung, wie er in Kap. 2.4 wiedergegeben wird, dass hier sowohl Laserscanner, Stereokameras als auch eine Kombination von Kamera und Radar bzw. Kamera und Laserscanner zum Einsatz kommen. Da die kostengünstige Bereitstellung eines zuverlässigen Laserscanners in naher Zukunft allgemein als unwahrscheinlich gilt, wird von dieser Sensorik abgesehen. Eine Fusion von monokularer Kamera mit Radar ist ein vielversprechender Ansatz, der bereits in verschiedenen Forschungsarbeiten betrachtet wurde. Eine Stereokamera bietet demgegenüber jedoch den Vorteil, dass eine aufwendige Fusion entfällt und durch eine direkte, dreidimensionale Analyse mehr Daten zur Verfügung stehen, als durch die Kopplung der jeweils zweidimensionalen Daten von Kamera und Radar.

2. Stand der Forschung

2.1. Grundlagen und Methoden der Bildverarbeitung

2.1.1. Grundlagen Kamerasystem

Lochkameramodell

Aufgrund der metrischen Szenenrekonstruktion, die Grundlage der vorliegenden Arbeiten ist, ist ein genaues Modell des verwendeten Kamerasystems notwendig. Die Basis dafür ist das Lochkameramodell. In der vorliegenden Arbeit wird zum besseren Verständnis der geometrischen Zusammenhänge die Notatin nach Craig (1989) verwendet.

Die Lochkamera ist ein einfacher Kameraaufbau, welcher bereits im 6. Jahrhundert vor Christus im chinesischen Raum erwähnt wurde. Das Licht, welches von der Oberfläche der beobachteten Objekte reflektiert wird, tritt durch die sehr kleine Lochblende der Kamera. Hinter dieser Lochblende befindet sich in einem definierten Abstand ein lichtsensitives Material bzw. ein Sensorchip. Die Kamera besitzt keinerlei Linsen. Das daraus abgeleitete Lochkameramodell beschreibt grundlegend den Prozess der Bildgewinnung, genauer gesagt die Beziehung zwischen einem 3D-Punkt vor der Kamera und dessen Projektion auf die Bildebene (siehe (Faugeras, 1993), (Horn, 1986) oder (Forsyth und Ponce, 2002)).

Das Lochkameramodell wird durch sein optisches Zentrum $\mathcal{O}$ und seine Bildebene $\mathcal{R}$ beschrieben, wobei die Distanz dazwischen als Kamerakonstante f bezeichnet wird (siehe Abb. 2.1). Die Objektivebene $\mathcal{F}$ verläuft durch das optische Zentrum der Kamera und ist parallel zur Bildebene. Die optische Achse ist die Linie senkrecht dazu und verläuft durch das optische Zentrum, wobei der Schnittpunkt zwischen der optischen Achse und der Bildebene als Kamerahauptpunkt bezeichnet wird.

Der 3D-Punkt ${}^{C}\mathbf{w}_i = [x_i\ y_i\ z_i]^T$ im Kamerakoordinatensystem C wird auf den Bildpunkt ${}^{J}\mathbf{m}_i = [u_i\ v_i]^T$ im Bildkoordinatensystem J projiziert, welcher dem Schnittpunkt zwischen $\mathcal{R}$ und dem Strahl (*Sehstrahl*) von ${}^{C}\mathbf{w}_i$ durch $\mathcal{O}$ entspricht. Der Ursprung des Kamerakoordinatensystems C ist das Loch der Lochkamera, wohingegen der Ursprung des Bildkoordinatensystems J der Kamerahauptpunkt in der Bildebene ist. Gleichung 2.1 definiert diese Beziehung mathematisch:

$$u_i = -f\frac{x_i}{z_i}\ ; \quad v_i = -f\frac{y_i}{z_i}. \tag{2.1}$$

Verschiebt man die Bildebene gedanklich aus ihrer ursprünglichen Position (Abstand

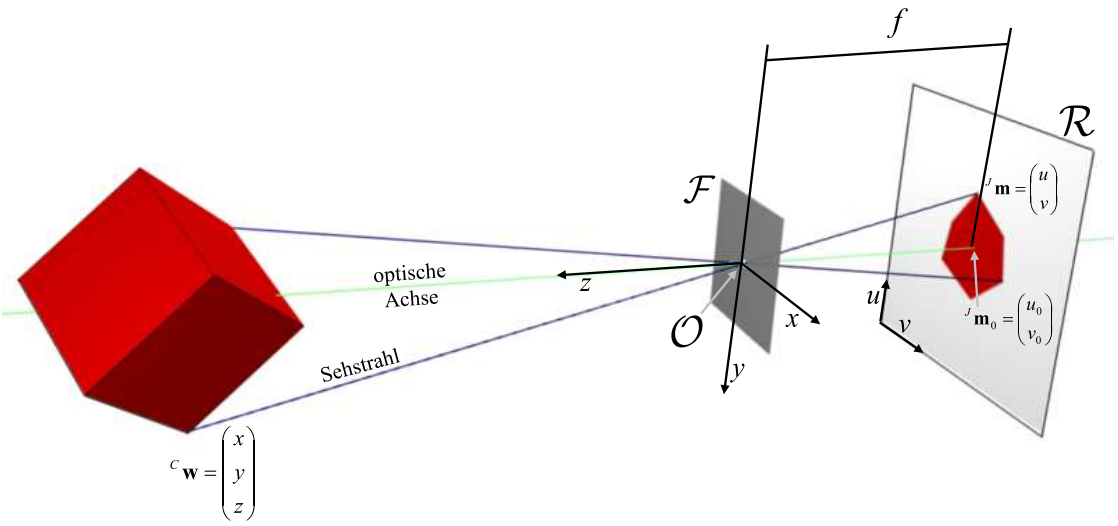


Abbildung 2.1.: Projektion mittels Lochkameramodell.

f hinter der Lochblende) vor die Lochblende, wobei die Kamerakonstante als Abstand beibehalten wird, so verschwindet das negative Vorzeichen in Gleichung 2.1. Dadurch werden die Zusammenhänge übersichtlicher und einfacher zu handhaben. Diese gedankliche Verschiebung der Bildebene wird in der vorliegenden Arbeit benutzt.

Die Projektion des 3D-Punktes ${}^{C}\mathbf{w}_i$ auf den 2D-Bildpunkt ${}^{J}\mathbf{m}_i$ kann mathematisch durch eine lineare Transformation in homogenen Koordinaten ausgedrückt werden.

Dafür wird der 3D-Punkt ${}^{C}\mathbf{w}_i$ als homogene Koordinate mittels ${}^{C}\tilde{\mathbf{w}}_i = [x_i \; y_i \; z_i \; 1]^T$ repräsentiert und ${}^{J}\mathbf{m}_i$ durch ${}^{J}\tilde{\mathbf{m}}_i = [u_i \; v_i \; 1]^T$. Die perspektivische Transformation (Gleichung 2.2) ist durch Matrix ${}^{J}_{C}\tilde{\mathbf{P}}$ definiert, wobei $\cong$ bedeutet, dass hier ein beliebiger Skalierungsfaktor angenommen werden kann.

$$ {}^{J}\tilde{\mathbf{m}}_i \cong {}^{J}_{C}\tilde{\mathbf{P}} \; {}^{C}\tilde{\mathbf{w}}_i \;, \text{ wobei } {}^{J}_{C}\tilde{\mathbf{P}} = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{2.2} $$

Durch die Projektion eines 3D-Punktes aus dem Kamerakoordinatensystem auf einen 2D-Punkt im Bildkoordinatensystem geht eine Dimension verloren. Genauer gesagt wird maßgeblich die Tiefeninformation gelöscht. Grund dafür ist, dass alle 3D-Punkte, die auf demselben Sehstrahl liegen, auch auf den gleichen Bildpunkt in der Bildebene projiziert werden. Die fehlende Tiefeninformation kann ohne zusätzliche Information nicht wieder rekonstruiert werden.

Da der Sensorchip in einzelne Pixel aufgeteilt ist, ist eine Umwandlung der Absolutwerte von Metern in Anzahl der Pixel sinnvoll. Dafür überführt die interne Transformationsmatrix ${}^{P}_{J}\tilde{\mathbf{P}}$ in Gleichung 2.3 die 2D-Punkte aus dem Bildkoordinatensystem J in das Pixelkoordinatensystem P. Der Ursprung des Pixelkoordinatensystems ist, wird wiederum die gedanklich vor den Kamerahauptpunkt verschobene Bildebene angenommen,

typischerweise in der oberen linken Ecke des Bildes definiert.

$$ {}^{P}\tilde{\mathbf{m}}_i = {}^{P}_{J}\tilde{\mathbf{P}}\ {}^{J}\tilde{\mathbf{m}}_i \ , \text{ wobei } {}^{P}_{J}\tilde{\mathbf{P}} = \begin{pmatrix} k_u & \gamma & u_0 \\ 0 & k_v & v_0 \\ 0 & 0 & 1 \end{pmatrix} \tag{2.3} $$

Die Parameter k_u und k_v definieren die effektive Anzahl von Pixeln pro Meter entlang der u bzw. v-Achse. Der Hauptpunkt ${}^{J}\mathbf{m}_0 = [u_0\ v_0]^T$ wird in Pixeln gemessen und zur Modellierung nicht orthogonaler u-v Achsen wird der sogenannte *Skew-Parameter* γ verwendet. Da in vielen Fällen, beispielsweise bei der Stereobildverarbeitung, ein Weltkoordinatensystem sinnvoll ist, werden mittels Matrix ${}^{C}_{W}\tilde{\mathbf{P}}$ Punkte aus dem Kamerakoordinatensystem C ins Weltkoordinatensystem W transformiert.

$$ {}^{C}\tilde{\mathbf{w}}_i = {}^{C}_{W}\tilde{\mathbf{P}}\ {}^{W}\tilde{\mathbf{w}}_i \ , \text{ wobei } {}^{C}_{W}\tilde{\mathbf{P}} = \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{2.4} $$

Die externen Parameter der Kamera werden durch die Parameter $\mathbf{R}$ und $\mathbf{t}$ (siehe Gleichung 2.4) definiert, was die Position und Orientierung des Kamerakoordinatensystems C relativ zum Weltkoordinatensystem W darstellt. Sind Welt- und Kamerakoordinatensystem identisch, so ist $\mathbf{R}$ eine 3×3 Einheitsmatrix und $\mathbf{t}$ ein dreidimensionaler Nullvektor.

Um eine vollständige Transformation von Weltkoordinaten in Pixelkoordinaten zu erhalten, können alle Matrizen mittels Matrixmultiplikation zu einer Matrix ${}^{P}_{W}\tilde{\mathbf{P}}$ zusammengefasst werden:

$$ {}^{P}\tilde{\mathbf{m}}_i = {}^{P}_{J}\tilde{\mathbf{P}} \cdot {}^{J}_{C}\tilde{\mathbf{P}} \cdot {}^{C}_{W}\tilde{\mathbf{P}} \cdot {}^{W}\tilde{\mathbf{w}}_i = {}^{P}_{W}\tilde{\mathbf{P}} \cdot {}^{W}\tilde{\mathbf{w}}_i. \tag{2.5} $$

Linsenverzeichnungen: Das Lochkameramodell ist nur präzise, wenn die Lochblende unendlich klein ist und keine Linsen benutzt werden. Diese Annahmen können in der Realität nicht erfüllt werden. Jedoch kann das Modell weiterhin als Grundlage für die Modellierung genutzt werden.

Die Linse vor der Blende führt zu Verzeichnungen im resultierenden Bild. Slama (1980) beschreibt Modelle für diese Verzeichnungseffekte. Nachdem die Parameter des Verzeichnungsmodells für ein Kamerasystem bestimmt wurden, können die aufgenommenen Bilder im Sinne einer idealen Lochkamera korrigiert werden. Auf Basis der korrigierten Bilder sind metrische Messungen möglich.

Zunächst werden die *tonnenförmigen* Verzeichnungen (siehe Abb. 2.2) modelliert. Durch Gleichung 2.6 ist es möglich, die Verschiebung Δu bzw. Δv zu berechnen, die durch die Linsenverzeichnung entstehen, wobei k_1, k_3, k_5, ... die Verzeichnungskoeffizienten sind und der Radius $r = \sqrt{{}^{J}u^2 + {}^{J}v^2}$ von der Bildmitte aus berechnet wird.

$$\begin{bmatrix} \Delta u_r \\ \Delta v_r \end{bmatrix} = \begin{bmatrix} {}^J u(k_1 r^2 + k_3 r^4 + k_5 r^6 + ...) \\ {}^J v(k_1 r^2 + k_3 r^4 + k_5 r^6 + ...) \end{bmatrix} \tag{2.6}$$

Um die Verzeichnungen ausreichend zu kompensieren werden meist drei Verzeichnungskoeffizienten verwendet. Die Verzeichnungen werden bei diesem Modell im Bildkoordinatensystem J beschrieben.

Neben den radialen Verzeichnungen werden auch die tangentialen Verzeichnungen mitmodelliert. Die tangentialen Verzeichnungen beschreiben die Verschiebung des Krümmungszentrums der Linsenoberflächen relativ zur optischen Achse. Gleichung 2.7 beschreibt diese Verschiebung mathematisch.

$$\begin{bmatrix} \Delta u_t \\ \Delta v_t \end{bmatrix} = \begin{bmatrix} 2k_2^J u^J v + k_4(r^2 + 2^J u^2) \\ k_2(r^2 + 2^J v^2) + 2k_4^J u^J v \end{bmatrix} \tag{2.7}$$

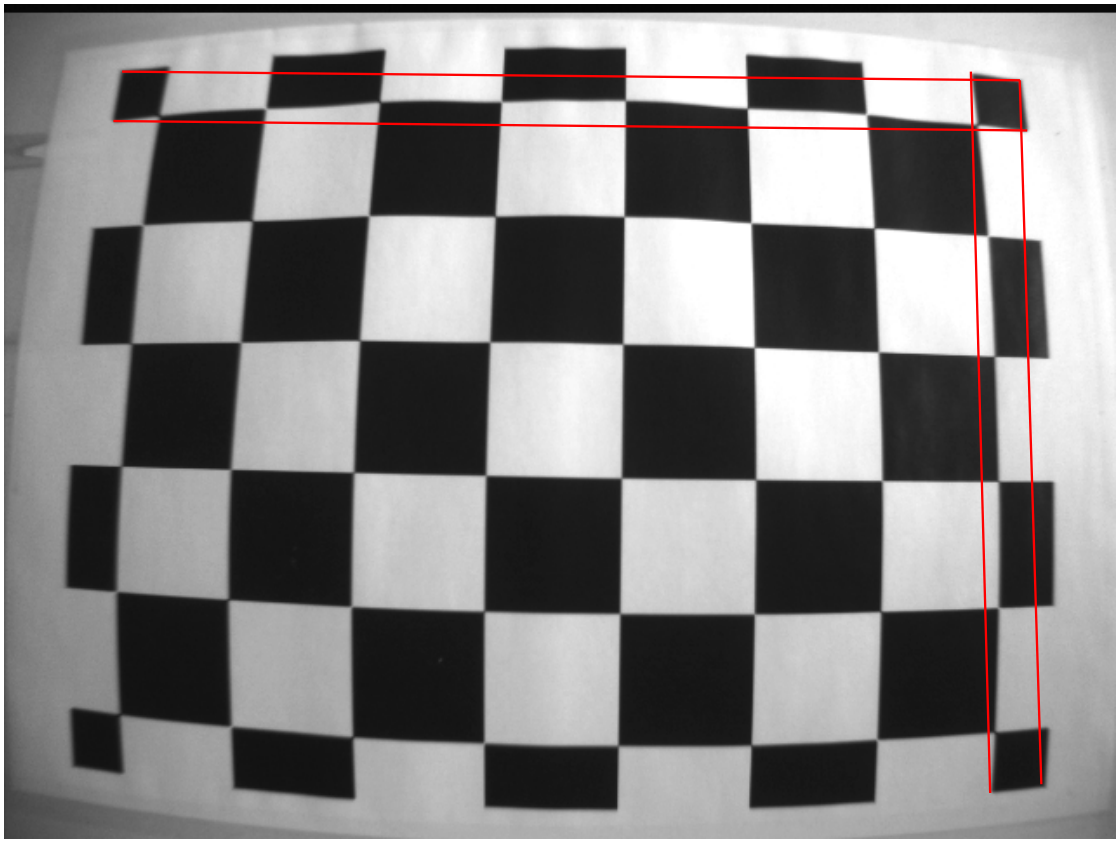


Abbildung 2.2.: Linsenverzeichnungen am Beispiel eines Schachbrettmusters. Durch die tonnenförmigen Verzeichnungen werden aus Geraden Kurven.

Zusammengefasst kann durch Gleichung 2.8 die Beziehung zwischen verzeichneten

Koordinaten $({}^{J}u,{}^{J}v)$ und unverzeichneten Koordinaten $({}^{J}u',{}^{J}v')$ beschrieben werden.

$$\begin{bmatrix} {}^{J}u' \\ {}^{J}v' \end{bmatrix} = \begin{bmatrix} {}^{J}u + \Delta u_r + \Delta u_t \\ {}^{J}v + \Delta v_r + \Delta v_t \end{bmatrix} \tag{2.8}$$

Somit werden die Verzeichnungseigenschaften des Kamerasystems über die Parameter $k_1 - k_5$, welche bei der Kamerakalibrierung gleich mitgeschätzt werden, definiert. Diese Parameter gehören zu den intrinsischen Kameraparametern.

Point Spread Function: Ein anderer Effekt, der durch die Verwendung einer endlich kleinen Blende hervorgerufen wird, ist die Defokussierung. Die Lochkamera projiziert jeden Punkt aus dem Objektraum scharf auf die Bildebene, wohingegen eine Kamera mit realer Blende den Punkt tiefenabhängig verbreitert.

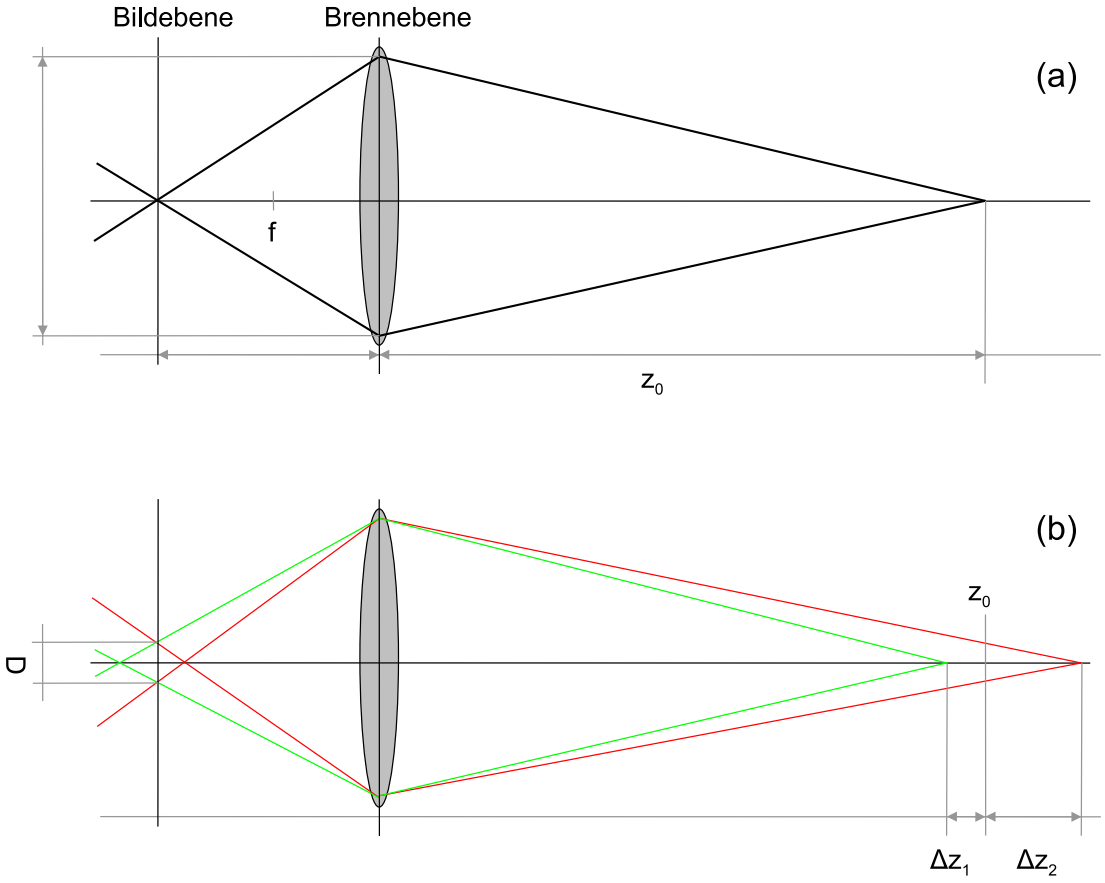


Abbildung 2.3.: Entstehung der Unschärfe bei endlicher Apertur. (a) Optimaler Abstand z_0 zur Blende. (b) Größerer bzw. kleinerer Abstand führt zu verbreitertem Punkt auf der Bildebene mit Durchmesser D.

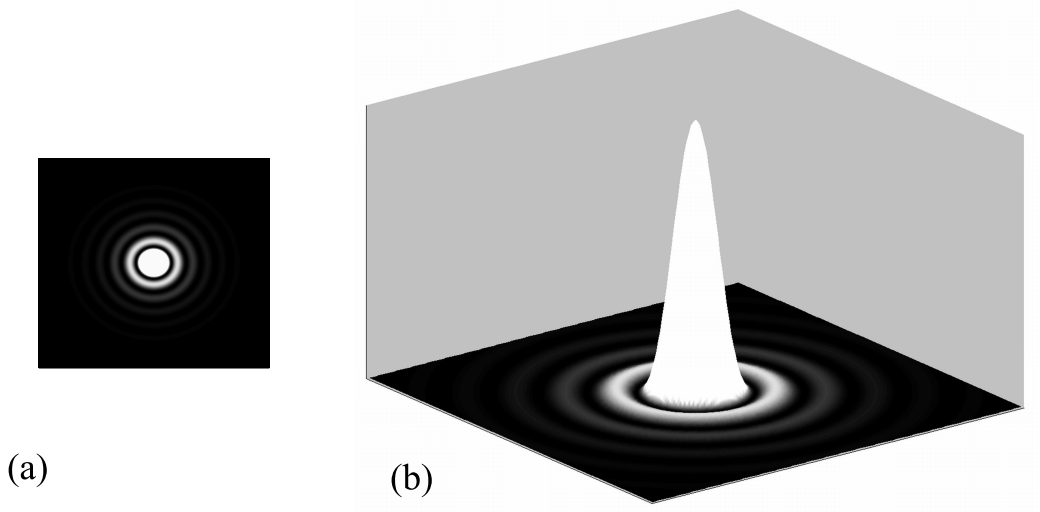


Abbildung 2.4.: Intensitätsverteilung bei Beugung des Lichts an kreisförmiger Blende. (a) Zweidimensionale Abbildung der Lichtintensität. (b) Dreidimensionale Darstellung des Intensitätsverlaufs mit dem Airy-Scheibchen, welches den überwiegenden Teil der Intensität beinhaltet, und mehreren konzentrischen Ringen, deren Intensität nach außen hin abnimmt.

Abb. 2.3 zeigt die Verbreiterung von Punkten in verschiedenen Abständen zur Linse, wobei bei optimalem Abstand z_0 keine Verbreiterung entsteht. Die Punkte in anderen Distanzen ($z_0 - \Delta z_1$ und $z_0 + \Delta z_2$) zur Brennebene erscheinen als Kreise mit dem Durchmesser D auf der Bildebene. Ist der Durchmesser eines solchen Kreises kleiner als die Größe eines Pixelelements, wird die Abbildung dennoch als ideal scharf wahrgenommen. Der Abstandsbereich, in dem Punkte als scharf wahrgenommen werden, wird als Schärfentiefe bezeichnet und hängt sowohl von der Kamerakonstante, der Brennweite und der Pixelgröße, als auch von dem Blendendurchmesser ab.

Durch die Veränderung des Blendendurchmessers wird auch direkt die Schärfentiefe beeinflusst: ein großer Blendendurchmesser liefert viel Licht zur Bildebene, verursacht aber eine kleine Schärfentiefe. Umgekehrt wird mit wenig Lichtund kleinem Blendendurchmesser, aber einem großen Schärfentiefebereich gearbeitet.

Physikalisch verursacht die Blende des Objektivs eine Beugung des Lichts (Pedrotti et al., 2005). Dadurch entsteht eine kleine Scheibe, die auch *Airy-Scheibchen* genannt wird, sowie konzentrische Ringe um das Scheibchen. Im Airy-Scheibchen ist nahezu die gesamte Beugungsintensität enthalten (siehe Abb. 2.4).

Dieses Verhalten ist mathematisch durch die PSF (*point spread function*) definiert,

welche die Intensitätsverteilung der Abbildung eines Punktes auf der Bildebene hinter einer realen Blende beschreibt. Diese rotationssymmetrische Verteilung kann mittels einer Bessel-Funktion erster Ordnung modelliert werden, wobei r dem euklidischen Abstand zur optischen Achse entspricht:

$$I(r) = I_0 \left(\frac{J_1(r)}{r} \right)^2 . \tag{2.9}$$

Um die Beschreibung zu vereinfachen, kann die Intensitätsverteilung in erster Näherung als gaußförmig angenommen werden, wobei die konzentrischen Ringe ausser Acht gelassen werden.

Stereo-Geometrie: Wie bereits im Kontext der Lochkamera erwähnt, gehen bei der Projektion eines 3D-Punktes auf die Bildebene Informationen verloren, hauptsächlich die Tiefeninformation des Punktes. Für die Applikationen, die in dieser Arbeit untersucht werden, ist aber die Tiefeninformation sehr wichtig und muss möglichst genau rekonstruiert werden.

Für den monokularen Fall gibt es Möglichkeiten, die Tiefeninformation nachträglich zu bestimmen. Zum einen kann die Entfernung zur Kamera von Objekten mit bekannten Geometriedaten über die Größe ihrer Abbildung ermittelt werden. Dieser Ansatz bedingt eine genaue Aussage über die Objektgeometrie und erreicht selbst dann nicht die Genauigkeiten, die im Rahmen dieser Arbeit von der Szenenrekonstruktion gefordert werden. Insbesondere trifft dies bei weit entfernten Objekten zu. Zum anderen macht man sich die Defokussierung zu Nutzen, welche außerhalb des Schärfentiefebereichs auftritt. Durch vorherige Kalibrierung kann der Zusammenhang zwischen PSF und Objektabstand ermittelt werden. Die erreichbaren Genauigkeiten für reale Sequenzen bei dieser Methode sind ebenfalls nicht ausreichend. Unter *Structure-from-Motion* sind Verfahren bekannt, die die Struktur der Szene aus einer Bildsequenz, die von einer sich bewegenden Kamera aufgenommen wurde, rekonstruieren (Ullman, 1979). Dabei benötigt man eine sich bewegenden Kamera, und dennoch ist die absolute Tiefengenauigkeit nicht ausreichend für die Anwendungen, welche im Rahmen dieser Arbeit untersucht werden.

Ähnlich wie im menschlichen Sehsystem, das aus zwei Augen besteht, wird daher eine zweite, räumlich versetzte Kamera benutzt, um Tiefeninformationen zurückzugewinnen. Durch Triangulation zwischen den korrespondierenden Abbildungen desselben 3D-Punktes in beiden Kamerabildern lässt sich die Entfernung des 3D-Punktes zur Kamera berechnen. Diese Methode bezeichnet man als Stereobildverarbeitung (engl.: Stereo-Vision). Für tiefgehendere Erläuterungen zu den Grundlagen der Stereobildverarbeitung wird an dieser Stelle auf Schreer (2007) verwiesen.

Die Anordnung der beiden Kameras, die zu den einfachsten Formeln zur Ermittlung der 3D-Punktkoordinaten führt, ist die Standard-Stereogeometrie oder auch Stereonormalfall genannt (siehe Abb. 2.5). Diese Anordnung ist ein Spezialfall der Epipolargeometrie.

Bei dieser Konfiguration sind die Bildebenen der beiden Kameras parallel. Die Verbindung der beiden optischen Zentren wird als Basislinie b bezeichnet. Die optischen Achsen

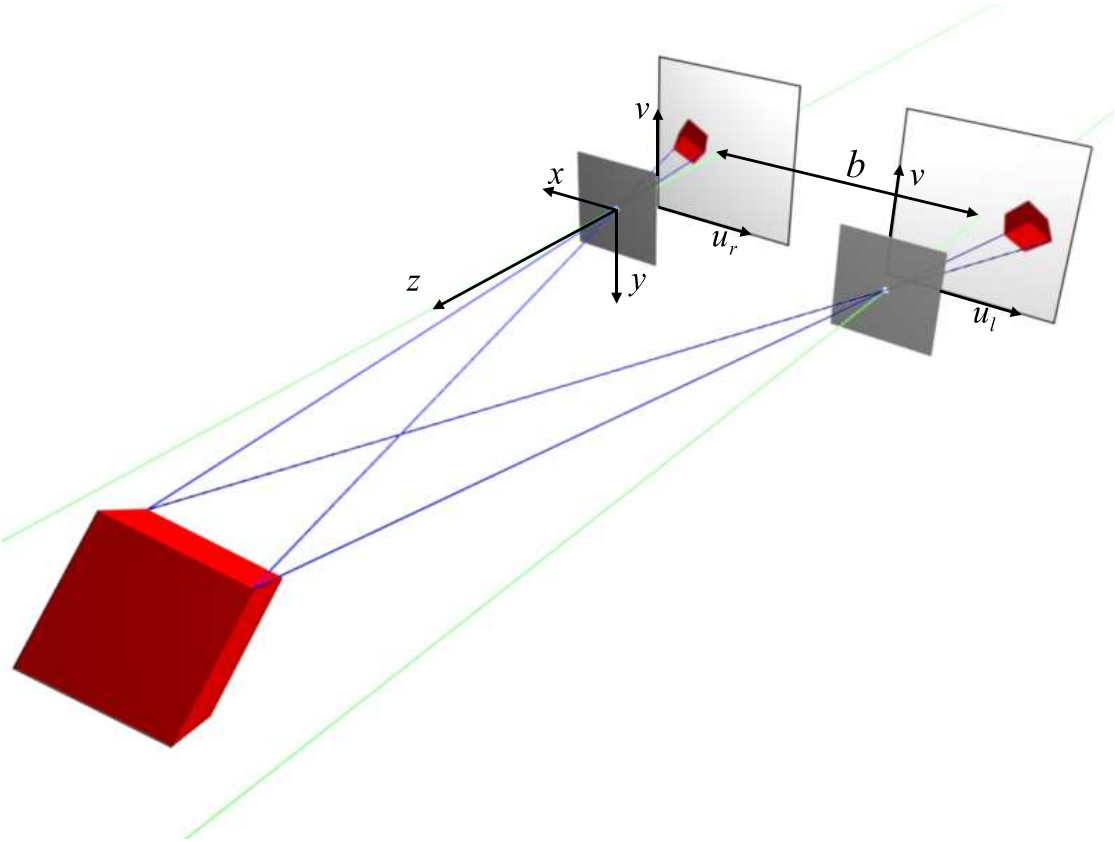


Abbildung 2.5.: Anordnung eines Stereokamerasystems in Standard-Stereogeometrie. Der Ursprung des Weltkoordinatensystems liegt im optischen Zentrum der rechten Kamera.

sind parallel und stehen senkrecht auf der jeweiligen Bildebene. Die Kamerakonstanten sind für beide Kameras gleich. Die Basislinie ist parallel zu den Bildzeilen und diese sind parallel zu den Epipolarlinien. Mechanisch ist diese Anordnung anspruchsvoll, weswegen man normalerweise die Kameras nur ungefähr in Standard-Stereogeometrie anordnet und später durch die Kalibirierung und Rektifizierung (siehe nächste Abschnitte) die aufgenommenen Bilder so manipuliert, als wären sie mit einer Standard-Stereogeometrie aufgenommen worden.

Das Epipolar-Constraint besagt, dass die Abbildungen desselben 3D-Punktes in den beiden Kamerabildern auf den korrespondierenden Epipolarlinien des Stereokamerasystems liegen müssen. Durch die Standard-Stereogeometrie kann somit die Korrespondenzsuche auf gleiche Bildzeilen im linken und rechten Kamerabild beschränkt werden, was die Komplexität der Suche verringert.

Die beobachtete Szene kann mithilfe von korrespondierenden Punkten im linken und rechten Kamerabild dreidimensional durch Berechnung der 3D-Punktkoordinaten rekonstruiert werden. Wie bereits erwähnt muss das Kamerasystem zunächst kalibriert und die

aufgenommenen Bilder anschließend rektifiziert werden (siehe nächster Abschnitt), um Bilder eines idealen Kamerasystems in Standard-Stereogeometrie zu erhalten. Anschließend werden in einer Korrespondenzsuche (siehe Kap. 2.1.2) die beiden Abbildungen desselben Szenenpunktes im linken und rechten Bild bestimmt. Dabei wird die horizontale Differenz zwischen den Abbildungen in den beiden Bildern als Disparität $d = u_r - u_l$ bezeichnet, wobei u_r die horizontale Pixelposition im rechten Kamerabild und u_l die horizontale Position im linken Bild definiert. Die Disparität gibt direkt Aufschluß über die Entfernung des 3D-Punktes zur Kamera: eine kleine Disparität beschreibt eine große Entfernung und umgekehrt. Der Zusammenhang ist allerdings nichtlinear (siehe auch Kap. 4.4.6). Die Pixelkoordinaten der beiden Bildpunkte zusammen mit den Parametern des Kamerasystem führen zu den 3D-Koordinaten des beobachteten Szenenpunktes im Weltkoordinatensystem W:

$$ {}^W x = \frac{f \cdot u_r}{d}, \quad {}^W y = \frac{f \cdot v}{d}, \quad {}^W z = \frac{b \cdot f}{d} \ , \tag{2.10} $$

wobei v die gemeinsame Bildzeile des linken und rechten Bildes ist. Es wird hier das optische Zentrum der rechten Kamera als Ursprung für das Weltkoordinatensystem gewählt.

Kalibrierung

Die Kalibrierung des Kamerasystems dient dazu, die intrinsischen und extrinsischen Parameter des Systems zu bestimmen. Dabei gehören Kamerakonstante, Pixelanzahl, Kamerahauptpunkt und Verzeichnungsparameter zu den intrinsischen Parametern und die Position und Orientierung der jeweiligen Kamera im Weltkoordinatensystem zu den extrinsischen Parametern.

Zunächst werden bei den meisten Methoden zur Kamerakalibrierung Bilder von einem bekannten Objekt, bei dem präzise Geometriedaten verfügbar sind (z.B. Schachbrettmuster), in verschiedenen Posen vor dem Kamerasystem aufgenommen. Das verwendete Verfahren benutzt ein planares Schachbrettmuster als Kalibrierobjekt, da sich die Abbildungen der Schachbrettecken darauf einfach und subpixelgenau extrahieren lassen. Anschließend werden mittels der vermessenen 2D-Koordinaten in den beiden Bildern die Pose des Objekts und die Kameraparameter optimiert. Dabei wird versucht, den Rückprojektionsfehler in den beiden Bildern zu minimieren, d.h. der Abstand zwischen der realen Position der Schachbrettecke im Bild und der Position, die im Pixelkoordinatensystem entsteht, wenn man die Ecke auf dem bekannten Objekt in geschätzter Pose vor der Kamera mit geschätzten Kameraparametern rückprojiziert. Die Pose des Objekts wird in dieser Optimierung zusammen mit den Kameraparametern geschätzt.

Grundsätzlich besteht die Möglichkeit, die Parameter durch die Lösung eines linearen Gleichungssystems zu bestimmen. Jedoch ist das nicht mehr möglich, wenn man auch die Verzeichnungsparameter mitschätzen will, da dadurch das Problem nichtlinear wird. Jedoch kann für eine Initialisierung der Parameter die lineare Methode benutzt werden und anschließend eine nichtlineare Optimierung auf der Basis dieses Ergebnisses gestartet werden.

In der vorliegenden Arbeit wird die Kalibriermethode nach Krüger (2007) verwendet. Vorteilhaft an dieser Methode ist das Kameramodell, da es die notwendigen Verzeichnungen beinhaltet. Außerdem wird ein automatischer Schachbretteckendetektor verwendet, was eine manuelle Korrespondenzbestimmung überflüssig macht. Dadurch kann in kurzer Zeit eine Vielzahl von Bildern des Schachbrettmusters ausgewertet werden, wodurch die Kalibrierung insgesamt genauer und robuster wird, was später Vorraussetzung für eine genaue 3D-Szenenrekonstruktion ist. Des Weiteren werden die gefundenen Schachbrettecken automatisch verknüpft und Ecken im Hintergrund ausgeblendet. Dadurch kann das Kamerasystem auch in beliebiger Umgebung kalibriert werden, unabhängig vom Hintergrund hinter dem Schachbrett.

Rektifizierung

Das Ziel der Rektifizierung ist es, aus den Bildern des kalibrierten Kamerasystems neue Bilder zu generieren, die den Bildern aus einem virtuellen Kamerasystem in Standard-Stereogeometrie entsprechen. Dazu werden die ursprünglichen Bilder so transformiert, dass die Epipolarlinien parallel zu den Bildzeilen sind und keinerlei Verzeichnungen mehr in den Bildern auftreten. Dies ist notwendig, da es mechanisch nicht möglich ist, ein Kamerasystem so präzise aufzubauen, dass die Anforderungen einer Standard-Stereogeometrie von vornherein erfüllt sind.

Die in dieser Arbeit verwendete Rektifizierung basiert auf dem Algorithmus nach Fusiello et al. (2000). Vorteilhaft ist hier die kompakte und effektive Berechnung der neuen Bilder, der Homographien (Abbildungsfunktionen zwischen alten und neuen Bildern) und der Parameter der virtuellen Kameras. Außerdem werden während der Berechnung die Transformationen zwischen ursprünglichen und virtuellen Kameras im Weltkoordinatensystem ausgegeben, welche für die Arbeiten in Abschnitt 4.4.9 wichtig sind.

In der ursprünglichen Version des Algorithmus nach Fusiello et al. (2000) werden keine Linsenverzeichnungen berücksichtigt. In der vorliegenden Arbeit wird daher ein erweiterter Algorithmus verwendet, der unverzeichnete Bilder ausgibt.

2.1.2. Stereobildverarbeitung

Wie bereits in Abschnitt 2.1.1 erwähnt, benutzt man für die Stereobildverarbeitung Bilder von zwei räumlich versetzten Kameras, wobei die Bilder zeitlich synchron aufgenommen werden, um die beobachtete Szene dreidimensional zu rekonstruieren. Dazu werden die beiden Bilder zunächst rektifiziert, um die Korrespondenzsuche nur auf einer Bildzeile in den beiden Bildern durchführen zu müssen. Anschließend werden korrespondierende Punkte in den Bildern gesucht, d.h. die Abbildung des gleichen 3D-Szenenpunktes in den beiden Stereobildern. Aus den Bildkoordinaten der beiden korrespondierenden Punkten läßt sich mithilfe der Kameraparameter die dreidimensionale Position des Punktes im Raum rekonstruieren.

Die Herausforderung für den Stereoalgorithmus ist demnach die Suche nach korrespondierenden Punkten im linken und rechten Bild, wofür verschiedenste Ansätze in der Literatur bekannt sind. Einige der nachfolgenden Veröffentlichungen geben einen

Überblick über den diesbezüglichen Stand der Forschung: (Barnard und Fischler, 1982; Dhond und Aggarwal, 1989; Koschan, 1993; Scharstein und Szeliski, 2002; Brown et al., 2003).

Die Korrespondenzsuche kann durch verschiedene Annahmen über die Beziehung der korrespondierenden Punkte zueinander unterstützt werden. Diese Annahmen oder Bedingungen werden in der Literatur als *Stereo-Constraints* bezeichnet. Das einfachste dieser Constraints ist das Compatibility-Constraint. Es bedingt, dass sich die Bildausschnitte, die den gleichen 3D-Punkt im linken und im rechten Bild repräsentieren, ähneln. Wie bereits beschrieben, werden aus Effizienzgründen Korrespondenzen nur entlang der Epipolarlinien gesucht, was als Epipolar-Constraint bezeichnet wird. Wird zudem die Suche auf einen bestimmten Disparitätsbereich beschränkt, so spricht man vom Disparity-Limit-Constraint. Vielfach verwandt wird auch das Uniqueness-Constraint (Marr und Poggio, 1979), welches besagt, dass ein Punkt in einem Bild nur genau einem Punkt im anderen Bild zugeordnet werden darf.

Des Weiteren ist das Ordering-Constraint zu erwähnen (Baker und Binford, 1981), welches eine gewisse räumliche Ordnung der Punkte zueinander vorschreibt. Ist ein Punkt in einem Bild links von einem anderen, so muss er in dem anderen Bild auch links von ihm sein. Bei Hinterschneidungen wird dieses Constraint allerdings oft verletzt und ist daher für Szenen mit großen Tiefensprüngen ungeeignet. Außerdem kann für bestimmte Anwendungen das Continuity-Constraint sinnvoll sein (Marr und Poggio, 1979). Es fordert eine gewisse Glattheit der aus Einzelpunkten bestehenden rekonstruierten Oberfläche. Dies verhindert Tiefensprünge und hat dadurch keine allgemeine Gültigkeit. Ein weniger häufig verwendetes Constraint, ist das Minimum-Weighted-Matching-Constraint (Fielding und Kam, 1997). Es beachtet auch die Gesamtfehlersumme über die ganze Bildzeile und versucht dabei möglichst viele Korrespondenzen zu etablieren. Eine weitere nützliche Bedingung ist der Left-Right-Consistency-Check (Fua, 1991). Dabei werden zunächst zu den Punkten im linken Bild korrespondierende Punkte im rechten Bild gesucht und danach umgekehrt. Schlussendlich werden nur Korrespondenzen übernommen, die in beiden Suchrichtungen gleichermaßen etabliert wurden. Somit werden vor allem Effekte die durch Verdeckungen entstehen minimiert. Weitere Stereo-Constraints sind in der Literatur zu finden: (Schreer, 2007) oder (Koschan, 1993).

Strukturen, die sich im Bild wiederholen sind ein grundsätzliches Problem bei der Korrespondenzbestimmung. Es treten dabei Mehrdeutigkeiten auf, die zu falschen Disparitäten und somit zu falschen Tiefenwerten führen. Besonders bei lokal arbeitenden Verfahren, aber auch bei globalen Verfahren führt dies zu falschen 3D Punkten. In der vorliegenden Arbeit wird dieses Problem noch genauer betrachtet (siehe Kap. 6.1).

Grundsätzlich können Stereo-Algorithmen in zwei Gruppen unterteilt werden, abhängig davon, wie groß der aktuell betrachtete Bereich der Bilder ist. Man unterscheidet dabei lokale und globale Verfahren.

Lokale Verfahren: Bei lokalen Verfahren werden immer nur einzelne Bildausschnitte betrachtet, jedoch nie das ganze Bild bzw. die komplette Bildzeile.

Eine der einfachsten Methoden Korrespondenzen in den beiden Stereobildern zu fin-

den, ist das Blockmatching Horn (1986). Dabei wird zu einem Ausschnitt (Block) im linken Bild I_l ein passender Ausschnitt im rechten Bild I_r gesucht. Dabei können verschiedene Vergleichsmaße benutzt werden. Beispiele für diese Vergleichmaße sind: der normierte Kreuzkorrelationskoeffizient (normalized cross-correlation (NCC)):

$$c_{\mathrm{NCC}} = \frac{\sum_{u_w,v_w}(I_l(u,v)-\overline{I_l})\cdot(I_r(u+d,v)-\overline{I_r})}{\sqrt{\sum_{u_w,v_w}(I_l(u,v)-\overline{I_l})^2}\cdot\sqrt{\sum_{u_w,v_w}(I_r(u+d,v)-\overline{I_r})^2}}, \tag{2.11}$$

die Sum-of-Absolute-Differences (SAD):

$$c_{\mathrm{SAD}} = \sum_{u_w,v_w} |I_l(u,v) - I_r(u+d,v)|, \tag{2.12}$$

die Sum-of-Squared-Differences (SSD):

$$c_{\mathrm{SSD}} = \sum_{u_w,v_w} (I_l(u,v) - I_r(u+d,v))^2, \tag{2.13}$$

oder die normalisierte Sum-of-Squared-Differences (NSSD):

$$c_{\mathrm{NSSD}} = \sum_{u_w,v_w} \left(\frac{I_l(u,v)-\overline{I_l}}{\sqrt{\sum_{u_w,v_w}(I_l(u,v)-\overline{I_l})^2}} - \frac{I_r(u+d,v)-\overline{I_r}}{\sqrt{\sum_{u_w,v_w}(I_r(u+d,v)-\overline{I_r})^2}} \right)^2, \tag{2.14}$$

wobei die Summe über u_w und v_w das Vergleichsfenster definiert und d die Disparität, I_l das linke Bild und I_r das rechte Bild repäsentieren. Ein weiteres Vergleichsmaß wird von Birchfield und Tomasi (1998) vorgestellt. Das vorgestellte Maß erhält eine zusätzliche Robustheit dadurch, dass es invariant gegen Sampling-Effekte ist. Diese Art der Kostenfunktion wird bei zuletzt veröffentlichten Stereoalgorithmen häufig genutzt, da sie sowohl robust, als auch schnell zu berechnen ist.

Ein Beispiel für einen erweiterten Blockmatching-Algorithmus wird von Franke und Joos (2000) vorgestellt (siehe Abb. 2.6). Hier werden verschiedene Probleme des ursprünglichen Algorithmus behoben oder zumindest verringert. Durch einen Interestoperator werden Blöcke mit konstanter Intensitätsverteilung, was aufgrund des Aperture-Problems (Marr und Ullman, 1981) zu keiner eindeutigen Korrespondenz führt, aus der Korrespondenzsuche ausgeschlossen. Dabei dient ein vertikales Kantenfilter (z.B. Prewitt-Filter) zur Differenzierung zwischen Blöcken, die eine zuverlässige Korrespondenzschätzung erlauben und Blöcken, bei denen ein zuverlässiges Resultat prinzipiell ausgeschlossen ist. Zum anderen wird dem Problem des hohen Rechenaufwands durch eine Pyramidenstruktur entgegengewirkt. Auf höheren Pyramidenstufen werden wenige Vergleiche durchgeführt, um eine initiale Tiefenschätzung zu erhalten. Diese wird dann über die Pyramidenstufen nach unten propagiert und weiter verfeinert, was insgesamt zu einer Reduzierung des Berechnungsaufwands führt. Außerdem werden beim ursprünglichen Blockmatching lediglich pixel-genaue Disparitäten bestimmt, was gerade in größeren Entfernungen zu einer ungewollten, groben Quantisierung der Tiefeninfor-

mationen führt. Um diese Tiefeninformationen zu verfeinern, werden die verfügbaren Disparitätswerte nachträglich über eine Parabelfunktion interpoliert und so das subpixelgenaue Maximum der Vergleichsfunktion ermittelt.

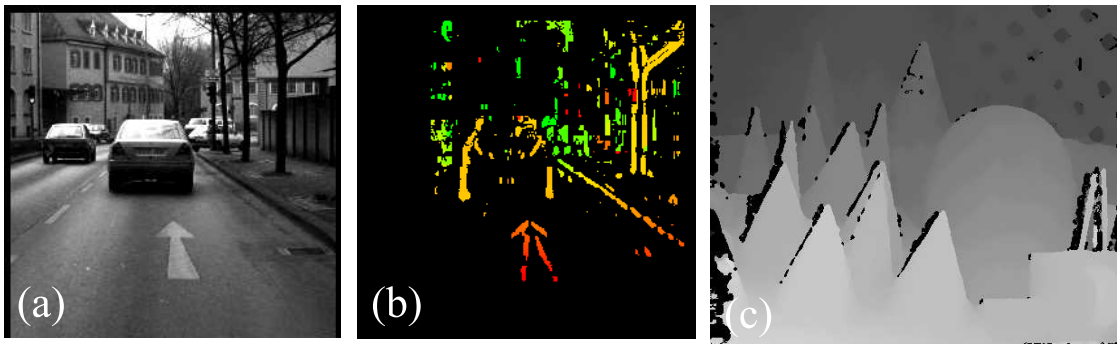


Abbildung 2.6.: Beispiele für Stereobildverarbeitungsergebnisse. (a) Eingangsbild aus der Veröffentlichung von Franke und Joos (2000). (b) Ergebnis zu Bild (a). 3D-Punkte werden nur an vertiaklen Kanten aufgrund des verwendeten Interestoperators extrahiert. Warme Farben bedeuten eine geringe Distanz zur Kamera, wohingegen kalte Farben große Distanzen repräsentieren. (c) Ergebnis von Hirschmüller (2005). Es zeigt sich eine wesentlich dichtere Rekonstruktion der beobachteten Szene. Je heller der Grauwert umso geringer die Distanz zur Kamera.

Ein Problem, welches bei der Interpolation der Vergleichmaße entsteht, ist das sogenannte Pixel-Locking. Dieser Effekt beschreibt das Phänomen, dass die interpolierten Werte unnatürlich häufig ganzzahlig bleiben. Dabei tritt der Effekt bei verschiedenen Vergleichmaßen auch verschieden stark auf (bei SSD beispielsweise stärker als bei SAD (Matthies et al., 2007)). Der Grund für diesen Effekt blieb bislang leider unentdeckt, jedoch gibt es Ansätze, den Effekt zu reduzieren oder zu umgehen. Ein Ansatz ist dabei die Idee des Lucas-Kanade Verfahrens zur optischen Fluss-Bestimmung (siehe Kap. 2.1.3) zu benutzen, um die Disparität zu verfeinern. Es wird zusätzlich eine affine Transformation zwischen den Vergleichsblöcken erlaubt, wodurch das Pixel-Locking nahezu vollständig verhindert wird (Stein et al., 2006). Die Experimente beschränken sich allerdings auf großflächige Ebenen, wodurch eine allgemeine Aussage ausbleibt.

Ein weiteres Problem bei Blockmatching Algorithmen besteht in der Annahme, dass die Disparität über den kompletten Block konstant ist. Diese Annahme wird jedoch an Objektkanten verletzt. Es entsteht eine Verschmierung der Kanten im Tiefenbild, wobei die korrekte Kante um etwa die halbe Blockgröße nach links und rechts verschmiert wird. Zur Reduktion dieses Effekts wird von Kanade und Okutomi (1994) ein Ansatz vorgestellt, wobei die einzelnen Blöcke in Größe und Form variieren können, je nach Charakteristik der Disparitätswerte in der Umgebung. Dadurch werden die Fehler an Kanten reduziert, allerdings ist dieses Verfahren sehr rechenintensiv. Ein anderer Ansatz von Hirschmüller et al. (2002) benutzt mehrere Blöcke, die gegenseitig ein konsistentes Ver-

halten absichern. Außerdem werden an Tiefensprüngen nachträglich die Blöcke halbiert und mittels der beiden Blöcke versucht, den Tiefensprung möglichst scharf darzustellen, indem die Trennlinie zwischen den beiden halben Blöcken optimiert wird.

Im Gegensatz zu den bislang vorgestellten Stereoverfahren gibt es eine weitere Kategorie, die nicht direkt auf den Intensitätswerten im Bild arbeiten, sondern zunächst sogenannte Features extrahieren und diese dann über die gesamte Bildzeile miteinander vergleichen. Diese Features können Ecken, Kanten, Liniensegmente o.ä. sein, oder aber eine abstrahierte Beschreibung eines Bildausschnitts wie z.B. bei der Rank- oder Census-Transformation. Der Nachteil dieser Verfahren ist die Dichte der resultierenden Tiefenkarte. Nur dort, wo Features im Bild präsent sind, können auch Korrespondenzen gefunden werden. Der große Vorteil ist jedoch die Verarbeitungsgeschwindigkeit, die für gewöhnlich sehr hoch ist.

Venkateswar und Chellappa (1995) benutzten eine hierarchische Feature-Anpassung, welche aus vier Typen von Features besteht: Linien, Kanten, Ecken und Flächen. Dabei wird von der höchsten Stufe, den Flächen nach unten bis hin zu den Linien durchpropagiert und somit eine *Grob-nach-Fein-Suche* implementiert.

Die Klassifizierung der Pixel anhand ihrer vier direkten Nachbarn wird von Franke und Kutzbach (1996) vorgeschlagen. Dabei können die Pixelnachbarn entweder dunkler oder gleich hell oder heller als das betrachtete Pixel sein, wodurch sich insgesamt $3^4 = 81$ Klassen ergeben. Durch die Klassenübereinstimmung werden Korrespondenzen auf der Epipolarlinie gebildet. Gibt es mehrere mögliche Korrespondenzen wird die Korrespondenz mit dem kleinsten resultierenden Disparitätswert benutzt, um Phantomobjekte mit geringer Distanz zur Kamera zu verhindern.

Sehr ähnlich dazu wird auch bei der Rank-Transformation lediglich gezählt wieviele Pixel in der Nachbarschaft dunkler als das betrachtete Pixel sind. Eine Erweiterung davon ist die Census-Transformation, wobei hier alle Differenzen der Nachbarn zum Zentrum wiederum in die drei Kategorien dunkler, gleich hell, heller unterteilt werden und diese Information in einem Bitstring gespeichert wird. Der Vergleich dieser Bitstrings erfolgt über die Hamming-Distanz, welche auswertet, wieviele Bits im String gleich sind. Zabih und Woodfill (1994) stellen beide Verfahren vor und verglichen diese.

Basierend auf der gleichen Grundidee werden von Stein (2004) die Koeffizienten aus der Census-Transformation zum Feature-Vergleich benutzt, um eine Berechnung des optischen Flusses durchzuführen. Die gleiche Technik kann jedoch auch für Stereoberechnungen benutzt werden. Hier steht auch wiederum die schnelle Berechnung im Vordergrund.

Die bisher vorgestellten Feature-basierten Verfahren sind lediglich in der Lage, pixelgenaue Korrespondenzen zu etablieren. Ein Ansatz, der es auch ermöglicht subpixel genaue Disparitäten zu etablieren, wird von Wöhler und Krüger (2003) vorgestellt. Hier werden im Bild zusammenhängende Strukturen mittels *Binary-Connected-Components-Analysis* gesucht und deren Kontur durch B-Splines beschrieben. Durch den Vergleich der Konturen im Linken und rechten Bild können Korrespondenzen ermittelt werden.

Globale Verfahren: Globale Verfahren zeichnen sich dadurch aus, dass sie eine komplette Epipolarlinie oder direkt das ganze Bild betrachten.

Ein erster globaler Ansatz für die Berechnung eines Disparitätsvektors für eine komplette Bildzeile geht auf Horn (1986) zurück. Hier wird der direkte Vergleich von Intensitätswerten mit einer Glattheitsbedingung für benachbarte Disparitäten in einem Variationsansatz verknüpft. Folgendes Funktional wird dabei minimiert:

$$e = \sum_{u,v} \left[\left(\nabla^2 d(u,v) \right)^2 + \lambda (I_l(u + d(u,v)/2, v) - I_r(u - d(u,v)/2, v))^2 \right]. \qquad (2.15)$$

Durch die Minimierung der zweiten Ableitung der Dispritäten werden Sprünge in der Tiefenschätzung vermieden. Der Gewichtungsfaktor λ reguliert diese Bedingung gegenüber dem direkten Intensitätsvergleich.

Die Methode der Dynamischen Programmierung wird in vielen Veröffentlichungen und Lehrbüchern (Forsyth und Ponce, 2002; Hartley und Zisserman, 2004) als globales Stereo-Verfahren empfohlen. Hierbei wird das Ordering-Constraint dazu benutzt, eine effiziente Implementierung der Disparitätsbestimmung zu erreichen. Vorteil dieses Verfahrens ist es, dass Bereiche in denen keine Information über die Disparität vorliegt, implizit aufgefüllt werden.

Darauf aufbauend gibt es auch Verfahren, die den daraus resultierenden Disparitätsraum (auch DSI für *disparity space image*) zusätzlich über mehrere bzw. alle Bildspalten optimieren. Roy und Cox (1998) benutzten beispielsweise Graph-Cuts, genauer gesagt das Maximum-Flow Verfahren aus der Graphen-Theorie dazu, eine globale Disparitätskarte zu generieren.

Sowohl von Faugeras und Keriven (1998) als auch von Slesareva et al. (2005) werden Variationsansätze benutzt, um die Disparitätskarte sowohl mithilfe eines Ähnlichkeitsmaßes, als auch durch globale Glattheitsbedingungen zu berechnen. Die Optimierung basiert bei Faugeras und Keriven (1998) auf einem Level-Set Ansatz, wohingegen von Slesareva et al. (2005) eine Grob-nach-Fein-Suche in einer Auflösungspyramide benutzt wird, um die berechneten Euler-Lagrange Gleichungen zu optimieren.

Ein Verfahren, welches sich zwischen globalen und lokalen Verfahren definiert, ist das Semi-Global-Matching (SGM, (Hirschmüller, 2005), siehe Abb. 2.6). Dabei wird grundsätzlich versucht eine globale 2D-Kostenfunktion zu minimieren, indem eine Vielzahl von eindimensionalen Kostenfunktionen optimiert werden. Dabei ist die Kostenfunktion wie folgt aufgebaut:

$$e(d) = \sum_{p} \left[c(p, d_p) + \sum_{q \in N_p} F_1 T(|d_p - d_q| = 1) + \sum_{q \in N_p} F_2 T(|d_p - d_q| > 1) \right]. \qquad (2.16)$$

Der erste Term $c(p, d_p)$ ist die pixelweise Kostenfunktion und kann ein beliebiges Vergleichsmaß sein. Sie wird für die aktuelle Pixelposition p und Disparität d_p berechnet.

Die Funktion T gibt 1 zurück, wenn die Bedingung erfüllt ist, sonst 0. Dadurch bewirkt der zweite Term eine Bestrafung für kleine Disparitätsunterschiede zu den benachbarten Disparitätswerten d_q mit dem Faktor F_1. Je kleiner der Werte für den vorzugebenden Parameter F_1 gewählt wird, desto mehr werden gekrümmte Oberflächen zugelassen. Genauso werden große Disparitätssprünge über den dritten Term mit dem Faktor F_2 bestraft, welcher ebenfalls vorzugeben ist, wobei dieser vom lokalen Intensitätgradienten abhängt. Durch diese Vorgehensweise entstehen scharfe Tiefensprünge, wie sie in anderen Verfahren oft vermisst werden. Das Verfahren berechnet diese Kostenfunktion über eindimensionalen Pfade im Bild in acht oder 16 Richtungen zu jedem Pixel mittels dynamischer Programmierung. Die Kosten aller Pfade werden anschließend für jedes Pixel und jede Disparität addiert. Am Ende wird über eine *Winner-Takes-All*-Entscheidung eine globale Tiefenkarte erzeugt. Ungültige Disparitätswerte werden gelöscht und über ihre Nachbarschaft interpoliert.

Sun et al. (2005) stellen die Korrespondenzsuche in einem Markov-Netzwerk dar und lösen dieses anschließend iterative mittels Belief Propagation. Die Ergebnisse gehören neben den Graph-Cut-Ansätzen laut Scharstein und Szeliski (2002) zu den genauesten Verfahren.

In verschiedenen Veröffentlichungen werden die unterschiedlichen Ansätze miteinander verglichen. Scharstein und Szeliski (2002) stellen eine Vielzahl von dichten Stereo-Verfahren vor und evaluieren anhand von Ground-Truth Daten deren Ergebnisse. Außerdem werden in Verbindung damit die Ground-Truth Daten und Implementierungen von Standard-Verfahren im Internet bereitgestellt. Dadurch steht ein vereinheitlichter Benchmark-Datensatz zur Verfügung, mit dem Autoren ihre Algorithmen untereinander vergleichen können.

Drei Optimierungsstrategien (lokale, semi-global und global), sowie sechs Kostenfunktionen werden von Hirschmüller und Scharstein (2007) miteinander verknüpft und verglichen. Speziell für Fahrerassistenzsysteme ist die Evaluierung von van der Mark und Gavrila (2006) gedacht. Hier liegt das Hauptaugenmerk auf dem Verhalten bei großen Tiefensprüngen, wie sie im Straßenverkehr vorkommmen. Dabei zeigt sich, dass für die verwendeten Sequenzen das Verfahren nach Hirschmüller et al. (2002), also ein lokales Verfahren mit Left-Right-Consistency-Check und expliziter Behandlung von Objektkanten, die besten Ergebnisse liefert.

2.1.3. Optischer Fluss

Ebenso wie die Stereobildverarbeitung die Verknüpfung von zwei räumlich versetzt aufgenommenen Bildern beschreibt, werden durch die Berechnung des optischen Flusses zwei zeitlich versetzt aufgenommene Bilder miteinander verknüpft. Dadurch lassen sich Informationen über die Bewegung der beobachteten Objekte ableiten bzw. die Kamerabewegung (auch „Ego-Motion“ genannt) in der Szene ermitteln. Es sind im Kontext des optischen Flusses verschiedene Zusammenhänge zwischen Bildebene und beobachteter Szene definiert. Das Bewegungsfeld (engl.: Motion-Field) oder auch Verschiebungsvektorfeld genannt, beschreibt die auf die Bildebene der Kamera projizierte Bewegung des dreidimensionalen Objekts. Der optische Fluss bescheibt hingegen die sichtbare Verschie-

bung des Bildinhalts von einem zum nächsten Bild. Der Szenenfluss ist die wahrgenommene dreidimensionale Bewegung des Objekts im Raum, wodurch der optische Fluss die Projektion des Szenenflusses ist.

Die Verschiebung des 3D-Punktes w_i von einem Zeitschritt (t_0) zum nächsten (t_1) verursacht eine Verschiebung des projizierten Punktes m_i im Bild. Fasst man diese Verschiebungsvektoren über alle abgebildeten 3D-Punkte zusammen, so erhält man das Verschiebungsvektorfeld oder Bewegungsfeld in der Bildebene.

Das Mirror-Ball-Experiment von Horn (1986) zeigt, dass Bewegungsfeld und optischer Fluss nicht zwingend gleich sind. Angenommen wird eine rotierende, texturlose, perfekte Kugel, die in zwei aufeinanderfolgenden Bildern ein Bewegungsfeld in der Bildebene, jedoch keinerlei optischen Fluss verursacht. Die Kugel dreht sich zwar, jedoch wird keine Veränderung im Bild dadurch verursacht, weshalb kein optischer Fluss entsteht.

Ein Phänomen, welches bereits von Wallach (1935) erwähnt wurde, ist der sogenannte Barber-Pole Effekt (siehe Abb. 2.7): Hier wird sowohl der Unterschied zwischen Bewegungsfeld und optischem Fluss veranschaulicht, als auch das Apertur-Problem (Marr und Ullman, 1981). Unter dem Apertur-Problem versteht man i.A. die Mehrdeutigkeit der Bewegungsschätzung, die sich u.a. daraus ergibt, dass man keine oder nur einen Teil der Objektkontur im Bild sieht.

Zusammengefasst verhindert also sowohl die Projektion von dreidimensionalen Objekten und deren ebenfalls dreidimensionaler Bewegung auf zweidimensionale Bilder, als auch das Apertur-Problem eine eindeutige Bestimmung der dreidimensionalen Bewegung von Objekten aus zwei zeitlich aufeinanderfolgenden Bildern. Was aus den beiden Bildern also bestimmt wird, ist der sogenannte optische Fluss, also das Verschiebungsvektorfeld, welches zur Veränderung der Intensitätsverteilung über die Zeit im Bild führt.

Über die Methoden zur Bestimmung des optischen Flusses aus Bildsequenzen gibt es lediglich Literaturüberblicke aus dem letzten Jahrzehnt, z.B. Barron et al. (1994). Liu et al. (1998) berücksichtigen neben der Genauigkeit die algorithmische Effizienz der Verfahren.

Der optische Fluss ist mathematisch durch seine beiden Bewegungskomponenten $\dot{u}$ und $\dot{v}$ definiert, wobei $\dot{u} = du/dt$ horizontale Bewegungen und $\dot{v} = dv/dt$ vertikale Bewegungen beschreibt. Angenommen, dass ein Oberflächenelement über die Zeit eine konstante Lichtintensität in Richtung Kamera reflektiert, kann das *Optical Flow Constraint* wie in Gleichung 2.17 definiert werden (Horn, 1986). Hierbei wird definiert, dass die Intensität I vor und I' nach der Verschiebung um $\dot{u}$ und $\dot{v}$ gleich bleibt, wobei Δt das Zeitintervall der Verschiebung angibt.

$$I(u + \dot{u}\Delta t, v + \dot{v}\Delta t, t + \Delta t) = I'(u, v, t) \tag{2.17}$$

Durch die Taylor-Reihenentwicklung kann das Optical Flow Constraint wie folgt umgeformt werde:

$$I(u, v, t) + \Delta u \frac{\partial I}{\partial u} + \Delta v \frac{\partial I}{\partial v} + \Delta t \frac{\partial I}{\partial t} + e = I'(u, v, t). \tag{2.18}$$

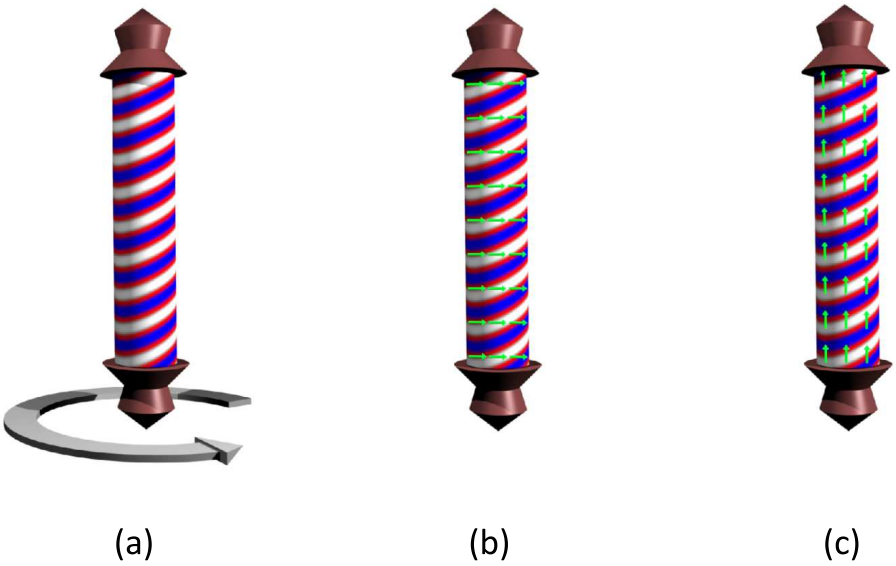


Abbildung 2.7.: Beispiel für den Unterschied zwischen Bewegungsfeld und optischem Fluss. (a) In Pfeilrichtung rotierende Barber-Pole. (b) Das durch die Bewegung entstandene Bewegungsfeld. Das Bewegungsfeld ist die Projektion der dreidimensionalen Bewegung auf die Bildebene. Die Bewegungsvektoren zeigen nach rechts. (c) Der durch die Bewegung entstandene optische Fluss. Der optische Fluss ist die Verschiebung der Intensitäten im zwei aufeinanderfolgenden Bildern. Die optischen Fluss-Vektoren zeigen auf Grund der diagonalverlaufenden Textur nach oben.

Diese Gleichung wird als *Gradient Constraint Equation* bezeichnet, wobei der Term e die Terme zweiter und höherer Ordnung für $\Delta u, \Delta v$ and Δt beinhaltet. Meist wird der Term e vernachlässigt und die Ableitungen der Intensität als I_u, I_v und I_t bezeichnet:

$$\nabla I(u,v,t) \cdot (\dot{u},\dot{v},0) + I_t(u,v,t) = 0, \text{wobei } \nabla I = \begin{bmatrix} I_u \\ I_v \\ I_t \end{bmatrix}. \tag{2.19}$$

Grundsätzlich werden in der Literatur, ähnlich wie bei der Stereoanalyse, zwei Arten von Ansätze verfolgt: die lokalen Verfahren und die globalen Verfahren.

Lokale Verfahren: Bei lokalen Verfahren wird versucht, in den beiden Bildern korrespondierende Features oder korrespondierende Bildausschnitte zu finden, wobei lediglich lokale Nachbarschaften im Bild berücksichtigt werden, jedoch nie das ganze Bild. Der zweidimensionale Verschiebungsvektor zwischen den korrespondierenden Bildpunkten wird als Flussvektor bezeichnet. Über das ganze Bild betrachtet entsteht so, je nach

Verfahren, ein spärliches bzw. dichtes Verschiebungsvektorfeld.

Lucas und Kanade (1981) benutzten die Gradient Constraint Gleichung (Gleichung 2.18) zusammen mit einer Gewichtungsfunktion, um den optischen Fluss zu bestimmen. Dabei wird angenommen, dass in lokaler Nachbarschaft der optische Fluss konstant ist. Dadurch wird das Problem, dass zu einem zweidimensionalen Flussvektor nur ein Messwert verfügbar ist, in ein überbestimmtes Gleichungssystem überführt. Für einen Flussvektor sind alle Messwerte der lokalen Nachbarschaft verfügbar, die jedoch über die Gewichtsfunktion derart verknüpft sind, dass das Zentrum der Nachbarschaft die höchste Relevanz bekommt. Durch die Least-Squares Lösung des Gleichungssystems können theoretisch an allen Stellen Flussvektoren berechnet werden. Nachteile des Verfahrens sind zum einen die Annahme, dass in der lokalen Nachbarschaft ein konstanter Fluss angenommen wird und zum anderen, dass eine konstante Intensität in der lokalen Nachbarschaft wiederum zum Aperturproblem führt. Dieses Verfahren ist auch Basis für den sogenannten KLT-Tracker. Dieser Trackingalgorithmus besteht zum einen aus dem beschriebenen Flussverfahren und zum anderen aus einer Eckenextraktion basierend auf der Veröffentlichung von Tomasi und Kanade (1991). Dabei wird versucht, den Verschiebungsvektor nur dort zu berechnen, wo es sinnvoll ist, nämlich an Ecken im Bild. Das Ergebnis sind nur einzelne Flussvektoren, welche jedoch schnell und genau zu berechnen sind.

Anandan (1989) verwendet die SSD (Sum-of-Squared-Differences), um Bildausschnitte in den beiden aufeinanderfolgenden Bildern einander zuzuordnen und so ein Verschiebungsvektorfeld zu berechnen:

$$SSD(u, v, \dot{u}, \dot{v}) = \sum_{j=-n}^{n} \sum_{i=-n}^{n} \left[I_{t_0}(u+i, v+j) - I_{t_1}(u + \dot{u}\Delta t + i, v + \dot{v}\Delta t + j) \right]^2 \ . \quad (2.20)$$

Über eine Suche nach dem kleinsten SSD-Wert kann für (u, v) die beste Verschiebung bestimmt werden.

Die Zuordnung der Bildauschnitte wird über eine Pyramidenstrategie realisiert, wobei zunächst in einer groben Auflösung Korrespondenzen gesucht werden, die dann an Stufen höherer Auflösung weiterpropagiert werden und dort das Verschiebungsvektorfeld verfeinert wird. Außerdem wird von Anandan (1989) ein Glattheitsterm ähnlich dem von Horn und Schunck (1980) benutzt.

Stein (2004) verwendet die Census-Transformation, um optische Fluss-Informationen zu generieren. Dazu wird das Bild zunächst nach geeigneten Bildausschnitten durchsucht. Diese Ausschnitte werden als Census-Koeffizienten repräsentiert und in einer Hashtabelle abgelegt. Durch die Hashtabelle lässt sich nun nach Korrespondenzen im zweiten Bild suchen und somit Flussvektoren etablieren. Diese Methode ist numerisch sehr effizient und sowohl für kleine als auch für sehr große Verschiebungen geeignet. In Abb. 2.8a wird ein Beispiel für das Berechnungsergebnis des Algorithmus gezeigt.

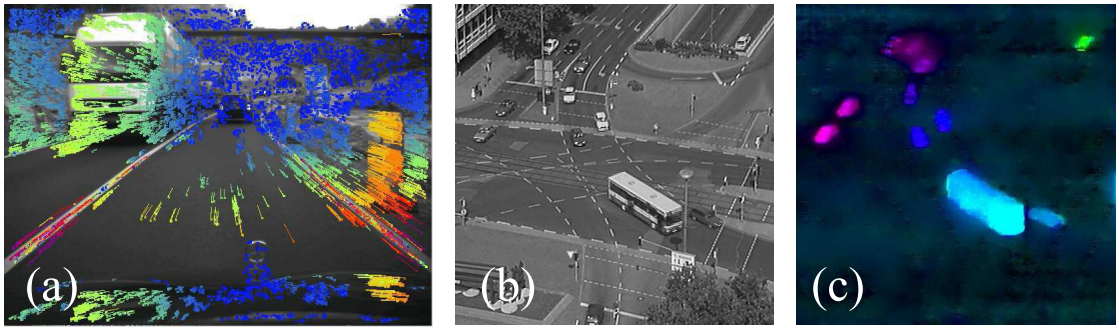


Abbildung 2.8.: Beispiele für die optische Fluss-Berechnung. (a) Ergebnis aus der Veröffentlichung von Stein (2004). Die Flussvektoren sind farbig dargestellt, wobei wärmere Farben schnellere Bewegungen repräsentieren. (b) Eingangsbild einer Testsequenz (Zach et al., 2007). (c) Dazugehöriges Ergebnis (Zach et al., 2007). Hier zeigt die Farbe die Richtung der Bewegung und die Helligkeit die Geschwindigkeit.

Globale Verfahren: Die globalen Verfahren erzielen ein dichtes Verschiebungsvektorfeld und betrachten dabei Bildpunkte aus beiden Bildern auf einmal. Es wird dabei versucht, mittels Regularisierung bzw. Variationsansätzen zusammen mit Glattheitsbedingungen die Mehrdeutigkeiten in einigen Regionen durch die Betrachtung der Umgebungsinformationen aufzulösen.

Horn und Schunck (1980) nutzten diese Gleichung in Kombination mit einer globalen Glattheitsbedingung als Funktional zur optischen Fluss Berechnung. Das globale Fehlerfunktional ist dabei

$$e_{\mathrm{HS}} = \int\int \left[(\nabla I \cdot (u, v, 0) + I_t)^2 + \lambda \left(|\nabla \dot{u}|^2 + \nabla \dot{v}|^2 \right) \right] dudv, \tag{2.21}$$

wobei

$$\nabla \dot{u} = \frac{\partial^2 \dot{u}}{\partial u^2} + \frac{\partial^2 \dot{u}}{\partial v^2} \text{ und } \nabla \dot{v} = \frac{\partial^2 \dot{v}}{\partial u^2} + \frac{\partial^2 \dot{v}}{\partial v^2} \ . \tag{2.22}$$

Das Fehlerfunktional e_{HS} des Regularisierungsansatzes besteht aus dem Datenterm, welcher die Intensitätsableitungen erster Ordnung beinhaltet und dem Glattheitsterm, welcher die höheren Ableitungen der Bewegung beinhaltet. Diese sind über den Regularisierungsparameter λ verknüpft. Das Fehlerfunktional wird iterativ mittels Euler-Lagrange-Gleichungen gelöst.

Durch die Regularisierung wird der optische Fluss in strukturlosen Regionen durch Interpolation der Bewegungsinformationen der umliegenden Regionen geschätzt.

Lange Zeit wurden an dieser Stelle keine großen Fortschritte verzeichnet, bis vor wenigen Jahren einige Veröffentlichungen (u.a. (Brox et al., 2004), (Bruhn, 2006) und (Bruhn et al., 2005)) zu einem neuen Meilenstein in der optischen Fluss-Berechnung führten. Ein neuer Meilenstein wurde dabei sowohl im Bezug auf die Genauigkeit der Ergebnisse als

auch in der Berechnungsgeschwindigkeit erreicht. Die Grundlage dafür bildet nach wie vor der Regularisierungsansatz in der Formulierung von Horn und Schunck (1980). Außerdem wird die Annahme von Lucas und Kanade (1981) benutzt, um lokale und globale Einflüsse zu kombinieren (Bruhn et al., 2005). Des Weiteren werden zusätzliche Constraints dazu benutzt, um das Ergebnis auf der mathematischen Seite zu verbessern. Zusätzlich zur Konstanz des Grauwerts, wie von Horn und Schunck (1980) beschrieben, wird auch eine Konstanz der räumlichen Grauwertableitung gefordert. Außerdem werden Diskontinuitäten explizit im Flussfeld erlaubt und während der Optimierung behandelt. Zusätzlich werden große Verschiebungen durch eine Coarse-to-Fine-Warping-Strategie berücksichtigt. Dabei wird auf mehreren Stufen einer Auflösungspyramide gerechnet und die Zwischenergebnisse zwischen den Stufen mittels Warping weitergegeben. Außerdem führt diese Strategie zu einer weiteren Verbesserung der Robustheit hinsichtlich Intensitätsrauschen, ebenso wird die Genauigkeit erhöht. Die Verarbeitung zwischen den Pyramidenstufen wird durch eine sogenannte Mehrgitterstrategie realisiert. Dies ermöglicht eine höhere Effizienz der Berechnung.

Ein ähnlicher Ansatz wird auch von Zach et al. (2007) verwendet (siehe Abb. 2.8), wobei hier die L^1-Norm als Vergleichsmaß benutzt wird. Zusätzlich wird eine Unterscheidung zwischen Struktur und Textur im Bild verwendet, um etwas genauere Ergebnisse zu erzielen. Die Lösung erfolgt über einen *Total-Variation*-Ansatz, welcher iterativ, ähnlich den von Horn und Schunck verwendeten Euler-Lagrange Gleichungen, funktioniert. Die Optimierung erfolgt hier auch über einen Pyramidenansatz. Hier wird auch eine Implementierung auf der Grafikkarte vorgeschlagen, welche wiederum eine Reduktion der Berechnungsdauer mit sich bringt.

Baker et al. (2007) stellen ein Benchmarking inklusiv Datensatz vor, welches zum Vergleich von dichten Flussverfahren geeignet ist. Es werden Ground-Truth Daten zur Verfügung gestellt und Entwickler können die Ergenisse ihrer Arbeiten direkt mit vorhandenen Verfahren vergleichen.

2.1.4. Szenenfluss

Als Szenenfluss (engl.: *Scene Flow*) bezeichnet man die vollständige, dreidimensionale Information über die Bewegung von 3D-Punkten in einer Szene. D.h. einem Punkt auf der Oberfläche wird ein sechsdimensionaler Vektor zugeordnet, welcher die dreidimensionale Position und den dreidimensionalen Bewegungsvektor des Punktes beinhaltet. Im Unterschied zum optischen Fluss wird die Bewegung vollständig dreidimensional repräsentiert. Der Szenenfluss kann als die 3D-Erweiterung des optischen Flusses gesehen werden oder umgekehrt ist der optische Fluss die Projektion des Szenenflusses auf die Bildebene. Die Literatur bietet weitaus weniger Ansätze zur Szenenflussberechnung als beispielsweise zur Berechnung des optischen Flusses. Man beschäftigt sich erst seit ca. zehn Jahren damit, da experimentelle Untersuchungen aufgrund nicht ausreichender Rechnertechnologien nicht oder nur beschränkt durchgeführt werden konneten. Das ist auch der Grund dafür, dass erst in den letzten Jahren brauchbare Ansätze aufgekommen sind, die nun dank moderner Technologien zu berechnen sind.

Grundsätzlich benötigt man zur Ermittlung des Szenenflusses mindestens vier Bil-

der: jeweils ein Stereobildpaar zum Zeitpunkt (t_0) und eins zum nächsten Zeitpunkt (t_1). Alleine eine Kombination von Stereo-Punkten mit einem optischen Flussfeld führt nicht zu der gleichen Information, da die 3D-Punkte zusammen mit den 2D-Verschiebungsvektoren nur fünfdimensionale Vektoren ergeben und dabei die Bewegungskomponente parallel der optischen Achse des Kamerasystems unbestimmt bleibt.

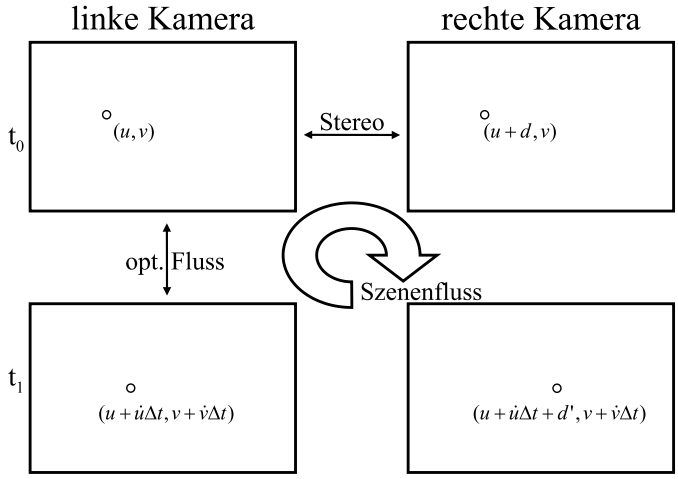


Abbildung 2.9.: Zusammenhänge zwischen den zwei Stereobildpaaren zu zwei aufeinanderfolgenden Zeitschritten.

Um diese Problematik zu umgehen wird von Franke et al. (2005) ein Kalman-Filter dazu benutzt, die fehlende Bewegungskomponente durch zeitliche Filterung zu ermitteln. Ein Nachteil ist hierbei die Vielzahl von Filtern, da jedem 3D-Punkt ein Filter zugeordnet wird. Um dennoch echtzeitfähig zu bleiben, kann nur eine verhältnismäßig geringe Anzahl von Szenenpunkten berechnet werden, was in einer spärlichen Szenenbeschreibung resultiert. Die Filterung bedingt außerdem eine gewisse Einschwingzeit und bringt eine Tiefpassfilterung der Bewegung mit sich. Für schnelle Bewegungsänderungen ist dieser Ansatz ungeeignet.

Der englische Begriff Scene Flow wurde maßgeblich durch die Arbeiten von Vedula et al. (1999) geprägt. In der Veröffentlichung werden Methoden vorgestellt, um entweder mit vollständigem Szenenwissen, mit Wissen über Stereokorrespondenzen oder ohne Vorwissen den Szenenfluss zu berechnen. Dies erfolgt jeweils auf der Analyse des optischen Flusses in den Bildern eines Multi-View Kamerasystems. Dieser Ansatz ist für Stereobilder nur bedingt geeignet.

Ein Variationsansatz wird von Huguet und Devernay (2007) benutzt, um gleichzeitig alle Komponenten des Szenenflusses zu bestimmen. Dabei setzt sich das zu minimierende Funktional $E(u, v, d, d')$ aus Glattheits- und Datenterm, welche durch den Regularisierungsparameter λ verbunden sind, zusammen:

$$E(u, v, d, d') = E_{Data} + \lambda E_{Smooth}. \tag{2.23}$$

Dabei beschreiben u und v die Bildkoordinaten im linken Bild, d die Disparität zum Zeitpunkt t_0 und d' die Disparität zum Zeitpunkt t_1. Der Datenterm E_{Data} besteht aus den vier Fehlerbeziehungen zwischen den Bildern:

$$E_{Data} = \int_{\Omega} (\beta_{fl} E_{fl} + \beta_{fr} E_{fr} + \beta_{st} E_{st} + \beta_s E_s) d\mathbf{x}, \tag{2.24}$$

wobei durch β_{fl}, β_{fr}, β_{st} und β_s Verdeckungen berücksichtigt und aus der Berechnung ausgeschlossen werden. Durch Ω wird der Berechnungsraum und durch $\mathbf{x}$ die Bildkoordinate mit $\mathbf{x} = (x, y, t)$ definiert. Die vier Fehlerterme in E_{Data} werden beschrieben durch:

$$E_{fl}(u, v, d, d') = \Psi(\delta(I_l, \mathbf{x}; I_l, \mathbf{x} + \mathbf{w})), \tag{2.25}$$

$$E_{fr}(u, v, d, d') = \Psi(\delta(I_l, \mathbf{x} + \mathbf{d}; I_l, \mathbf{x} + \mathbf{w} + \mathbf{d'})), \tag{2.26}$$

$$E_{st}(u, v, d, d') = \Psi(\delta(I_l, \mathbf{x} + \mathbf{w}; I_l, \mathbf{x} + \mathbf{w} + \mathbf{d'})), \tag{2.27}$$

$$E_s(u, v, d, d') = \Psi(\delta(I_l, \mathbf{x}; I_l, \mathbf{x} + \mathbf{d})). \tag{2.28}$$

Die Ψ-Funktion ist definiert als $\Psi(s^2) = \sqrt{s^2 + \varepsilon^2}$ (mit $\varepsilon = 0.001$). Die Verwendung der Ψ-Funktion führt zu einer robusten Fehlerfunktion, korrespondierend zur L^1 Norm, die jedoch überall differenzierbar ist.

Das Ähnlichkeitsmaß ist definiert als:

$$\delta(I, \mathbf{x}; I', \mathbf{y}) = |I'(\mathbf{y}) - I(\mathbf{x})|^2 + \gamma |\nabla I'(\mathbf{y}) - \nabla I(\mathbf{x})|^2, \tag{2.29}$$

wobei $\nabla = (\partial_x, \partial_y)^T$.

Der Glattheitsterm ist dabei definiert als:

$$E_{Smooth} = \int_{\Omega} \Psi(|\nabla u|^2 + |\nabla v|^2 + \sigma |\nabla (d' - d)|^2 + \mu |\nabla d|^2) d\mathbf{x}, \tag{2.30}$$

wobei σ und μ zur zusätzlichen Steuerung des Optimierungsverhaltens benutzt werden. Es wird zur Berechnung ein Pyramidenansatz benutzt, um auch große Veränderungen korrekt berechnen zu können. Die Ergebnisse sind sehr aufwendig zu berechnen und benötigt für VGA-Aufnahmen mehrere Minuten Rechenzeit, um den vollständigen Szenenfluss auf einer modernen Rechnerplattform zu berechnen.

Wedel et al. (2008) beschreiben ein Verfahren, welches zur Berechnung des Szenenflusses wiederum die 3D-Rekonstruktion von der Bewegungsschätzung trennt. Es wird also zunächst mittels Stereobildverarbeitung eine spärliche oder dichte Tiefenkarte erzeugt, um dann wiederum auf einem Regularisierungsansatz basierend die dreidimensionale Bewegung zu schätzen. Der Vorteil ist hier die deutlich kürzere Berechnungsdauer.

Pons et al. (2003) benutzten ebenfalls einen Variationsansatz, um Oberflächen im Raum zu generieren und über statistische Ähnlichkeitsmaße die beobachtete Oberfläche bestmöglich zu rekonstruieren. Die Bewegungsinformationen werden anschließend durch den gleichen Ansatz unter Verwendung des nächsten Stereobildpaares extrahiert.

Der Ansatz von Li und Sclaroff (2008) verwendet Wahrscheinlichkeitsverteilungen für den optischen Fluss und die Stereorekonstruktion um daraus verlässliche Szenenflussdaten zu generieren. Unsicherheiten in der Korrespondenzbildung für die Fluss- bzw.

Stereoberechnung werden dadurch in der globalen Szenenflussberechnung berücksichtigt, was zu einem stabileren Ergebnis führt. Außerdem wird ebenfalls ein Pyramiden-Ansatz verwendet. Synthetische und reale Testdaten zeigen die Leistungsfähigkeit des Ansatzes. Die Rechenzeit beträgt auf einer modernen Rechnerplattform ca. 2 Minuten für vier Bilder in QVGA-Auflösung.

Ein spärlicher Ansatz zur Bestimmung des Szenenflusses wird von Schmidt et al. (2007) vorgestellt. Das Verfahren basiert auf der Modellierung von Grauwertkanten in raum-zeitlichen ROIs. Es wird dabei gemäß der PSF (siehe Abschnitt 2.1.1) eine Sigmoid-Funktion an die Grauwertkante in aufeinanderfolgenden Bildern angepasst und mithilfe der Kantenparameter Korrespondenzen etabliert, wodurch 3D-Punkte bestimmt werden. Da es sich um eine raum-zeitliche Kanteninterpretation handelt, kann außerdem der optische Fluss und die Tiefengeschwindigkeit ermittelt werden. Das Verfahren und die im Rahmen dieser Arbeit entwickelten Erweiterungen werden in Abschnitt 4.2.1 genauer erklärt.

2.1.5. Szenensegmentierung

Da in der vorliegenden Arbeit ausschließlich 3D-Daten verarbeitet werden, wird an dieser Stelle nur auf Verfahren zur 3D-Punktewolkensegmentierung eingegangen.

Das Ziel der Segmentierung ist es, die 3D-Punktewolke der vorliegenden Szene in einzelne Objekte zu unterteilen. Dabei wird entweder auf modellfreie Verfahren, i.A. als Cluster-Verfahren bezeichnet (Duda und Hart, 1973) oder auf modell-basierte Verfahren zurückgegriffen, wobei der ICP-Algorithmus eine wichtige Rolle spielt (siehe Kap. 2.1.7). Sehr ähnliche Methoden kommen auch in der Photogrammetrie auf Basis von Laserscanner-Daten zum Einsatz (siehe z.B. (Brenner, 2005; Rottensteiner et al., 2005)).

Zur Segmentierung kommen oft die klassischen Cluster-Ansätze zum Einsatz, wie das agglomerative und das divisive Clustering. Hierbei muss keine feste Clusteranzahl vorgegeben werden. Diese Verfahren benötigen allerdings ein Abbruchkriterium, was je nach Anwendung schwer zu definieren ist. Anders wird beim k-Means Clustering (MacQueen, 1967) eine feste Anzahl von Clustern vorgegeben, was ebenfalls abhängig vom Szenario schwierig sein kann. Weitere interessante Methoden zum Clustering basieren auf der Graphentheorie. Hier wird die Graph-Cut Optimierung (Shi und Malik, 1997) bzw. sehr ähnlich dazu das spektrale Clustering (Shi und Malik, 2000) dazu benutzt, um eine optimale Trennung der Objekte in der Punktewolke zu erreichen. Außerdem zu erwähnen ist das sogenannte Mean-Shift-Clustering, welches ursprünglich zur Farbbildsegmentierung entwickelt wurde (Comaniciu und Meer, 2002). Hierbei muss weder eine Clusteranzahl noch ein Abbruchkriterium definiert werden. Das Verfahren ist jedoch für 3D-Punktewolken nur bedingt geeignet.

Ein Clusterverfahren, welches auch in der vorliegenden Arbeit zum Einsatz kommt, wird von Konrad (2006) vorgestellt. Das Verfahren ist besonders für attributierte Punktewolken geeignet, wenn beispielsweise zusätzlich zu den 3D-Koordinaten auch Bewegungsinformationen für Szenenpunkte verfügbar sind. Es werden zwei Bedingungen an benachbarte Punkte des gleichen Clusters gestellt: zum einen muss der euklidische Ab-

stand im 3D-Raum unter einer definierten Schwelle θ_{3D} liegen und zum anderen muss der Geschwindigkeitsunterschied ebenfalls kleiner einer Schwelle θ_v sein. Sind beide Bedingungen erfüllt, so werden beide Punkte dem gleichen Cluster zugeordnet. Über einen Rekursionsansatz wird anschließend gewährleistet, dass die Clusternummern alle sinnvoll vergeben sind, d.h. Punkte eines Clusters auch abschließend alle die gleiche Clusternummer haben. Die Grundidee des Ansatzes geht auf Bock (1974) zurück. Das Verfahren wurde ursprünglich zum Clustering von Radarpunkten entwickelt. Abb. 2.10 zeigt das Verfahren angewendet auf Radardaten.

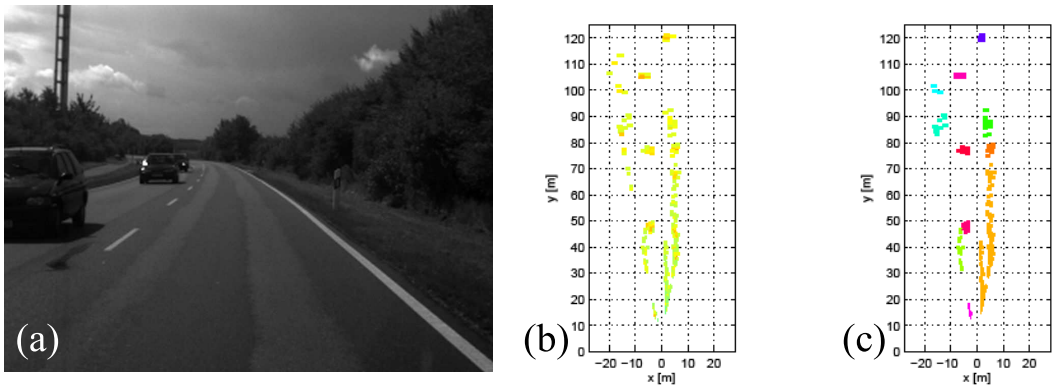


Abbildung 2.10.: Beispiel für die Segmentierung von Radardaten nach Konrad (2006). (a) Kamerabild der Szene. (b) Gefilterte Radardaten (Farben definieren Signalstärke). (c) Segmentierte Objekte (gleiche Farbe = gleiches Cluster).

Der ICP-Algorithmus (Besl und McKay, 1992) ist ursprünglich zur Registrierung von 3D-Formen (Punktewolken, Liniensegmente, Flächen, etc.) entwickelt worden. Durch die Erweiterung von Zhang (1992) wird zusätzlich auch eine Segmentierung erreicht, da während der Poseschätzung auch die Menge der korrespondierenden Datenpunkte verändert wird. Ist der Abstand eines Punktes zum Modell über eine Schwelle D_{max}, so wird der Punkt aus der Punktewolke entfernt. Der Parameter D_{max} wird dabei im statistischen Sinne nach jeder Iteration ermittelt. Schlussendlich liefert der Algorithmus neben der Objektpose auch die segmentierte Objektpunktewolke.

Ein weiterer Ansatz zur Szenensegmentierung wird von Schmidt et al. (2007) vorgestellt. Hier wird ebenfalls auf bewegungsattributierten Punkten gearbeitet und zunächst ein hierarchisches Clustering durchgeführt. Anschließend wird mithilfe eines empirisch bestimmten Schwellwerts der hierarchische Baum partitioniert, was in einer Vielzahl von einzelnen Clustern resultiert. Diese einzelnen Cluster werden entsprechend ihrer mittleren Geschwindigkeit zusammengefasst und durch einen einhüllenden Zylinder repräsentiert. Jeder Zylinder bildet nun eine Objekthypothese, welche anschließend durch ein Partikel-Filter verfolgt wird. Somit erhält man eine Beschreibung der Szene, ohne die Objekte darin genauer zu kennen und kann dreidimensionale Objekte über die Zeit

in der Szene verfolgenden.

2.1.6. 3D-Pose-Estimation

Das Ziel der Pose-Estimation ist es, aus den Informationen eines Sensors die Lage und Orientierung (zusammengefasst die Pose) eines Objekts zu schätzen. Dabei wird meist auf ein mehr oder weniger detailliertes Modell des Objekts zurückgegriffen.

Man unterscheidet dabei verschiedene Kategorien von Objekten: rigide Objekte, artikulierte Objekte und Freiformen (auch nicht-rigide Objekte genannt). Ein rigides Objekt (alle festen Körper) hat im 3D-Raum sechs Freiheitsgrade, wodurch auch sechs Parameter zur vollständigen Definition der Pose benötigt werden, wobei diese sich in drei Translations- und drei Rotationsparameter aufteilen. Artikulierte Objekte (z.B. der menschliche Körper oder Teile davon, Roboter, Züge, Auflieger-LKW) haben zusätzlich noch eine endliche Anzahl von interne Freiheitsgraden, welche zusätzlich als Parameter zu den sechs Parametern der rigiden Objekte hinzukommen. Nicht-rigide Objekte (z.B. Kabel, Schläuche und alle nicht festen Körper) haben eine unendliche Anzahl von Freiheitsgraden. Daher wird meist eine Modellfunktion mit einer endlichen Anzahl von Freiheitsgraden bestimmt, um die Pose approximieren zu können.

Haralick et al. (1989) beschreiben drei verschiedene Klassen des Pose-Estimation Problems: die 2D-2D Pose-Estimation ermittelt die Pose eines 2D-Objekts aus einem Bild, wohingegen die 3D-3D Pose-Estimation die Pose eines 3D-Objekts aus 3D Punkten bzw. 3D-Daten schätzt. Eine weitere Klasse bezeichnet die 2D-3D Pose-Estimation: hier wird versucht alle sechs Poseparameter eines dreidimensionalen Objekts aus einem Bild zu schätzen. Diese Klasse wird in der Photogrammetrie auch als externes Orientierungsproblem bezeichnet (Kraus et al., 1997). Lösungsansätze finden sich für diese Klasse zahlreich in der Literatur (z.B.: (Lowe, 1991; Phong et al., 1996; Rosenhahn et al., 2003)).

Das Hauptproblem bei der 2D-3D Pose-Estimation ist die Schätzung der Entfernung des Objekts zur Kamera. Hier treten vergleichsweise hohe Ungenauigkeiten auf, da der einzige Anhaltspunkt zur Schätzung der Entfernung die bekannte Größe des Objekts ist. Selbst bei exakt bekannter Größeninformation stellt sich das Problem, dass für kleine Entfernungsänderungen die Objektgröße im Bild quasi gleich bleibt. Hingegen werden durch ein Stereo-Kamerasystem wesentlich bessere Tiefeninformationen gewonnen, was zu einer erhöhten Genauigkeit der Entfernungsschätzung führt (Faugeras, 1993). Da es im Kontext der vorliegenden Arbeit sehr wichtig ist, möglichst genaue Pose-Parameter zu ermitteln, werden lediglich 3D-3D Verfahren verwendet.

Eine Gruppe von Pose-Estimation Verfahren sind ansichtsbasierte oder *appearance-based* Methoden. Dabei wird direkt das Abbild des Objekts mit Bildern des Objekts in bekannter Pose verglichen. Hierunter zählt auch das monokulare Template-Matching (von Bank et al., 2003) und das multiokulare Template-Matching, auf welches Krüger (2007) näher eingeht.

Eine weitere Gruppe von Verfahren wird als *Model-based Segmentation* bezeichnet. Hierbei werden die Parameter einer geeigneten geometrischen Repräsentation adaptiert, beispielsweise über eine Konturensuche. Zu diesen Verfahren zählen die *Active Contours*

(Kass et al., 1988) und der CCD-Algorithmus (Hanek, 2004). Diese Verfahren eignen sich insbesondere für nicht-rigide Objekte mit einer nicht beschränkten Anzahl von internen Freiheitsgraden.

Die letzte Gruppe von Verfahren ordnet Merkmale (engl.: Features) dem Modell zu und ermittelt die Pose des Modells daraus. Hierunter fällt auch der Iterative Closest Point (ICP) Algorithmus, der beispielsweise von Besl und McKay (1992) beschrieben ist und auf den in Kap. 2.1.7 näher eingegangen wird.

Eine detaillierte Übersicht über multiokulare Pose-Estimation Verfahren gibt Krüger (2007) wieder. Auf Verfahren speziell für artikulierte Objekte wird in Kap. 2.2 genauer eingegangen.

2.1.7. Punktbasierte 3D-Modellanpassung

Ein Spezialfall der 3D-3D Pose-Estimation ist die punktbasierte Modellanpassung. Dabei wird vorausgesetzt, dass die Eingangsdaten als Punkte im Raum verfügbar sind. Diese Repräsentation findet man in erster Linie bei Daten aus Laserscannern, aber auch als Resultat einer Stereobildverarbeitung. Nachdem das Objekt aus der initialen Punktewolke extrahiert wurde (siehe Abschnitt 2.1.5), wird ein mehr oder weniger detailliertes Modell des Objekts benutzt, um dessen Pose zu ermitteln. Dabei kann das Modell unterschiedlich beschrieben werden, beispielsweise ebenfalls durch eine Punktewolke oder durch eine Menge von Linien, Kurven oder Oberflächen. Zusammengefasst lässt sich das Problem folgendermaßen ausdrücken:

Gesucht wird eine Transformation T, die die Pose des Modells M in der Art verändert, sodass die Objektpunktewolke O mit dem Modell bestmöglich übereinstimmt. Diese Übereinstimmung wird erreicht, wenn die Distanz *dist* zwischen den Objektpunkten und dem Modell minimal ist:

$$\min_{T} \; dist[T(M), O]. \tag{2.31}$$

Die im Folgenden beschriebenen Algorithmen haben ihren Ursprung in der Punktewolkenregistrierung, d.h. in der Ermittlung der Transformation zwischen zwei Punktewolken. Dabei wurde zunächst angenommen, dass zwei Punktewolken des gleichen Objekts vorhanden und die richtigen Korrespondenzen zwischen den Punkten bekannt sind. Unter diesen Voraussetzungen gibt es zwei Möglichkeiten, die eine geschlossene Least-Squares-Lösung bieten: zum einen das auf Quaternionen basierende Verfahren nach Horn (1987b) und zum anderen der Ansatz nach Arun et al. (1987), welchem eine Singulärwertzerlegung (engl.: Singular Value Decomposition (SVD)) zugrunde liegt.

Für den Fall, dass die Korrespondenzen zwischen den Punktewolken nicht initial bekannt sind, wurde Anfang der 90er Jahre der *Iterative Closest Point* (ICP) Algorithmus entwickelt, u.a. von Besl und McKay (1992) und Zhang (1992). Außerdem wurden hier auch erstmals andere Repräsentationen für das Modellobjekt eingeführt und man beschäftigte sich ab diesem Zeitpunkt auch mit der Behandlung von Messfehlern.

Die verschiedenen Varianten des ICP-Algorithmus (vgl. (Rusinkiewicz und Levoy, 2001)), die darauf folgten, lassen sich durch die folgenden Verarbeitungsschritte charakterisieren:

- **Selektion** der Objektpunkte
- **Korrespondenzbildung** zwischen Modell und Punktewolke
- **Gewichtung** der Punktpaare
- **Rückweisung** von Punktpaaren
- Generierung einer **Fehlermetrik** für die verbleibenden Punktpaare
- **Minimierung** der Fehlermetrik

Selektion: In der Literatur werden verschiedene Ansätze zur Punktselektion erwähnt. Die wichtigsten dabei sind:

- Benutztung aller Punkte
- Uniformes Subsampling aller Punkte
- Zufällige Auswahl
- Betrachtung weiterer Merkmale (z.B. Intensitätsgradient im Bild)

In der Regel wird die Art der Selektion durch die Anzahl von verfügbaren Punkten bestimmt. Sollten sehr viele Punkte verfügbar sein (z.B. durch die Verwendung eines Laserscanners), machen Subsampling oder Zufallsauswahl die Registrierung schneller ohne an Qualität zu verlieren. Dabei ist allerdings auch stets zu beachten, dass mindestens genau so viele Punkte vorhanden sein müssen, wie es zu bestimmende Pose-Parameter gibt.

Korrespondenzbildung: Auch hier gibt die Literatur bereits verschiedenste Ansätze vor:

- Finde den nächsten Punkt des Modells (wenn das Modell auch durch Punkte o.ä. beschrieben ist)
- *Normal shooting*: Die Verbindungsgerade startet im Datenpunkt und endet als Normale auf der Modelloberfläche
- *Reverse Calibration*: Der Sehstrahl des Sensors verbindet Datenpunkt und Modell
- Eine der drei beschriebenen Methoden, allerdings unter Berücksichtigung weiterer Merkmale (Intensität o.ä.)

Wird der nächste Punkt auf dem Modell gesucht, so empfiehlt sich die Verwendung eines *k-d Trees* (Zhang, 1992). Auch das sogenannte *Closest-Point Caching* (Simon, 1996) führt hier zu einer signifikanten Reduktion der Berechnungskomplexität.

Wird das Modell hingegen kontinuierlich beschrieben, kommen meist die Varianten Normal shooting oder Reverse Calibration, je nach Sensoreigenschaften zum Einsatz.

Gewichtung: Um die Optimierung zu stabilisieren, können die Korrespondenzpaare zusätzlich noch gewichtet werde:

- Konstantes Gewicht für alle Punktpaare
- Gewichtung umgekehrt proportional zur Distanz, um den Einfluss von Ausreißern zu verringern
- Gewichtung anhand eines weiteren Merkmals (z.B. Intensität)
- Gewichtung anhand der Sensorcharakteristik und der Ungenauigkeit der Messdaten

Auch hier ist das Messverfahren ausschlaggebend für die Art der Gewichtung. Zur Ausreißerbehandlung kann außer der umgekehrt proportionalen Gewichtung zur Distanz auch auf Verfahren aus der robusten Statistik zurückgegriffen werden, beispielsweise durch eine Gewichtung mittels M-Schätzer (Rey, 1983).

Rückweisung: Um durch ungültige Korrespondenzen nicht die Optimierung zu stören, können an dieser Stelle Punktpaare auch zurückgewiesen werden:

- Rückweisung der schlechtesten $n\%$ Paare (abhängig von der Distanz)
- Rückweisung von Paaren mit einer Distanz größer einer definierten Schwelle
- Rückweisung von Paaren mit einer Distanz größer als n-mal die Standardabweichung der Distanzverteilung aller Punkpaare
- Rückweisung an Objektkanten
- Rückweisung über die Eigenschaften der benachbarten Punktpaare (hierfür wird eine Stetigkeit der Oberflächen angenommen)
- Keine Rückweisungen

Abhängig vom Messverfahren ist eine Rückweisung sehr sinnvoll oder sogar unabdingbar, da sonst keine Konvergenz bei der Optimierung erfolgt. Da bei einem Messverfahren auch immer sogenannte Ausreißer entstehen, z.B. durch Meßfehler verursacht, ist deren Behandlung wichtig, um dennoch ein genaues Registrierungsergebnis zu erhalten.

Fehlermetrik und Optimierung: Für Probleme mit einer festen Punktezuordnung sind seit längerem geschlossene Lösungsverfahren bekannt, entweder auf Basis einer Singulärwertzerlegung (SVD) oder mittels Quaternionen. Dabei unterscheiden sich die Ergebnisse der Verfahren nur geringfügig (Eggert et al., 1997).

Ändert sich die Punktezuordnung bzw. handelt es sich um ein Oberflächenmodell, wird in der Literatur auf nichtlineare Optimierungsalgorithmen verwiesen (z.B. Levenberg-Marquardt). Auch eine stochastische Suche mittels *Simulated annealing* wird vorgeschlagen (Blais und Levine, 1995).

Bei einer iterativen Lösung können nach jeder Iteration neue Korrespondenzen gebildet, Gewichtungen berechnet und Rückweisungen durchgeführt werden. Dadurch wird eine zusätzliche Stabilität in der Optimierung erreicht, die bei geschlossenen Lösungen nicht zu erreichen ist.

Die Vorteile bei einer Modellanpassung mittels ICP-Algorithmus sind zum einen die hohen Robustheit des Ansatzes und zum anderen in der gleichzeitigen Effizienz der Berechnung. Der Nachteil liegt in der Vorverarbeitung, da nicht direkt auf Kamerabildern gearbeitet werden kann, sondern zunächst eine Stereoanalyse durchgeführt werden muss.

2.1.8. Bewegungsanalyse

Die Schätzung von Objektbewegungen aus Bildsequenzen ist eine oft thematisierte Fragestellung in der Literatur. Grundsätzlich muss die Applikation genauer betrachtet werden, um die richtige Methode zur Bewegungsschätzung auswählen zu können. Im Folgenden werden die gängigsten Kategorien erläutert.

Instantane Bewegungsschätzung

Bei der instantanen Bewegungsschätzung werden gleichzeitig Bilder aus mehreren Zeitschritten verwendet, um die Bewegung eines Objekts zu berechnen, wobei keine zeitliche Filterung verwendet wird und somit kein Einschwingverhalten vorhanden ist. Nachdem modellbasiert die Pose im aktuellen Zeitschritt bestimmt wurde, wird dasselbe Objektmodell dazu verwendet, die Bewegung zum nächsten Zeitschritt zu ermitteln. Dabei wird kein weiteres Wissen, beispielsweise über die Objektbewegung aus vorherigen Schätzungen, verwendet. Gleichzeitiger Vor- und Nachteil hierbei ist die fehlende Filterung. Das Ergebnis der Bewegungsschätzung steht bei diesen Verfahren direkt zur Verfügung, ohne dass eine Einschwingzeit benötigt wird. Wird eine Bewegung von einem zum anderen Zeitschritt falsch oder gar nicht ermittelt, so hat auch dies direkten Einfluss auf darauf aufbauende Algorithmen. Nachteile können bei schnellen Bewegungen oder Bewegungsänderungen entstehen, da diese Bewegungsschätzung nur einen geringen Konvergenzradius besitzt. Wird die Bewegungsschätzung also ungenau initialisiert, kann die korrekte Bewegung womöglich nicht ermittelt werden. Für diese Art der Bewegungsschätzung wird kein Bewegungsmodell benötigt, weshalb sie sehr vielseitig einsetzbar ist. Ein Beispiel für eine solche instantane Bewegungsschätzung ist der *ShapeFlow-Algorithmus*, welcher von Hahn et al. (2008b) vorgestellt wird.

Detektionsbasierte Bewegungsschätzung

Die detektionsbasierte Bewegungsschätzung (engl.: *tracking by detection*) bedingt einen leistungsfähigen Detektor, der in jedem Zeitschritt aufs Neue dazu benutzt wird, das zu verfolgende Objekt zu erkennen. Anschließend werden detektierte Objekte bereits vorhandenen Bewegungstrajektorien aus den verstrichenen Zeitschritten zugeordnet. Dazu dient im einfachsten Fall ein Distanzmaß zwischen detektierter Objektposition und der

Trajektorie oder einer Ähnlichkeitsfunktion zwischen den Objekten. Kann einem detektierten Objekt keine Trajektorie zugeordnet werden, so wird eine neue Trajektorie begonnen. Trajektorien, deren Objekt für mehrere Zeitschritte nicht sichtbar war, werden gelöscht.

Der Vorteil dieser Verfahren liegt in der, ähnlich zur instantanen Bewegungsschätzung, schnellen Reaktionszeit ohne Einschwingverhalten. Nachteil ist jedoch, dass ein robuster Detektor Voraussetzung für diese Art der Bewegungsschätzung ist. Außerdem können Fehldetektionen direkt zu Ausreißern in der Bewegungsschätzung führen. Ein Beispiel für ein solches Verfahren ist das Gesichts-Detektions-System, welches von Yan und Forsyth (2004) vorgestellt wird.

Flussbasierte Bewegungsschätzung

Alternativ zur Bewegungsschätzung mittels Detektor wird oft eine flussbasierte Bewegungsschätzung implementiert. Hierbei werden objektunabhängige Korrespondenzen in aufeinanderfolgenden Zeitschritten etabliert und dadurch der optische Fluss berechnet (siehe Abschnitt 2.1.3). Durch die Analyse dieses Flussfeldes ist es möglich, die Objektbewegung zu schätzen. Die Schwierigkeit dabei liegt in der dreidimensionalen Bewegungsschätzung aus zweidimensionalen Flussdaten. Dies kann zum einen zu einer fehlenden Bewegungskomponente parallel der optischen Achse des Kamerasystems führen. Zum anderen kann es aber auch zu Fehlern in der Bewegungsschätzung kommen, da Mehrdeutigkeiten auftreten.

In verschiedenen Veröffentlichungen werden die Probleme betrachtet, die sich bei der Analyse von Fluss-Daten ergeben:

- **Fermüller und Aloimonos (1997) und Fermüller und Aloimonos (1995):** Analyse von 2D-Verschiebungsvektoren für Structure-from-Motion und Egomotion, Betrachtung von Bewegungen rigider Objekte, Constraints für Verschiebungsvektorfelder
- **Brodsky et al. (1998):** Ausschließliche Analyse der Richtung von 2D-Verschiebungsvektoren, Mehrdeutigkeiten in der Schätzung der 3D-Bewegung eines rigiden Objekts werden untersucht
- **Horn (1987a):** Beschreibung der „Gefährlichen Flächen"-Problematik: Mehrdeutigkeiten bei der Verwendung des optischen Flusses zur Objektbewegungsschätzung
- **Fermüller et al. (2001):** Analyse des Einflusses von Pixelrauschen auf die Berechnung des optischen Flusses und den dadurch resultierenden Fehlern
- **Nagel (1989):** *Oriented Smoothness Constraint*: Wo ist ein Smoothness Constraint zur Fluss-Berechnung sinnvoll?

Zusammengefasst sieht man, dass in der Litertur verschiedene Verfahren zur Bewegungsschätzung existieren, die allerdings alle 2D-Bewegungsvektoren bedingen. Diese

2D-Vektoren sind nur dort zu etablieren, wo eine Ecke im Bild vorhanden ist. Sind lediglich Kanten des Objekts zu sehen, können diese Methoden nicht eingesetzt werden, da das Apertur-Problem hier einen zu großen Einfluss hat.

Um aus den 2D-Flussdaten 3D-Objektbewegungen zu schätzen, können auch sogenannte Flow-Templates benutzt werden. Ju et al. (1996) zeigen solche Templates für verschiedene 3D-Bewegungen.

Ein anderer Ansatz, um 2D-Bewegungsdaten zu einer 2D-Objektbewegung zusammenzufassen, ist das sogenannte Constraint-Line-Clustering. Hier werden nicht nur 2D-Bewegungen, sondern auch die Objektkontur dazu benutzt, eine vollständige Bewegung zu beschrieben. Nachteil hierbei ist, dass lediglich Translationen ermittelt werden können. Vorteil ist jedoch, dass das Aperturproblem hier berücksichtigt und umgangen wird. Es ist also auch für Flussvektoren an Kanten einsetzbar.

Auch auf Basis von vollständigen Szenenflussdaten können 3D-Objektbewegungen extrahiert werden. Problem ist hierbei die aufwendige Szenenflussberechnung.

Bewegungsschätzung mittels probabilistischer Filterung

Werden über eine längere Sequenz Messungen eines dynamischen Prozesses aufgezeichnet, dessen Dynamik weitestgehend bekannt ist, so ist es sinnvoll eine probabilistische Filterung zu verwenden. Zur vereinfachten Erklärung wird das Beispiel der Fahrzeugverfolgung mittels Radar-Sensor herangezogen. Die Bewegungsdynamik des Fahrzeugs lässt sich gut beschreiben, da durch die Trägheit des Fahrzeugs und durch das Bewegungsmodell (z.B.: Ackermann-Modell) sich eine Bewegung sinnvoll prädizieren lässt. Die Messungen aus dem Radar-Sensor sind mit einer Unsicherheit behaftet, die in erster Linie durch das Messrauschen entsteht. Außerdem wird auch der Vorhersage mittels Bewegungsmodell eine gewisse Unsicherheit zugeordnet, durch die die Güte der Prädiktion beschrieben wird. Benutzt man nun ein Kalman-Filter, das am weitesten verbreitete probabilistische Filter, wird der Zustand aus dem letzten Zeitschritt zusammen mit dem Vorhersagemodell und der aktuellen Messung dazu benutzt, auf den aktuellen Zustand des Systems zu schließen. Das Ergebnis ist ein Zustand, der sowohl von der aktuellen Messung als auch von der Prädiktion mittels Bewegungsmodell abhängt.

Durch die Verwendung eines Kalman-Filters kann nicht nur eine zeitliche Filterung unter Berücksichtigung eines Prädiktionsmodells bereitgestellt werden, es ist außerdem möglich, Zustände des Systems zu schätzen, die nicht direkt durch Messungen verfügbar sind. Am Beispiel der Fahrzeugverfolgung mittels Radar-Sensor wird nicht direkt die Position des Fahrzeugs gemessen, sondern die vom Fahrzeug reflektierten elektromagnetischen Wellen. Ist der Zusammenhang zwischen Messung und Zustand linear, kann das ursprüngliche Kalman-Filter verwendet werden. Ist dieser Zusammenhang nichtlinear, muss eine Erweiterung des Kalman-Filters benutzt werden. Das Extended Kalman Filter (EKF) (Thrun et al., 2005) verwendet den nichtlinearen Zusammenhang, welcher intern allerdings durch die Verwendung der Ableitungen wieder linearisiert wird. Eine andere Variante für nichtlineare Zusammenhänge bildet das Unscented Kalman Filter (UKF) (Thrun et al., 2005). Hier werden die nichtlinearen Zusammenhänge direkt benutzt, indem man eine kleine Menge von Zuständen durch die nichtlineare Vorhersage

propagiert, um später über die Einzelergebnisse auf Mittelwert und Varianz des Zusammenhangs zu schließen.

Das Kalman-Filter ist ein unimodales Filter, da es jeweils nur einen Zustand und eine Prädiktion verarbeiten kann. Eine multimodale Erweiterung stellt das Partikel-Filter dar. Hier können mehrere Messungen gemeinsam verarbeitet werden und ein multimodaler Zustand wiedergegeben werden.

Der größte Nachteil bei der probabilistischen Filterung liegt in den Voraussetzungen. Zum einen muss sowohl für die aktuelle Messung, als auch für das Vorhersagemodell eine Unsicherheit bekannt sein. Außerdem muss ein Vorhersagemodell für den beobachteten Prozess gefunden werden.

Eine ausführliche Beschreibung verschiedener Filtermethoden liefert Thrun et al. (2005).

2.2. Erkennung und Verfolgung von Personen in Bildsequenzen

Schon sehr früh versuchten Menschen die Körperhaltung und die Bewegung von Tieren und von anderen Menschen zu analysieren. Bereits Aristoteles (384-322 v. Chr.) befasste sich mit diesem Thema. Später wurde von Leonardo da Vinci (1452-1519 n. Chr.) große Fortschritte bei der exakten Modellierung der menschlichen Anatomie erreicht. In seinen Arbeiten wurden auch die Grundlagen für aktuelle Modelle (z.B. *Kinematic Chains*) gelegt. Daraufhin befassten sich weitere Maler der Renaissance mit dem Thema, um ihre Gemälde noch realistischer gestalten zu können. Auch der Wissenschaftler Galileo Galilei (1564–1642) studierte im Barock-Zeitalter den Bewegungsgapparat von Tieren aus mechanischen Gesichtspunkten. Im Zeitalter der Fotografie wurden besonders zwei Namen im Zusammenhang mit der Erforschung von menschlicher Bewegung bekannt: der Franzose Étienne-Jules Marey (1830-1904) und der US-Amerikaner britischer Abstammung Eadweard Muybridge (ebenfalls 1830-1904). Beide fotografierten mit mehreren Kameras oder mehreren Belichtungen des gleichen Films die Bewegung von Menschen und Tieren, um diese anschließend analysieren zu können. Im Anschluß wurden in der Biomechanik noch weitere Arbeiten in diesem Bereich veröffentlicht. Ein wichtiger Zwischenschritt war dann das sogenannte *Motion Tracking* mittels reflektierender Marken. Damit wurden Bewegungsabläufe aufgezeichnet, um anschließend mittels Computer Grafik synthetische Bildsequenzen realistischer Bewegungsabläufe generieren zu können. Einsatz fanden diese Verfahren häufig in der Filmindustrie. Schlussendlich wurde dann der Forschungsbereich der Computer Vision in dem Feld aktiv, um auch ohne diese störenden Marker Bewegungsabläufe aus Bildsequenzen extrahieren zu können. Seit ca. 20 Jahren beschäftigen sich Bildverarbeiter auf der ganzen Welt mit diesem komplexen und anspruchsvollen Thema mit dem Ergebnis, dass es immer noch sehr schwierig ist, die Bewegungsabläufe des Menschen exakt zu verfolgen und es daher noch keine allgemeingültige Methode dafür gibt. Einen historischen Überblick über die Entwicklungen auf diesem Gebiet wird von Rosenhahn et al. (2007b) wiedergegeben.

In der Literatur finden sich Zusammenfassungen über die Arbeiten im Bereich Human-Pose-Estimation und Bewegungsanalyse (Sappa et al., 2005; Aggarwal und Cai, 1999;

Liang Wang, 2003; Moeslund et al., 2006; Gavrila, 1999). Auf einzelne Arbeiten wird im Folgenden, geordnet nach dem verwendeten Kamerasystem, eingegangen.

2.2.1. Monokulare Systeme

Eine ausgiebige Zusammenfassung über die Analyse von menschlichen Bewegungen in monokularen Bildsequenzen wird von Sminchisescu (2006) wiedergegeben. Hier werden zunächst einige der Probleme dargestellt, welche bei der Pose-Estimation bzw. bei der Bewegungsanalyse von Personen aus einer monokularen Bildsequenzen auftreten, z.B. Mehrdeutigkeiten, Tiefenschätzung, Bekleidung und Verdeckungen. Es werden bekannte Lösungsansätze vorgestellt, Schwierigkeiten der Systeme herausgestellt und Herausforderungen und offene Probleme als Ideen für zukünftige Systeme dargestellt.

Mehrere Arbeiten sind in der Literatur zu finden, die auf der Analyse von monokularen Farbbildsequenzen basieren (Beispiele: (Schmidt et al., 2006), (Siddiqui und Medioni, 2006), (Noriega und Bernier, 2007)). Von Schmidt et al. (2006) wird ein 3D-Oberkörper-Modell, bestehend aus mehreren Zylindern, ins Bild projiziert und mittels Partikel-Filter verfolgt. Dabei werden Farb- und Kantenmerkmale im Bild extrahiert und zur Modellanpassung benutzt. Der mittlere, euklidische Fehler, der bei der Verfolgung über eine Sequenz entsteht, liegt bei ca. $7 - 12$ cm.

Zusätzlich zu einer Farbkamera wird von Kleinehagenbrock et al. (2002) ein Laserscanner dazu benutzt, Personen zu erkennen, wobei beide Sensoren an einem mobilen Roboter montiert sind. Der Laserscanner scannt seine Umgebung in einer Höhe von ca. 30 cm über dem Boden, die Kamera ist auf einer Höhe von ca. 140 cm montiert. Aus dem Laserscan lassen sich Beinpaare extrahieren, wodurch ein erstes Merkmal für die Existenz einer Person feststeht. Anschließend wird in dem Farbbild der Kamera eine Hautfarbendetektion durchgeführt, um Gesicht und evtl. auch die Hände bzw. Arme zu erkennen. Passen alle Merkmale zusammen, wurde eine Person erfolgreich erkannt und kann anschließend in der Bildsequenz verfolgt werden.

Sowohl Moeslund et al. (2005) als auch Bullock und Zelek (2005) stellen Verfahren vor, die die menschliche Schulter-Arm-Hand-Sektion im Bild verfolgen. Von Bullock und Zelek (2005) wird ein Framework vorgestellt, welches die Detektion der Schulter-Arm Partie und die Verfolgung beinhaltet. Zur Initialisierung wird eine definierte Bewegung detektiert und dadurch die Parameter des Modells definiert, wobei zeitlich unveränderliche Attribute, wie Farbverteilung auf dem Arm und Größe, Aufpunkt und Bewegungsradius, und zeitlich veränderliche Parameter, wie die Pose des Arms, gesetzt werden. Anschließend wird mithilfe eines Partikelfilters die Pose über die Zeit verfolgt. Die Likelihood-Funktion für das Filter basiert dabei auf dem Vergleich des Farbbildinhaltes mit der gespeicherten Farbverteilung des Abbilds des Arms.

Am MIT wurde das sogenannte *Pfinder*-System Mitte der 90er Jahre entwickelt (Wren et al., 1997). Das System basiert auf der Analyse von monokularen Farbbildsequenzen. Zunächst wird ein Modell der statischen Szene berechnet, um Veränderungen zu detektieren. Personen werden darin mittels Blobanalyse detektiert und durch ein zweidimensionales Modell repräsentiert. Die Genauigkeit der Pose-Estimation bzw. der Bewegungsanalyse der Personen wird in Pixeln im Bild angegeben und beträgt im Mittel $1 - 2$

Pixel. Eine metrische Genauigkeit ist nicht angegeben.

Ramanan et al. (2007) zeigen ein selbstlernendes System zum Tracking von Personen in monokularen Bildsequenzen. Dabei werden durch eine Farb- und Kantenanalyse Regionen im Bild gefunden, die als Körperteilhypothesen gespeichert werden. Außerdem werden signifikante Key-Poses in der Bildsequenz gesucht, die wiederum zur Generierung eines neuen Modells verwendet werden. Die einzelnen Körperregionen werden aus den Bildern herausgefiltert und als Repräsentation abgelegt, um sie in den folgenden Frames wieder zu finden. Dabei wird eine detektionsbasierte Bewegungsschätzung benutzt, die in jedem Frame versucht, die abgelegten Repräsentationen wiederzufinden. Hierzu wird eine Mischung aus Template-Matching und Klassifikator benutzt. Das Verfahren ist auch in der Lage mehrere Personen über die Zeit zu verfolgen.

Wertung: Die intensiven Arbeiten an den monokularen Systemen, gerade der letzten Jahren, zeigen eine enorme Robustheit in Bezug auf die Erkennung und Verfolgung von Personen in der beobachteten Szene. Je nach Abbildungsgröße lassen sich auch die einzelnen Körperteile detektieren und Bewegungen grob bestimmen. Für das in dieser Arbeit untersuchte Szenario sind die Möglichkeiten eines monokularen Systems jedoch nicht ausreichend. Die metrische Genauigkeit ist zu gering, da insbesondere die Tiefengenauigkeit unzureichend ist. Für die Sicherheitsapplikation im Produktionsbereich ist dies nicht akzeptabel, wenn auch die hohe Robustheit der Ansätze wünschenswert ist.

2.2.2. Multi-View

Unter Mulit-View versteht man in der Literatur meist die Aufnahme von Bildern durch mehrere Kameras (meist 4, 8, 16 oder mehr) mit einer sehr großen Basisbreite. Die Kameras sind um das Objekt positioniert, sodass überlappend möglichst alle Teile des Objekts gesehen werden. Im Vergleich zur Stereobildverarbeitung wird hier nicht eine Ansicht des Objekts als Oberfläche rekonstruiert, sondern die gesamte Oberfläche als dreidimensionaler Körper.

Von Sigal et al. (2006) wird ein Datensatz mit rund 50.000 Bildern und dazugehörigen Ground-Truth Pose-Daten bereitgestellt, um sowohl Pose-Estimation Algorithmen als auch Handlungserkennungssysteme zu evaluieren. Die Sequenzen dieses sogenannten HumanEva-Datasets zeigen dabei unterschiedliche Personen, die definierte Handlungen (Gehen, Joggen, Kommunikationsgesten, Werfen und Fangen, Boxen oder eine Bewegungsabfolge) durchführen. Ziel der Bereitstellung dieses Datensatzes ist es, eine einheitliche Datenbasis zum Vergleich von Pose-Estimation und Trackingverfahren zu haben und daneben auch Handlungserkennungsalgorithmen evaluieren zu können.

Von der gleichen Arbeitsgruppe wurden auch eigene Arbeiten zum Thema Erkennung von Verfolgung von Personen in Bildsequenzen veröffentlicht. Von Bhatia et al. (2004) werden die Trainingsdaten aus dem beschriebenen HumanEva-Projekt dazu benutzt, einen Klassifikator zu erstellen, der selbstständig Teile des menschlichen Körpers detektiert. In der Arbeit wurden so Kopf, Wade, Ober- und Unterarm detektiert.

Auch in den zahlreichen Arbeiten von Rosenhahn (Rosenhahn et al., 2005, 2007a, 2008) werden Multi-View Aufnahmen zur Pose-Estimation der oberen Körperpartie bzw. des

ganzen Körpers benutzt. Dabei wird zunächst in allen verfügbaren Bildern die Silhouette der Person auf Level-Sets basierend extrahiert. Anschließend wird eine Pose-Estimation unter Verwendung eines Modells durchgeführt, welches 21 Freiheitsgrade besitzt. Dabei wird als Initialpose die Pose aus dem letzten Zeitschritt verwendet. Letztlich wird die Pose nochmals verfeinert indem Korrespondenzpunkte zwischen Modell und Bildinhalten gebildet werden und die Pose mittels ICP-Algorithmus optimiert wird. Das vorgestellte System wurde mit einem markerbasierten System verglichen. Die Ergebnisse zeigen eine mittlere Abweichung von ca. 2° für den Gelenkwinkel eines Ellbogens.

Rosenhahn et al. (2008) beziehen Constraints in die Optimierung mit ein, die sich aus der Interaktion der Person mit einem Sportgerät ergeben. Beispiele hierfür sind ein Snowboard, wobei die Annahme fest verbundener Füße gemacht werden kann, oder ein Fahrrad, wo man davon ausgeht, dass die Füße zum einen stets mit dem Fahrrad verbunden sind, und zum anderen nur kreisende Bewegungen ausführen können. Diese Annahmen werden in die Modellierung mit einbezogen Constraints führen zu einer Stabilisierung der Pose-Estimation und Bewegungsschätzung.

Weitere Arbeiten von Rosenhahn et al. (2007a) beinhalten auch die Modellierung von Kleidung an der beobachteten Person. Dabei wird die Kleidung in die Poseschätzung mit einbezogen und bei dem Silhouettenansatz berücksichtig. Außerdem können äußere Kräfte (z.B. Wind oder Bewegung der Person) die Kleidung beeinflussen und so zu einer sehr realistischen Gestaltung der angezogenen Person führen. Diese zusätzlichen Beschreibungen führen indes auch zu einem Mehr an Berechnungsaufwand, wodurch sich diese Ansätze nochmals von der Echtzeitfähigkeit entfernen.

Eine weitere Arbeitsgruppe, die sich intensiv mit der Pose-Estimation von menschlichen Körpern beschäftigt hat, ist die Gruppe um Lars Mündermann an der Stanford University (Mündermann et al., 2006, 2007). Zunächst wird hier eine visuelle Hülle aus den Multi-View Aufnahmen generiert. Danach kommt der sogenannte artikulierte ICP-Algorithmus zum Einsatz. Er bildet zunächst Korrespondenzen zwischen dem Modell, welches aus Oberflächenpunkten, Gelenken sowie rigiden Strukturen dazwischen besteht, und der visuellen Hülle. Anschließend wird eine Transformation berechnet, um die Punkteabstände zu minimieren, welche gleichzeitig die Bewegung der einzelnen, rigiden Körperteile beschreiben. Dabei wurden Kamerakonfigurationen mit 4, 8 16, 32 und 64 Kameras evaluiert. Die mittlere Genauigkeit der Pose-Estimation für den ganzen Körper liegt bei ca. 1-3 cm, wobei die Genauigkeit mit der Anzahl der Kameras steigt.

Basierend auf Shape-from-Silhouette, was sehr ähnlich zu dem Visual Hull Ansatz (Mündermann et al., 2007) ist, werden von Cheung et al. (2000) Personen zunächst aus den Multi-View Aufnahmen extrahiert. Ist die Person anschließend durch sogenannte Voxel, d.h. durch Volumenelemente, repräsentiert, werden Ellipsen benutzt, um die verschiedenen Körperteile zu repräsentieren und deren Pose zu ermitteln. Die Aufnahmen werden von fünf Kameras bereitgestellt, wobei eine Angabe der metrischen Genauigkeit fehlt.

Sundaresan und Chellappa (2006) verwenden acht Kameras, um menschliche Bewegungen in Bildsequenzen zu verfolgen. Dabei werden sowohl Bewegungsmerkmale als auch Merkmale in den Einzelbildern dazu benutzt, die Bewegung bestmöglich zu schätzen.

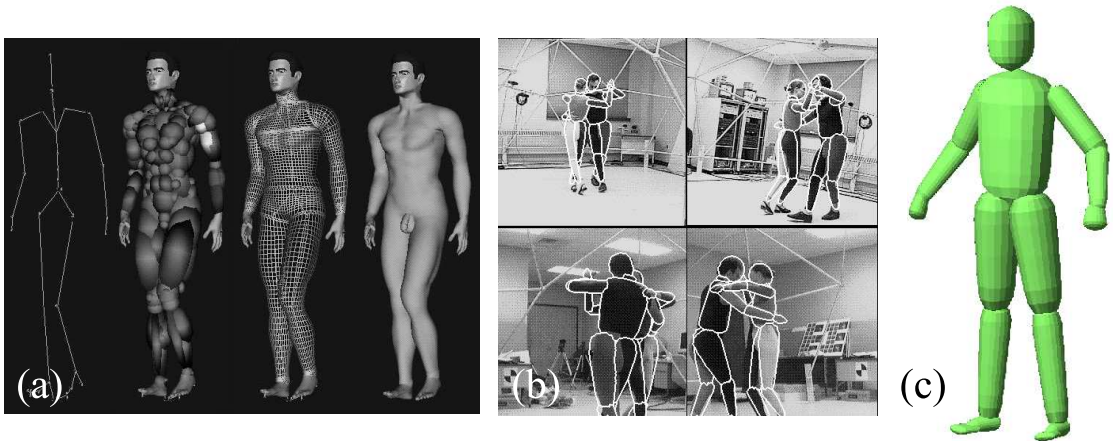


Abbildung 2.11.: Beispiele für die Modellierung von Menschen: (a) Modellierung mittels Skelett, Metaballs und Hülle (Plänkers und Fua, 2003). (b) Modellierung mittels Super-Quadrics (Gavrila und Davis, 1996). (c) Super-Quadrics-Modell (Sundaresan und Chellappa, 2006).

Es werden hier keine Silhouetten oder visuelle Hüllen extrahiert. Aus dem optischen Fluss in den Bildern der einzelnen Kameras wird versucht, die dreidimensionale Bewegung des Körpermodells zu extrahieren. Das verwendete Modell besteht aus sogenannten Super-Quadrics (siehe Abb. 2.11c).

Ein Ansatz, der maßgeblich auf der Kantenanalyse in allen vier Mutli-View Aufnahmen basiert, wird von Gavrila und Davis (1996) vorgestellt. Auch hier wird ein Modell bestehend aus Super-Quadrics (ebenso wie von Sundaresan und Chellappa (2006) verwendet) genutzt, um eine Pose-Estimation der beobachteten Person zu berechnen. In allen vier Aufnahmen wird eine Kantenanalyse durchgeführt. Anschließend wird das Modell in verschiedenen Konfigurationen ins Bild projiziert und per Chamfer-Matching mit den extrahierten Kanten verglichen. Zur Initialisierung der Modellparameter wird eine Hintergrundeliminierung benutzt und anschließend mittels Hauptkomponentenanalyse die Hauptachse der Person aus den extrahierten Vordergrundpixel bestimmt. Das System ist in der Lage, einfache Bewegungen (z.B. Winken), aber auch komplexe Bewegungen mehrerer Personen zu tracken (z.B. Paar tanzt Tango, siehe Abb. 2.11b). Eine Erweiterung dessen wird von Hofmann und Gavrila (2009) vorgestellt. Hier werden die Komponenten Pose-Estimation, Bewegungsintegration und Modelladaption miteinander verknüpft, was zu einer Erhöhung der Systemrobustheit führt.

Wertung: Die Genauigkeit der Multi-View Ansätze ist sehr hoch und wünschenswert für die Applikation, die in der vorliegenden Arbeit betrachtet wird. Jedoch ist der Aufwand für solche Ansätze sehr hoch. Zum einen werden mehrere verteilte Kameras benötigt, welches im industriellen Produktionsszenario nicht immer möglich ist. Zum anderen ist die Rechenzeit ebenfalls sehr hoch. Die beispielhafte Rechenzeit für einen

Zeitschritt für das Verfahren von Rosenhahn et al. (2007a) beträgt ca. fünf Minuten. Dies macht die Anwendbarkeit in Echtzeit auf heutigen Systemen unmöglich.

2.2.3. Stereo

Im Gegensatz zu den aufgeführten Multi-View Ansätzen werden bei der Stereobildverarbeitung lediglich kleine Basisbreiten und zwei oder drei Kameras benutzt.

Von Plänkers und Fua (2003) wird ein sehr detailliertes Modell benutzt, um die Pose von Personen zu extrahieren und anschließend zu tracken. Der Ansatz basiert auf Stereo-Punkten aus einem trinokularen Kamerasystem und der extrahierten Silhouette der Person. Außerdem wird in dem Ansatz die Form des Modells entsprechend der beobachteten Person adaptiert. Das Modell besteht dabei aus einem Skelett auf dem 230 sogenannte Metaballs sitzen (siehe Abb. 2.11a). Durch die Modelladaption werden sowohl Skelett als auch Metaballs optimiert, um ein möglichst gutes Abbild der beobachteten Person zu ermitteln. Durch die Stereodaten und ins Bild projizierte Umrisse wird zunächst Zeitschritt für Zeitschritt die Pose des Modells ermittelt. Anschließend wird eine Optimierung über die ganze Sequenz gestartet, welche gleichzeitig die Pose verfeinert und eine Modelladaption durchführt, um die Parameter des Skeletts, aber auch die Parameter der Metaballs zu optimiert.

Von Delamarre und Faugeras (1998) wird ein detailliertes Hand-Modell in Kombination mit Stereobildverarbeitung dazu benutzt, die Pose der gesamten Hand inklusive aller interner Freiheitsgrade (insgesamt 27) zu ermitteln. Die Pose-Parameter werden über die Zeit mittels Kalman-Filterung getrackt. Ein ICP-Ansatz übernimmt die Pose-Estimation mittels 3D-Modell und 3D-Punktewolke. Das Modell besteht dabei aus Kegelstümpfen für die rigiden Abschnitte und aus Kugeln für die Gelenke.

Von Narayanan et al. (2005) wird ein System zur Verfolgung von Köpfen in Bildsequenzen vorgestellt. Dabei werden zunächst Stereo-Daten aus der Szene extrahiert und eine Vordergrund-Segmentierung durchgeführt. Für diese Segmentierung wird der Hintergrund mittels Disparitätsstatistiken modelliert. Weicht der Disparitätswert nun von der modellierten Statistik ab, handelt es sich um ein Objekt im Vordergrund. Durch eine Zusammenhangsanalyse wird nun die Person im Vordergrund extrahiert. Anschließend wird eine Ellipse, deren Maße durch die Entfernungsmessung über die Disparitätswerte bestimmt wird, als grobes Kopf-Modell benutzt. Das Tracking erfolgt durch den Vergleich des Bildinhalts in zwei aufeinanderfolgenden Frames und der Initialisierung der Bewegung über die Annahme einer konstanten Geschwindigkeit.

Von Ziegler et al. (2006) wird ein Oberkörpermodell mittels ICP an eine Stereo-Punkt-Wolke angepasst, welche mittels mehrerer Stereokameras, die um die Person verteilt sind, generiert wird. Dabei wird das Modell selbst nicht durch geometrische Formen, sondern durch einzelne Punkte repräsentiert. Dadurch entfällt eine Point-to-Plane Korrespondenzsuche und das Problem kann auf den ursprünglichen Point-to-Point Ansatz zurückgeführt werden. Anschließend wird die Pose inklusive aller interner Parameter mittels Unscented Kalman Filter verfolgt. Die Positionsgenauigkeit liegt hier bei ca. 7 cm, wobei der Vergleich auf einer manuell gelabelten Ground-Truth basieren.

Auch von Demirdjian und Darrell (2002) wird ein Oberkörpermodell, welches aus

drei Zylindern und einer Kugel besteht, mittels ICP an eine 3D-Punktewolke angepasst. Dabei werden zunächst die Köperteile einzeln angepasst und dann, unter Verwendung verschiedener Constraints, eine Gesamtpose des Oberkörpers ermittelt. Ein Aussage über die Genauigkeit der Pose-Estimation wird nicht gemacht. Von Demirdjian (2003) werden zusätzlich dazu noch weitere Constraints eingeführt, beispielsweise für die maximale Bewegungsgeschwindigkeit und die maximalen Gelenkwinkel.

Das von Lin (1999) vorgestellte Verfahren ist in der Lage, die Hand-Arm-Region einer Person über die Zeit in Bildsequenzen zu verfolgen. Grundlage für die Methode ist eine dichte Tiefenkarte der Szene. Die Hand-Arm Region wird dabei durch drei überlappende Ebenen dargestellt, die an Regionen gleicher Tiefe angepasst werden. Dabei wird die Annahme getroffen, dass die Achsen des Ober- und Unterarms und der Hand stets senkrecht zur optischen Achse des Sensors sind. Eine Aussage über die Genauigkeit der Verfolgung wird nicht gemacht.

Das VooDoo System wird von Knoop et al. (2006) vorgestellt und fusioniert die Daten verschiedener Sensoren zur Schätzung der menschlichen Pose. Dabei können 3D-Punkte aus einem Stereo Algorithmus genauso wie Tiefenpunkte eines Time-of-Flight-Sensors und 2D-Merkmale in monokularen Aufnahmen zum Einsatz kommen. Basierend auf einem Kegelstumpf-Modell und einer ICP-Optimierung werden die verschiedenen Eingangssignale dazu benutzt, die Pose der jeweiligen Person möglichst genau zu ermitteln. Die Modellanpassung mittels ICP erfolgt über eine Point-to-Plane Korrespondenzbildung, die iterativ nach dem Pose-Update jeweils wiederholt wird.

In den Arbeiten von Hahn et al. (2007, 2008a,b) wird der sogenannte Multiocular Contracting-Curve-Density (MOCCD) Algorithmus zur Poseschätzung der menschlichen Hand-Unterarm Region benutzt. Der MOCCD basiert auf dem CCD Algorithmus, welcher in der Lage ist, Kontur-Modelle an eine Grauwertkante im Bild anzupassen. Das Verfahren ist ein Pose-Refinement und benötigt daher eine Initialpose. Vorteil dieses Verfahrens gegenüber beispielsweise Chamfer-Matching Techniken ist das sich selbst-adaptierende Separierungskriterium, mit dessen Hilfe iterativ die Konturanpassung durchgeführt wird. Dabei werden Pixelstatistiken innerhalb und außerhalb der Kontur ermittelt und solange durch Poseveränderungen optimiert, bis sich die Statistiken größtmöglich unterscheiden. Die Kontur wird durch die Projektion eines 3D-Modells in die multiokularen Aufnahmen berechnet. Für das Verfahren sind keine Vorberechnungen wie Stereobildverarbeitung oder Bildrektifizierung notwendig, da direkt auf den Kamerabildern gearbeitet wird. Zur zeitlichen Verfolgung wird ein Multi-Hypothesen-Ansatz zur Pose Initialisierung im nächsten Zeitschritt verwendet, wonach dann für jede Hypothese ein Pose Refinement durchgeführt wird. Ein Winner-takes-All Ansatz wählt dann die beste Hypothese für diesen Zeitschritt aus. Hahn et al. (2008b) erweitern das Verfahren zum sogenannten ShapeFlow-Algorithmus. Dieser ist in der Lage, ohne zeitliche Filterung die Pose und deren zeitliche Ableitung, die Bewegung, durch gleichzeitige Pose-Anpassung in drei aufeinanderfolgenden Frames, zu ermitteln. Von Hahn et al. (2008a) wird eine weitere Arbeit vorgestellt, die auf diesem Ansatz aufbaut. Hier werden die durch den MOCCD ermittelte Trajektorien zur Handlungserkennung und Langzeitprädiktion benutzt.

Wertung: Bezüglich Hardwareaufwand und Rechenzeitanforderung ist der Stereoansatz zur raum-zeitlichen Pose-Estimation vielversprechend. Es ist mehr gefordert, als bei monokularen Ansätzen, jedoch wird der Aufwand der Multi-View Systeme vermieden. Die Robustheit für die angestrebte Applikation ist auch gegeben, wie beispielsweise die Arbeiten von Hahn et al. (2008b) zeigen. Bezüglich der Genauigkeit müssen hier noch Fortschritte angestrebt werden.

2.3. Sichere Mensch-Roboter Interaktion

2.3.1. Überblick

Das Ziel der sicheren Mensch-Roboter Interaktion ist es, durch die Interaktion bzw. Kooperation zwischen Mensch und Roboter die Handlungen beider in der Art zu vereinen, dass die Produktivität gesteigert wird und gleichzeitig kein Sicherheitsrisiko für den Arbeiter entsteht. Vorteile, die sich aus der Interaktion ergeben, betreffen neben der Kombination der Fähigkeiten beider auch den platzsparenden Aufbau der Anlage, die schnelle Bearbeitung, die hohe Ergonomie und die zusätzlich möglichen Anwendungen, die sich durch die Interaktion ergeben. Industrieroboter sind schnell, stark, ausdauernd und positionsgenau und eignen sich insbesondere für sich wiederholende Arbeiten. Der Mensch dagegen ist in komplizierten Handhabungsarbeiten unerreicht geschickt und kann sehr flexibel auf ungeplante Situationen reagieren. Als Anwendungsgebiete kommen vorallem die Produktion, aber auch die Dienstleitung, die Medizintechnik und der Haushalt in Betracht.

Aus Sicherheitsgründen müssen zur Zeit die Arbeitsräume von Industrierobotern und Menschen durch Schutzeinrichtungen getrennt werden. Gesetzlich ist diese Voraussetzung an verschiedenen Stellen verankert: In Teil 1 und 2 der Industrieroboternorm *ISO10218* und auch in der Maschinenrichtlinie *98/37/EG* des Europäischen Parlaments. Industriell eingesetzte Sicherheitssysteme überwachen daher sicher das Eindringen des Menschen in die Roboterzelle, was für den Fall des Eindringens und einer somit eintretenden potenziellen Gefährdung des Arbeiters einen Stillstand des Industrieprozesses zur Folge hat. Dabei unterscheidet man zwischen traditionellen Schutzgittern, die ein Eindringen von Personen physikalisch verhindern, und der OTS-Strategie (ohne trennende Systeme). Diese Systeme nutzten meist Laserscanner oder andere optische Verfahren, um ein Eindringen von Personen zu erkennen und daraufhin die Anlage zu stoppen.

Diese strengen Anforderungen sind damit begründet, dass die Roboter ihre Umwelt nicht wahrnehmen können und damit das Risiko einer Kollision zwischen Mensch und Roboter zu groß ist. Da der Roboter seinen Arbeitsbereich nicht selbst überwacht, kann ein Mensch, der die Roboterzelle betreten hat, in die Nähe des Roboters gelangen und dessen Trajektorie kreuzen, was eine Kollision zwischen Roboter und Mensch zur Folge haben kann. Dies ist zum Schutz des Menschen strengstes zu verhindern und unterbindet daher eine optimale Arbeitsteilung durch einen stetigen Wechsel von manuellen und automatischen Tätigkeiten auf Prozessebene.

Die grundlegende Fragestellung bei der sicheren Mensch-Roboter-Interaktion ist: Wie

gut kann die Sicherheit des Menschen bei einer sinnvollen Art der Kooperation gewährleistet werden? Einen Überblick über den Stand der Forschung auf diesem Gebiet bieten beispielhaft folgende Veröffentlichungen: (Goodrich und Schultz, 2007; Thiemermann, 2002; Ebert, 2003).

Auf Grund der beschriebenen sicherheitstechnischen Vorschriften befinden sich alle im folgenden beschriebenen Systeme (außer das in Abschnitt 2.3.5 beschriebene SafetyEYE-System) noch im Forschungsstadium. Diese Forschungsaktivitäten zeigen, dass zur Überwachung von möglichen Kollisionen zwischen Mensch und Roboter verschiedene Sensorkonzepte zum Einsatz kommen können, z.B. kapazitive Hüllen, Ultraschall oder Motorstromüberwachung. Allen voran wird jedoch die Bildverarbeitung als zielführend in diesem Forschungsbereich angesehen. Beispielhaft für5 die Vielzahl an Forschungsaktivitäten sind Projekte wie „Die sichere Mensch-Roboter-Kooperation“ der Kuka AG, der „Cooperating Robot“ der ETH Zürich und das System des National Institute of Industrial Safety aus Japan (Thiemermann, 2002).

Im Folgenden werden beispielhaft einige Systeme zur sicheren Mensch-Roboter Interaktion beschrieben und analysiert.

2.3.2. Überwachung mittels PMD-Sensorik

Winkler (2007) stellen einen Ansatz zur Überwachung des Arbeitsbereichs eines Roboters mit einer Tiefenbildkamera (PMD) vor. Zusätzlich zu den Daten aus der Kamera werden die Gelenkwinkel des Roboters genutzt, um diesen in der Szene modellieren zu können. Durch dieses Robotermodell mit der aktuellen Roboterpose kann der Roboter im aufgenommenen Tiefenbild unterdrückt werden. Für jedes Robotersegment ist ein Quader als Schutzzone definiert, dessen Größe von der aktuellen Geschwindigkeit des Roboters abhängt. Die gemessenen dreidimensionalen Punkte des Tiefenbildes können nun auf eine Verletzung der definierten Schutzzonen überprüft werden. Dazu werden die Punkte, die nach der Unterdrückung des Roboters übrig bleiben, entsprechend verschiedener Sicherheitsregeln analysiert.

Vorteilhaft an der Verwendung dieser PMD-Technologie ist eine direkte Bereitstellung von Tiefeninformationen. Nachteilig ist jedoch, dass die Auflösung einer solchen Kamera meist niedrig ist, wodurch die Genauigkeiten der Szenenanalyse ebenfalls gering sind.

2.3.3. Das Projekt team@work

Thiemermann (2002) und Spiengler und Thiemermann (2002) stellen das *team@work*-System vor, ein System, das die gemeinsame Montage von kleinen Werkstücken durch Mensch und Roboter ermöglicht. Dafür sind über dem gemeinsamen Arbeitsbereich drei Kameras angebracht. Es wurde ein Bildverarbeitungsalgorithmus entwickelt mit dessen Hilfe Hände und Nacken aufgrund der charakteristischen Farbe und Textur erkannt werden. Der Abstand zwischen detektierten menschlichen Körperteilen und dem Roboter wird zur Begrenzung der maximalen Geschwindigkeit des Roboters verwendet. Entsteht eine Gefahrensituation wird die Geschwindigkeit des Roboters gedrosselt oder der Roboter ganz zum Stillstand gebracht.

Die Arbeitsraumüberwachung besteht aus einem 3D-Bildverarbeitungssystem, das mit drei CCD-Farbbildkameras ausgestattet ist. Über eine Kombination des Referenzbildverfahrens und einer Farb- und Texturerkennung werden permanent Position, Geschwindigkeit und Beschleunigung von Werker und Roboter bestimmt. Für das Überwachungssystem sind oberhalb der Anlage in einer Höhe von ca. vier Metern drei Kameras angebracht. Es handelt sich hierbei um CCD-Kameras, die Farbbilder in VGA-Qualität (640x480 Pixel) liefern. Eine Kamera befindet sich direkt über dem Arbeitsplatz und schaut senkrecht herunter. Die anderen beiden sind in der gleichen Höhe links und rechts über dem Arbeitsplatz installiert, um zum einen dreidimensionale Informationen zu erhalten und zum anderen bei möglichen Verdeckungen des Werkers durch den Roboter immer mindestens zwei Kameras, die von ihrem Standpunkt aus den Arbeitsbereich noch komplett sehen, zu haben. Um Störungen der Bildverarbeitung weitestgehend zu unterbinden, wird zusätzlich eine spezielle Beleuchtung angebracht, die mit konstanter Frequenz strahlt. Weitere Informationen zur Bildverarbeitungsalgorithmik namens *Control Vision* sind nicht verfügbar.

Allerdings werden Kollisionen mit anderen Teilen des Menschen oder anderen Objekten wie Werkstücke nicht detektiert. Es ist auch unklar, ob das System mit großen Werkstücken oder bekleideten Körperteilen umgehen kann. In der Dissertation von Thiemermann (2002) liegt das Hauptaugenmerk auf dem Zusammenspiel aller Komponenten des Interaktionssystems und der Auslegung bzw. der Planung des jeweiligen Arbeitsplatzes.

2.3.4. Das Projekt SIMERO

Das Projekt *SIMERO* (Sichere Mensch/Roboter-Kooperation und -Koexistenz) der Universität Kaiserslautern hatte zum Ziel, ein Überwachungssystem zur sicheren Kooperation zwischen Mensch und Roboter zu entwickeln. Das Überwachungssystem besteht dabei aus VGA-Farbkameras bzw. Tiefenbildsensoren, die am Rand der Roboterzelle montiert sind. Durch die Kalibrierung des Kamerasystems lässt sich der Arbeitsbereich dreidimensional überwachen. Zur Segmentierung der Kamerabilder in Freiraum bzw. unbekannte Objekte kommen Background-Modeling- bzw. Change-Detection-Verfahren zum Einsatz. Dabei werden Silhouetten aus den einzelnen Bildern extrahiert und über ein Referenzbildverfahren Änderungen in der Szene registriert. Neben den Farbkameras wird auch die Verwendung von Tiefenbildsensoren, ebenso wie Winkler (2007), untersucht.

Eine Kollisionserkennung, basierend auf den Silhouettendaten des Roboters und den anderen Objekten im Raum, leitet Informationen über eine drohende Kollision zur Bahnplanung weiter. In der Bahnplanung wird dann eine kollisionsfreie Bahn ermittelt oder notfalls die Roboterbewegung eingeschränkt bzw. gestoppt.

Bei diesem Verfahren wird keine Unterscheidung zwischen Personen und anderen Gegenständen, die evtl. keine Gefahr darstellen, gemacht. Außerdem wird die Szene stets statisch bedrachtet, d.h. es wird keine Aussage über die Bewegung der Objekte gemacht und somit auch keine Prädiktion der Objektposition.

2.3.5. Das SafetyEYE System

Das SafetyEYE-System der Firma Pilz[1] ermöglicht die simultane Überwachung von bis zu 50 statischen, im dreidimensionalen Raum festlegbaren Schutzzonen, die von einem Kamerastandpunkt aus mittels drei eng beieinanderliegender Kameras unter Verwendung von Stereoalgorithmen überwacht werden. Es kommen dabei zwei redundant arbeitende Verfahren zum Einsatz: ein korrelationsbasiertes und ein konturbasiertes Stereoverfahren. Die Erkennung einer Schutzzonenverletzung sowie die Reaktion darauf ist innerhalb einer Grenze von 350 ms gewährleistet. Dabei werden nur statische Daten, d.h. synchron aufgenommene Daten aus einem Zeitschritt verwendet und keine dynamischen Bewegungsdaten.

Quadcam[2] ist ein weiteres Sicherheitssystem, welches ähnlich wie SafetyEYE aus einer Perspektive pro Überwachungsmodul eine statische Schutzzone überwachen kann.

2.3.6. Das Projekt LiSA-Roboter

Das Mitte 2009 abgeschlossene Verbundvorhaben LiSA wurde mit Mitteln des Bundesministeriums für Bildung und Forschung (BMBF) gefördert und hatte die Entwicklung und Erprobung eines Assistenzroboters zum Einsatz in Laborumgebungen von Life-Science Unternehmen zum Ziel (Elkmann et al., 2008). Im Vordergrund stand dabei auch die Sicherheit bei der Zusammenarbeit mit dem Roboter.

Zentrale Entwicklungsziele waren bei dem Projekt:

- Entwicklung und Aufbau einer mobilen Plattform für die spezifischen Anforderungen unter Laborbedingungen
- Navigation in dynamischer Umgebung, flexibler Einsatz bei sich ändernder Laborausstattungen
- Entwicklung eines Manipulator und eines Sicherheitskonzeptes, das die Interaktion mit Menschen unter Berücksichtigung der Erfüllung aller Sicherheitsanforderungen ermöglicht
- Bestimmung der Aufnahme- und Ablagepositionen und der handzuhabenden Objekte
- sensorgeführte Bewegung des Manipulators
- multimodale Mensch - Assistenzsystem Kommunikation

Zur Navigation in den Laborräumen dient eine mobile Plattform zusammen mit einer interaktiv gelernten Karte. Die Kommunikation zwischen Mensch und Roboter wird in erster Linie über Sprachein- und -ausgabe ermöglicht.

[1] Internetseite des SafetyEYE-Systems der Firma Pilz: *http://www.safetyeye.de/*
[2] Internetseite des Quadcam-Systems der Firma Castell: *http://www.castell.com/*

In dem Projekt wurden verschiedene Sicherheitsmechanismen untersucht, um den Menschen bestmöglich vor Gefahren, die durch den Serviceroboter und speziell durch den Manipulator entstehen, zu schützen.

Zu den Sicherheitskonzepten, die im Rahmen des Projektes untersucht werden, gehören:

- reduzierte Masse
- geringe Geschwindigkeiten
- Sensoren zur Kollisionsvermeidung bzw. -erkennung (Thermographie, taktiler Sensor (künstliche Haut))
- mechanische Komponenten zum Schutz vor Verletzungen (Nachgiebigkeit, Klemmschutz, Polsterung)

Der LiSA-Roboter verfügt über drei Kameras: zwei Graustufen-Kameras und eine Thermografie-Kamera. Die beiden Graustufen-Kameras werden gemeinsam mittels Stereobildverarbeitung dazu benutzt, Interaktionsobjekte zu erkennen. Die Thermografie hingegen ist ausschließlich für die Sicherheit der Personen zuständig. Bei jeder Bewegung des Manipulators wird auf Basis der Temperaturinformationen analysiert, ob es zu einer Gefährdung der Personen kommen könnte. Dabei werden Thermobildfolgen genutzt, um die menschliche Bewegung zu erkennen und gegebenenfalls die Aktion des Manipulators zu stoppen.

Zusätzlich zum kamerabasierten Sicherheitskonzept kommt auch ein taktiler Sensor am Manipulator zum Einsatz, um Kollisionen zu erkennen und darauf zu reagieren. Der Sensor erfasst die einwirkenden Kräfte und stoppt daraufhin bei Überschreiten eines definierten Grenzwerts den Manipulator. Dadurch sind Berührungen erlaubt, Verletzungen werden aber verhindert. Dies bedingt jedoch eine geringe Geschwindigkeit des Manipulators, um die Reaktionszeit des Sensors und dessen Signalverarbeitung auszugleichen.

Abschließend kann gesagt werden, dass die Projektziele erreicht wurden und der daraus entstandene Roboter jetzt in weiteren Tests seine Einsetzbarkeit in Life-Science Unternehmen beweisen muss.

2.3.7. Wertung

Aus den Eigenschaften und der Bewertung der einzelnen Systeme aus dem Stand der Forschung lässt sich ableiten, dass gerade auf dem Gebiet der Überwachung schon sehr fortgeschrittene Entwicklungen vorliegen. Da viele der vorgestellten Systeme jedoch auf Farbbildverarbeitung zur Erkennung der Haut aufbauen, wird vorausgesetzt, dass keine Handschuhe o.ä. getragen werden, was in der realen Produktion nicht realisierbar ist. Das SafetyEYE System arbeitet nicht unter dieser Voraussetzung, es arbeitet auf Grauwertbildanalyse. Ein Nachteil an diesem System ist, dass keine Unterscheidung zwischen Personen und unkritischen Gegenständen stattfindet, wodurch unnötige Stillstände der

Anlage verursacht werden können. Außerdem wird in dem System keine Bewegungsanalyse des Werkers durchgeführt, wodurch keine Prädiktion erfolgen kann und somit die Schutzzonen statisch definiert werden müssen.

2.4. Erkennung und Verfolgung von Fahrzeugen in Bildsequenzen

Hinter der bildbasierten Erkennung und Verfolgung von Fahrzeugen im Straßenverkehr stehen drei hauptsächliche Anwendungen. Zum einen sollen Sicherheitssysteme durch das Wissen um andere Verkehrsteilnehmer in der Lage sein, durch Brems- bzw. Lenkeingriffe gefährliche Situationen auf der Straße zu vermeiden. Zum anderen wird die Distanz zum vorausfahrenden Fahrzeug ermittelt, um damit einen Abstandsregeltempomaten (engl.: Adaptive Cruise Control, ACC) zu steuern. Außerdem wird die Vision vom autonom fahrenden Auto weiter verfolgt. Durch die Veranstaltungen *Grand Challenge* und *Urban Challenge* des US-amerikanischen Verteidigungsministeriums DARPA[3] wurde eine schnelle Entwicklung auf diesem Gebiet forciert.

In der Literatur werden verschiedene Systeme beschrieben, die sich mit der Detektion anderer Fahrzeuge beschäftigen und eine Distanzschätzung zwischen dem eigenen und dem detektierten Auto durchführen. In der Zusammenfassung von Sun et al. (2006) wird auf verschiedene Fahrzeugsdetektionssysteme eingegangen, die alle auf Bildverarbeitung basieren. Es werden sowohl monokulare als auch binokulare Stereosysteme charakterisiert und ein Ausblick auf zukünftige Herausforderung auf diesem Gebiet gegeben.

Im Folgenden sollen einige Systeme vorgestellt werden, um den aktuellen Stand der Forschung wiederzugeben. Dabei sind die Arbeiten nach der verwendeten Sensorik geordnet.

2.4.1. Monokulare Systeme

Ein System zur Segmentierung und Detektion von Fahrzeugen mithilfe eines monokularen Kamerasystems wird von Collado et al. (2004) vorgestellt. Dazu wird ein Fahrzeugmodell, welches durch sieben Parameter beschrieben wird, verwendet. Ein Algorithmus bestimmt die Position des Fahrzeugs unter Berücksichtigung der Symmetrie der Kanten im Bild, der Form des Modells und des Schattens des Fahrzeugs. Eine Evaluierung dieser Methode ist nicht verfügbar.

Von Dellaert und Thorpe (1997) wird ein System vorgestellt, welches die Position von vorausfahrenden Fahrzeugen mittels Kalman-Filtering verfolgt. Mittels Hough-Transformation werden die Konturen anderer Fahrzeuge erkannt und über eine Bounding-Box (engl. für eine umschließende 2D- oder 3D-Form) zusammengefasst. Außerdem wird durch einen Klassifikator der Bildinhalt der Bounding-Box nach Fahrzeugen durchsucht. Ein 3D-Modell wird für das gefundene Fahrzeug benutzt und im 3D-Raum verfolgt, wobei die Messung jeweils die 2D-Projektion dieses 3D-Modells ist. Somit ist die 3D-Position

[3] Internetseite des US-amerikanischen Verteidigungsministeriums DARPA: *http://www.darpa.mil*

des Fahrzeugs verfügbar, die in der Evaluation auch genauer betrachtet wird. Für ein monokulares System werden gerade in der Abstandsschätzung, welche mittels Radardaten evaluiert wird, sehr gute Werte erreicht, d.h. der Fehler der Abstandsschätzung ist kleiner 1 m bei einem mittlerem Abstand von etwa 60 m zum vorausfahrenden Fahrzeug.

Im Bezug auf die Verkehrsüberwachung wurden wichtige Arbeiten von Koller et al. (1993), von Kollnig und Nagel (1997) und von Haag und Nagel (1999) veröffentlicht. Die Grundlage ist dabei die Beobachtung einer Straßenkreuzung mittels einer fest installierten Kamera. Dazu wird zunächst in der monokularen Bildsequenz eine Kantenanalyse durchgeführt. Die Fahrzeuge werden als Polyeder-Modelle dargestellt. Die Modellkanten werden an die Grauwertkanten im Bild angepasst, um die Pose des Fahrzeugs zu ermitteln. Für die anschließende Objektverfolgung in der Bildsequenz werden mehrere Randbedingungen betrachtet. Ein 3D-Szenenmodell gibt Aufschluss über mögliche Fahrkorridore und über mögliche Verdeckungen. Außerdem wird ein Beleuchtungsmodell dazu verwendet, Schatten besser aus der Pose-Estimation ausblenden zu können. Ein Bewegungsmodell beschreibt physikalisch die möglichen Bewegungen der Fahrzeuge, um eine bessere Prädiktion zu erhalten. Die Analyse des optischen Flusses in Kombination mit einem Extended Kalman Filter ermöglichen dann die robuste Verfolgung in der Bildsequenz.

2.4.2. Sensor-Fusions Systeme

Von Kato et al. (2002) wird ein System bestehend aus einer Kamera und einem Radarsystem dazu benutzt, andere Verkehrsteilnehmer zu erkennen. Dabei werden die Vorteile beider Sensoren genutzt, um möglichst genau die Position des vorausfahrenden Fahrzeugs zu bestimmen. Das Radar hat eine geringe laterale Auflösung, ist aber in der Lage die Distanz zu einem Objekt genau zu bestimmen. Die Kamera hingegen hat orthogonale Eigenschaften: eine hohe laterale Auflösung und eine schlechte Distanzschätzung. Daher wird die Distanz zum Objekt über die Radarinformation extrahiert und die Objektabmessungen werden über die Kamera geschätzt. Die Evaluierung beschränkt sich auf die Genauigkeit der Bounding-Box im Bild.

Von Steux et al. (2002) wird eine Kombination aus monokularer Farbkamera und Radarsensor vorgestellt. Hierbei werden die Ergebnisse aus vier verschiedenen Bildverarbeitungsalgorithmen mit den Radarmessungen verknüpft. Aus diesen Informationen wird über eine low-level Fusion die aktuelle Position des anderen Fahrzeugs extrahiert. In der Bildverarbeitung werden Schatten, Rücklichter, Liniensegmente und symmetrische Merkmale im Bild erkannt. In einem Rückschlußprozess wird nach der Fusion mit den Radardaten versucht, falsche Detektionen auszuschließen und die richtige Position der Fahrzeuge zu ermitteln. Eine metrische Evaluierung der Ergebnisse ist nicht verfügbar.

Wender und Dietmayer (2007) stellen einen Fusionsansatz basierend auf einer Kamera und einem Laserscanner vor, um damit Fahrzeuge zu detektieren. Dabei werden Objekthypothesen für Fahrzeuge zunächst aus den Punkten des Laserscanners generiert und die Pose des Fahrzeugs ermittelt. Anschließend wird mithilfe dieser Poseinformation die Seitenansicht des Fahrzeugs aus dem Kamerabild ausgeschnitten und entsprechend einer frontoparallelen Ansicht der jeweiligen Seite rektifiziert. Diese normalisierte Ansicht der

Fahrzeugseite wird anschließend durch einen Kaskadenklassifikator klassifiziert, um abschließend eine Aussage zu erhalten, ob es sich bei dem Objekt um ein Fahrzeug handelt oder nicht. In der Veröffentlichung wird nur auf die Güte des Klassifikators eingegangen, jedoch nicht auf die metrische Genauigkeit der Pose-Estimation in den Laserscannerdaten.

Von Kämpchen (2007) wird ebenfalls eine Kombination aus Laserscanner und Kamera zur Objekterkennung im Straßenverkehr benutzt. Zur Fusion der Daten der beiden Sensoren wird eine merkmalsbasierte Fusion benutzt. Es wird also eine getrennte Vorverarbeitung durchgeführt, um Merkmale des gleichen Fahrzeugs aus beiden Sensoren zu erhalten. Anschließend werden diese Merkmale fusioniert, was als Basis zur endgültigen Fahrzeugerkennung dient. In der Arbeit wird nicht nur das Folgefahren als Applikation betrachtet, sondern auch Stausituationen und Kreuzungsszenarien untersucht. Hier wird gezeigt, dass das System auch bei hohen Beschleunigungsverhalten der beobachteten Fahrzeuge in der Lage ist, diese in den Bildern zu verfolgen.

2.4.3. Stereo-Systeme

Von Nedevschi et al. (2007) wird ein System mit verschiedenen Applikationen vorgestellt, welches auf einer Stereokamera basiert. Dabei wird sowohl eine Spurverlauf-, Fahrzeug- und Fussgängererkennung und -verfolgung durchgeführt, als auch der befahrbare Bereich vor dem eigenen Fahrzeug erkannt. Um Hindernisse bzw. andere Verkehrsteilnehmer initial erkennen zu können, wird zunächst eine Tiefenkarte mittels Stereobildverarbeitung erstellt. Anschließend wird ein sogenanntes *Occupancy-Grid* (engl. für Belegungskarte) implementiert, welches den Bereich vor dem Fahrzeug in einzelne „freie“ und „besetzte“ Segmente unterteilt. Nachdem größere besetzte Bereiche zu jeweils einem Objekt zusammengefasst wurden, werden für jedes Objekt die Parameter Position, Größe, Längenverhältnis und Orientierung ermittelt. Anschließend werden die Objekte basierend auf Größe und Längenverhältnis klassifiziert, ein geeignetes 3D-Modell entsprechend der Pose ins Bild projiziert und mittels Chamfer-Matching angepasst. Eine Aussage über metrische Genauigkeiten wird in der Veröffentlichung nicht gemacht.

An der ETH Zürich wurden verschiedene Methoden entwickelt, um die Umwelt um das eigene Fahrzeug analysieren zu können. Leibe et al. (2006) beschreiben ein umfangreiches System, welches zunächst auf der Fusion von Structure-from-Motion mit Stereobildverarbeitung basiert. Anschließend werden die vorhandenen Objekte detektiert und klassifiziert mit dem Ansatz von Leibe und Schiele (2004). Die Detektion gibt dabei die Objektorientierung in groben 30°-Schritten an. In dieser Veröffentlichung wird wiederum keine Aussage bezüglich der metrischen Genauigkeit der Methoden gemacht.

Von Barth und Franke (2008) wird ein System zur Verfolgung anderer Fahrzeuge auf Basis einer Stereokamera vorgestellt. Dabei werden in der beobachteten Szene zunächst sogenannte 6D-Vision-Vektoren etabliert (siehe Kap. 2.1.2). Anschließend werden diese Vektoren nach Ort und Bewegung segmentiert und somit Objekte gebildet. Diese Objekte werden anschließend mittels Kalman-Filterung über die Zeit verfolgt. Auf synthetischen Sequenzen funktioniert das Verfahren sehr gut, wobei auch hier schon eine relativ große Einschwingzeit von ca. 20 Zeitschritten benötigt wird. Eine Genauigkeitsabschätzung

auf realen Daten steht nicht zur Verfügung.

Eine Erweiterung dieses Verfahrens wird von Hermes et al. (2009) vorgestellt. Hier werden die Daten dieser Bewegungsanalyse mit einer Trajektorienklassifikation verknüpft, um eine möglichst genaue Langzeitprädiktion der Fahrzeugbewegung zu erhalten.

Dang et al. (2002) verwenden eine Kombination von Stereo-Punkten mit optischem Fluss, um die Position und Geschwindigkeit anderer Fahrzeuge zu ermitteln. Dabei werden die Informationen aus der Stereobildverarbeitung mit dem extrahierten Flussfeld zusammen in einem Extended-Kalman-Filter Ansatz verarbeitet. Anschließend lassen sich auf Basis der Einzelpunkte Objekte durch Clustering etablieren und deren Geschwindigkeit schätzen. Auch hier zeigt sich wieder ein Einschwingverhalten bei der Geschwindigkeitsschätzung, was durch das verwendete Kalman-Filter bedingt ist. Eine Analyse der Genauigkeit fehlt in der Veröffentlichung.

Von Toulminet et al. (2006) wird eine Stereokamera dazu benutzt, vorausfahrende Fahrzeuge zu detektieren und deren Distanz zum eigenen Fahrzeug zu ermitteln. Das System besteht hauptsächlich aus zwei Verarbeitungsstufen. Zuerst wird eine 3D-Rekonstruktion der Szene vor dem eigenen Fahrzeug mittels Stereobildverarbeitung berechnet. Anschließend wird in nur einem der beiden Kamerabilder eine Symmetrieanalyse durchgeführt. Man geht dabei davon aus, dass die Rückfront von Fahrzeugen eine gewisse Symmetrie im Bild darstellt. Es wird dadurch eine Bounding-Box um das detektierte Fahrzeug berechnet und anschließend mit den zuvor ermittelten 3D-Daten verknüpft. In den folgenden Zeitschritten wird das Fahrzeug lediglich zweidimensional im Bild verfolgt und jeweils aktuell gemessene 3D-Daten damit verknüpft.

2.4.4. Wertung

Die Orientierung des detektierten Autos bleibt in vielen Veröffentlichungen unbestimmt, ist jedoch für eine Bewegungsprädiktion notwendig. Außerdem fehlt meist eine Evaluierung anhand einer unabhängig bestimmten Ground-Truth. Die Bewegungsinformation wird bei den meisten Verfahren durch ein Kalmanfilter aus der Bildsequenz extrahiert. Dies bedingt ein Einschwingverhalten, was im Kreuzungsbereich hinderlich ist.

Die monokularen Systeme erreichen grundsätzlich keine ausreichende Genauigkeit bei der Abstandsschätzung. Das Problem bei den Sensorfusionssystemen ist der hohe Aufwand. Eine Kamera-Laserscanner-Kombination ist derzeit zu kosteninstensiv für den Serieneinsatz. Eine vielversprechende Alternative ist das Stereokamerasystem. Diese kostengünstige Variante bietet eine ausreichende Tiefengenauigkeit.

3. Ausrichtung der Arbeit

3.1. Ziele dieser Arbeit

Grundsätzliches Ziel dieser Arbeit ist es, ein allgemeines System bereitzustellen, welches robust und mit einer möglichst hohen Genauigkeit die Position, Orientierung und Bewegung von rigiden oder artikulierten Objekten auf Basis der Stereobildsequenzanalyse ermittelt. Außerdem sollen anwendungsspezifische Systemausprägungen verfügbar sein, die nochmals genauer auf die Eigenschaften und Schwierigkeiten in zwei Szenarien eingehen und spezielle Lösungen aufzeigen, wobei auch eine hohe Robustheit gefordert wird, da es sich in beiden Szenarien um Sicherheitssysteme handelt.

Hauptaugenmerk bei der Bereitstellung des allgemeinen Systems bzw. den einzelnen Systemausprägungen für zwei Szenarien liegt auf der Einbringung von Rückkopplungen in die Verarbeitungshierarchie. Hierdurch soll die Genauigkeit und die Robustheit des Systems weiter verbessert werden. Dabei sollen die Ergebnisse eines Verarbeitungsschritts nicht nur den nachfolgenden Schritten zugänglich sein. Auch die vorhergehenden Schritte sollen durch eine Rückkopplung verbessert werden.

Bei der Bewegungsanalyse sollen nicht nur laterale Objektbewegungen möglichst genau ermittelt werden, sondern auch kleine und große Änderungen des Abstands zwischen Objekt und Kamera sollen erfaßt werden. Daher ist es notwendig, alle vier Bilder, die die Stereokamera in zwei Zeitschritten bereitstellt, zu nutzen. Durch umfangreiche Experimente sollen die vorgestellten Systemausprägungen gegeneinander und auch gegenüber Methoden aus dem Stand der Forschung evaluiert werden. Hier soll gezeigt werden, dass in schwierig auszuwertenden Bildsequenzen, deren Auswertung sehr anspruchsvoll ist, das System genau und robust funktioniert. Ground-Truth-Daten, die von einem unabhängigen System ermittelt wurden, sind dafür notwendig und sollen in beiden Szenarien benutzt werden. Abschließend sollen die Schlußfolgerungen, die sich im Rahmen dieser Arbeit ergeben, zusammengefasst werden und als Basis für zukünftige Entwicklungen dienen.

3.2. Überblick

Im ersten Abschnitt der vorliegenden Arbeit wird neben der Einführung noch auf den Stand der Forschung eingegangen, wobei neben allgemeinen Bildverarbeitungstechniken auch auf Verfahren zur Erkennung und Verfolgung von Personen bzw. Fahrzeugen in Bildsequenzen und auf Ansätze der sicheren Mensch-Roboter-Interaktion eingegangen wird. Im anschließenden zweiten Abschnitt wird das vorgestellte System beschrieben. Zunächst wird auf die einzelnen Systemkomponenten eingegangen, darauffolgend wer-

den die einzelnen, anwendungsspezifischen Systemausprägungen vorgestellt. Außerdem werden zwei Ansätze vorgestellt, mit deren Hilfe die Systemrobustheit weiter gesteigert werden kann, indem Rückkopplungen zwischen den einzelnen Verarbeitungsschritten eingebracht werden. Im dritten Abschnitt wird die ausführliche Evaluierung aller vorgestellter Verfahren und Systemausprägungen beschrieben. Dazu wird zunächst das unabhängige System zur Ermittlung der Ground-Truth-Daten erläutert, gefolgt von den Untersuchungen im Produktionsszenario und im Straßenverkehrsszenario. Abschließend werden auch die beiden Rückkopplungsansätze an realen Sequenzen evaluiert. Im letzten Abschnitt wird die Arbeit zusammengefasst und ein Ausblick auf zukünftige Arbeiten gegeben, die noch offene, hier angesprochene Problemstellungen behandeln.

Teil II.

Das System zur Analyse der Position, Orientierung und Bewegung von rigiden und artikulierten Objekten

4. Entwickelte Systemkomponenten

In diesem Kapitel wird das entwickelte Gesamtsystem dargestellt und dazu insbesondere die einzelnen Komponenten des Systems beschrieben. Neben Verfahren aus dem Stand der Forschung kommen hier überwiegend eigene, neu entwickelte Verfahren zum Einsatz. Die Verfahren werden erläutert und es wird aufgezeigt, weshalb sie vorteilhaft gegenüber bekannten Verfahren sind. Dabei wird zunächst kurz die Basis aus der Literatur als Ausgangspunkt für die Neuentwicklungen beschrieben. Anschließend werden die Neuerungen vorgestellt und Eigenschaften erläutert, die eine Verbesserung gegenüber den Methoden aus dem Stand der Forschung sind.

4.1. Grundidee und Systemüberblick

Die vorliegende Arbeit hat das Ziel, ein allgemeingültiges Systemkonzept für die Analyse der Position, Orientierung und Bewegung von rigiden und artikulierten Objekten vorzustellen, welches für verschiedene Szenarien geeignet ist. Als Beispielszenarien werden zum einen die sichere Mensch-Industrieroboter-Interaktion im Produktionsumfeld und zum anderen die Kreuzungsassistenz für den Fahrer im Straßenverkehrsumfeld herangezogen.

Das Gesamtsystem besteht aus den Hauptteilen Szenenflussberechnung, Vorsegmentierung, Pose-Estimation und Bewegungsanalyse (siehe Abb. 4.1). Dabei dient die Szenenflussberechnung zunächst dazu, eine dreidimensionale Oberflächenrekonstruktion der beobachteten Szene durch Ermittlung einer Vielzahl von 3D-Punkten zu erhalten und außerdem die Bewegung der einzelnen Punkte zu bestimmen. Um anschließend die Szene in einzelne Objekte zu gliedern, wird eine Vorsegmentierungsstufe verwendet, die im Wesentlichen aus einem graphenbasierten Clusterverfahren und einer vom Szenario abhängigen Clusterauswahl besteht. Auf Basis des dadurch ermittelten Objektpunkteclusters wird durch eine Pose-Estimation die Pose des Objekts ermittelt und die Bewegung des Objekts auf Basis der Szenenflussinformationen berechnet.

Die beschriebene Verarbeitungskette läuft dabei nicht nur nacheinander ab, sondern beinhaltet mehrere Rückkopplungen. Durch diese Rückkopplungen werden Verarbeitungsergebnisse dazu benutzt, um vorhergehende Systemkomponenten zu optimieren und die Robustheit des Gesamtsystems zu erhöhen. Auf die einzelnen Rückkopplungen wird in Kap. 5 bzw. in Kap. 6 näher eingegangen.

Einer der Vorteile dieses Konzepts ist die Modularität. Es können je nach Problemstellung die einzelnen Komponenten gegen Algorithmen mit gleicher Funktion ausgetauscht werden. Außerdem ist es möglich, verschiedene Algorithmen parallel laufen zu lassen und die Ergebnisse anschließend zu fusionieren, um eine Redundanz in der Verarbeitung zu erhalten.

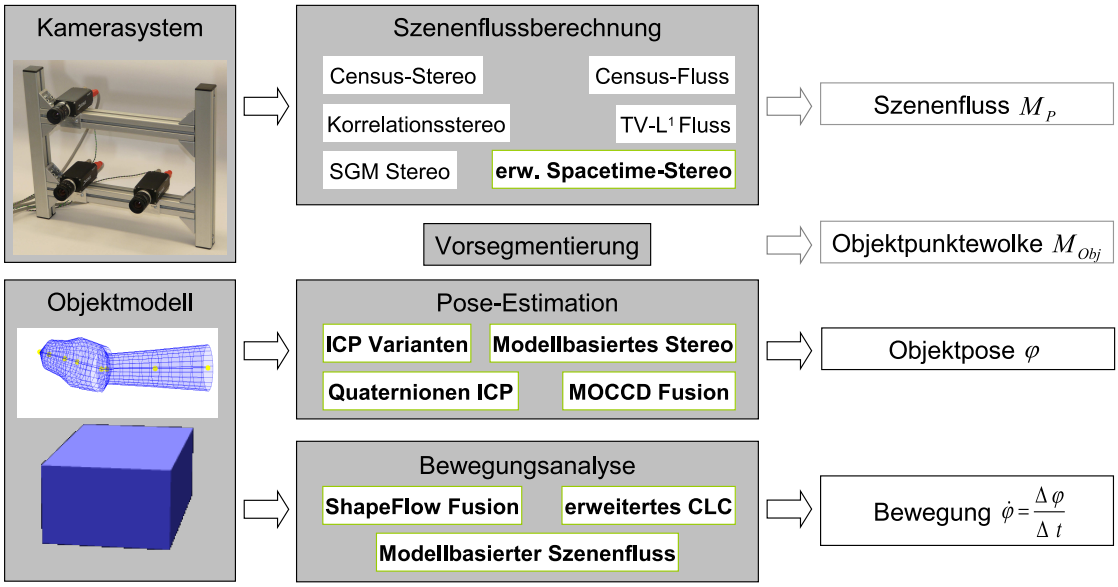


Abbildung 4.1.: Überblick über das Gesamtsystem mit den Hauptteilen Szenenflussberechnung, Vorsegmentierung, Pose-Estimation und Bewegungsanalyse.

Im Folgenden werden zunächst die im Rahmen dieser Arbeit entwickelten Systemkomponenten erläutert. Anschließend werden verschiedene Systemausprägungen beschrieben, wobei diese aus den entwickelten Systemkomponenten bzw. aus bekannten Methoden aus dem Stand der Forschung bestehen. Abschließend werden noch zwei Ansätze vorgestellt, um Modellwissen in einzelne Verarbeitungsschritte rückzukoppeln, wodurch die Systemrobustheit weiter gesteigert wird.

4.2. Berechnung des Szenenflusses

Für die Berechnung des Szenenflusses stehen in der Literatur verschiedene Verfahren zur Verfügung. Wie in Kap. 2.1.4 beschrieben, sind Verfahren, die einen dichten Szenenfluss für die komplette Szene zur Verfügung stellen, derzeit nicht in Echtzeit zu realisieren. Daher beschränkt sich die vorliegende Arbeit auf Methoden zur spärlichen Berechnung dieser Daten, d.h. es wird nicht für jeden einzelnen Bildpunkt der Szenenfluss bestimmt, sondern nur für eine für die Anwendung ausreichende Anzahl von Bildpunkten.

Die vorgestellten Verfahren arbeiten auf Basis von rektifizierten Bildern, sodass die Epipolarlinien des Stereokamerasystems parallel der Bildzeilen der beiden Stereobilder verlaufen (siehe Kap. 2.1.1).

4.2.1. Erweitertes Spacetime-Stereo

Das sogenannte Spacetime-Stereo stellt einen vielversprechenden Ansatz zur spärlichen Szenenflussberechnung dar (Gövert, 2006; Schmidt et al., 2007). Das Verfahren beinhal-

tet in seiner ursprünglichen Form drei grundlegende Verarbeitungsschritte: Suche nach Interestpixeln (Interestoperator), raum-zeitliche Grauwertmodellierung und die Korrespondenzanalyse. Durch den Interestoperator wird nach vertikalen Kanten im Bild gesucht. Diese Kanten werden durch eine sigmoide Modellfunktion approximiert. Abschließend werden diese modellierten Kanten in einer Korrespondenzanalyse verglichen und daraus Szenenflussinformationen berechnet.

Interestoperator: Der Interestoperator besteht aus einem Filter für vertikale Kanten im Bild. Hierzu wird der entsprechende *Sobel-Operator* (Forsyth und Ponce, 2002) benutzt:

$$\mathbf{S_Y} = \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix}. \tag{4.1}$$

Anschließend werden die Kanten zeilenweise betrachtet. Mittels des *Non-Maximum-Suppression-Verfahrens* nach Canny (1986) wird jeweils das Pixel einer Kante mit dem betragsgrößten Gradienten gesucht, welches einen Interestpixel darstellt.

Raum-zeitliche Approximation: Um die Interestpixel während der Korrespondenzanalyse vergleichen und zuordnen zu können, werden die Kanten, die durch die Interestpixel beschrieben werden, mittels einer Modellfunktion approximiert.

Wird eine Kante im Bild abgebildet, entsprechen die Intensitätswerte nicht exakt dem Verlauf der realen Kante, weil dieser während der Aufnahme gemäß der *Point Spread Function* (PSF) des Kamerasystems (siehe Kap. 2.1.1) „verschmiert“ wird. Da dieser Verlauf der Faltung der idealen Kante mit der PSF entspricht, kann die Kante durch eine sigmoide Funktion approximiert werden. Hierzu wurde aufgrund seiner mathematischen Eigenschaften der *Tangens-Hyperbolicus* gewählt (Krüger, 2007). Der Zusammenhang zwischen der Sigmoidfunktion und dem Tangens-Hyperbolicus ist in Gl. 4.2 dargestellt.

$$\mathrm{sig(x)} = \frac{1}{1+\mathrm{e}^{-\mathrm{x}}}\ ,\ \tanh(\mathrm{x}) = \frac{2}{\mathrm{e}^{2\mathrm{x}}+1}\ ,\ \tanh\left(\frac{\mathrm{x}}{2}\right) = 2\,\mathrm{sig(x)} - 1 \tag{4.2}$$

Durch diese Approximation kann eine Kante in einer Zeile durch Gleichung 4.3 beschrieben werden.

$$I_s(\mathbf{p}, u) = p_1 \cdot \tanh(p_2 \cdot u + p_3) + p_4 \tag{4.3}$$

Dabei stellt $I_s(\mathbf{p}, u)$ den Grauwert in Abhängigkeit von der Position u innerhalb der Zeile und des Parametervektors $\mathbf{p} = [p_1, p_2, p_3, p_4]$ dar (siehe Abb. 4.2). Die einzelnen Parameter beeinflussen hierbei folgende Eigenschaften des Funktionsverlaufs:

- p_1: Amplitude der Funktion (Kontrast)

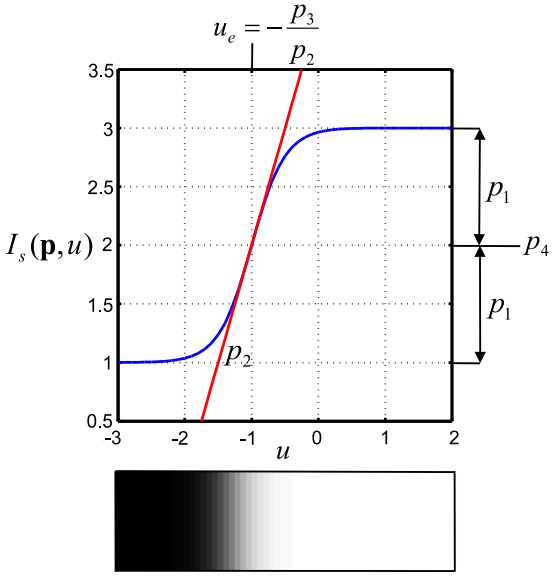


Abbildung 4.2.: Approximation einer Kante in einer Zeile. Beispielhaft ist der Verlauf des Tangens-Hyperbolicus mit dem Parametervektor $\mathbf{p} = [1, 2, 2, 2]$ dargestellt. Es wird der Einfluss der einzelnen Parameter verdeutlicht.

- p_2: Steilheit der Kante (Schärfe), fallende oder steigende Kante entsprechend dem Vorzeichen
- p_3: Maß für die Position des betragsgrößten Gradienten $u_e = -\frac{p_3}{p_2}$ (siehe Gl. 4.10)
- p_4: Intensität an der Stelle u_e

Durch diese Approximation kann die Stelle u_e subpixelgenau bestimmt werden, was bei der Korrespondenzanalyse zu subpixelgenauen Disparitäten führt. Der beschriebene Tangens-Hyperbolicus als Approximation einer Kante innerhalb einer Zeile kann auf eine raum-zeitliche *Region-Of-Interest (ROI)* um einen Interestpixel erweitert werden. Da sich der Kantenverlauf in dieser ROI sowohl über die Bildzeilen als auch über die Zeitschritte verändern kann, werden die skalaren Parameter p_1 bis p_4 zu Parameter-Funktionen, welche abhängig von der Bildzeile v und dem Zeitschritt t sind:

$$\mathbf{p} = [p_1(v,t), p_2(v,t), p_3(v,t), p_4(v,t)]. \tag{4.4}$$

Somit ergibt sich folgende Gleichung für die Grauwertmodellierung:

$$I_s(\mathbf{p}, u, v, t) = p_1(v,t) \cdot \tanh[p_2(v,t) \cdot u + p_3(v,t)] + p_4(v,t). \tag{4.5}$$

Durch diese Art der Modellierung lässt sich die Schrägheit der Kante (Kantenverlauf über die Bildzeilen) und die Bewegung der Kante innnerhalb der raum-zeitlichen ROI

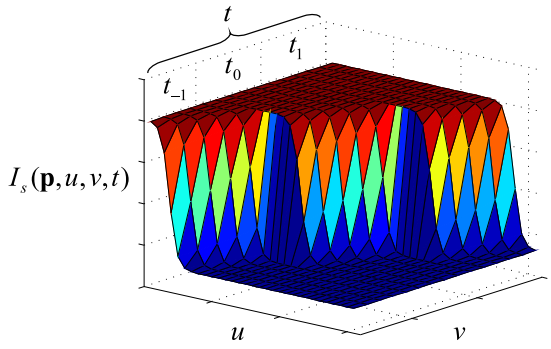


Abbildung 4.3.: Approximation einer Kante für einen raum-zeitlichen Bereich. Es ist der synthetisierte Verlauf $I_s(\mathbf{p}, u, v, t)$ entsprechend des gefitteten Tangens-Hyperbolicus über einen räumlichen Bereich u, v und drei Zeitschritte t_{-1}, t_0 und t_1 (raum-zeitliche ROI) dargestellt.

beschreiben (siehe Abb. 4.3). Die Größe der raum-zeitlichen ROI ist durch die zu betrachtenden Bildzeilen und Zeitschritte definiert. Außerdem richtet sich die Breite der ROI nach dem horizontalen Kantenverlauf bzw. der PSF des Kamerasystems, sowie nach der Objektgeschwindigkeit.

Die Funktionen $p_1(v,t), p_2(v,t), p_3(v,t)$ und $p_4(v,t)$ sind in den hier betrachteten Szenarien immer Polynome maximal vom Grad 2 in v und t, deren Parameter durch eine nichtlineare Optimierung ermittelt werden. Durch diese Polynome lassen sich die Veränderungen der Parameter p_1 bis p_4 innerhalb der raum-zeitlichen ROI modellieren. Die Parameter werde so optimiert, dass der Unterschied zwischen dem synthetischen und dem realen Kantenverlauf so gering wie möglich ist. Ist der Restfehler nach der Optimierung zu hoch, d.h. sind sich die synthetische und die reale Kante nicht ähnlich genug, wird der entsprechende Interestpixel verworfen und nicht weiter betrachtet, was beispielsweise an Ecken im Bild passiert, wenn die Modellfunktion nicht den Bildinhalt repräsentieren kann. Um den Rechenaufwand für diese Optimierung zu verringern, werden Annahmen über den Verlauf einer Kante innerhalb einer ROI gemacht. Zum Beispiel wird angenommen, dass der maximale Grauwert der Kante über die Zeitschritte konstant bleibt, wodurch die Polynome unvollständig werden und sich die Anzahl der zu schätzenden Parameter verringert. In Hinblick auf die Echtzeitfähigkeit ist der Fit des Tangens-Hyperbolicus dennoch zu rechenaufwendig, weshalb ein Ansatz zur Linearisierung der Optimierung entwickelt wurde (Gövert, 2006; Schmidt et al., 2007). Dadurch steht eine geschlossene Lösung für die Grauwertmodellierung zur Verfügung, welche jedoch qualitativ schlechtere Ergebnisse als die nichtlineare Variante liefert.

Korrespondenzanalyse: Um korrespondierende Punkte auf einer Epipolarlinie zu finden, müssen die Interestpixel des linken Bildes ($l_i = l_1, ..., l_n$) mit denen im rechten Bild ($r_j = r_1, ..., r_m$) verglichen werden. Ergebnis dieses Vergleichs ist ein Wert für die Ähnlichkeit, welcher abhängig vom Gewählten Vergleichsmaß ist. Anhand diese

Ähnlichkeitswertes soll entschieden werden, welche Punkte korrespondieren.

Gövert (2006) hat gezeigt, dass der Vergleich der Polynom-Parameter des Tangens-Hyperbolicus-Ansatzes keine zufriedenstellenden Ergebnisse liefert. Daher werden je Interestpixel synthetische Intensitätswerte für die raum-zeitliche ROI entsprechend der gefitteten Modellfunktion berechnet. Diese synthetischen Intensitätswerte $I_s(\mathbf{p}, u, v, t)$ stellen die zu vergleichenden Merkmale dar. Zeilenweise werden diese Intensitätswerte für jedes Interestpixels im linken Bild ($I_s(\mathbf{p_{l_i}}, u, v, t) = \mathbf{L_i} = \mathbf{L_1}, ..., \mathbf{L_n}$) mit denen jedes Interestpixels im rechten Bild ($I_s(\mathbf{p_{r_j}}, u, v, t) = \mathbf{R_j} = \mathbf{R_1}, ..., \mathbf{R_m}$) verglichen ($u, v, t$ laufen entsprechend des Mittelpunktes und der Größe einer ROI). Es gibt verschiedene Ähnlichkeitsmaße für diesen Vergleich (siehe Abschnitt 2.1.2). Die besten Resultate werden laut Gövert (2006) mit der SSD erzielt (Gleichung 4.6), d.h. es werden die $SSD(\mathbf{L_i}, \mathbf{R_j})$ für alle Kombinationen von Interestpixel-ROIs $\mathbf{L_i}$ und $\mathbf{R_j}$ berechnet. Ist dieser Wert $SSD(\mathbf{L_i}, \mathbf{R_j})$ klein, sind die synthetischen Intensitätswerte in den ROIs $\mathbf{L_i}$ und $\mathbf{R_j}$ ähnlich, was bedeuten kann, dass die zugehörigen Interestpixel l_i und r_j den gleichen 3D-Punkt beschreiben.

$$SSD(\mathbf{L_i}, \mathbf{R_j}) = \sum_{\tilde{u}} \sum_{\tilde{v}} \sum_{\tilde{t}} (\mathbf{L_i}(\tilde{u}, \tilde{v}, \tilde{t}) - \mathbf{R_j}(\tilde{u}, \tilde{v}, \tilde{t}))^2 \tag{4.6}$$

Die $SSD(\mathbf{L_i}, \mathbf{R_j})$ der einzelnen Vergleiche der synthetischen Intensitätswerte innerhalb der Interestpixel-ROIs werden in einer Matrix $\mathbf{E_{SSD}}$ (4.7) gespeichert.

$$\mathbf{E_{SSD}} = \begin{pmatrix} SSD(\mathbf{L_1}, \mathbf{R_1}) & SSD(\mathbf{L_1}, \mathbf{R_2}) & \cdots & SSD(\mathbf{L_1}, \mathbf{R_m}) \\ SSD(\mathbf{L_2}, \mathbf{R_1}) & SSD(\mathbf{L_2}, \mathbf{R_2}) & \cdots & SSD(\mathbf{L_2}, \mathbf{R_m}) \\ \vdots & \vdots & \ddots & \vdots \\ SSD(\mathbf{L_n}, \mathbf{R_1}) & SSD(\mathbf{L_n}, \mathbf{R_2}) & \cdots & SSD(\mathbf{L_n}, \mathbf{R_m}) \end{pmatrix} \tag{4.7}$$

Für jedes Element in $\mathbf{E_{SSD}}$, d.h. jede mögliche Korrespondenz, gibt es einen Disparitätswert. Diese Disparitätswerte sind analog in einer Matrix $\mathbf{D}$ (4.8) abgelegt.

$$\mathbf{D} = \begin{pmatrix} d(l_1, r_1) & d(l_1, r_2) & \cdots & d(l_1, r_m) \\ d(l_2, r_1) & d(l_2, r_2) & \cdots & d(l_2, r_m) \\ \vdots & \vdots & \ddots & \vdots \\ d(l_n, r_1) & d(l_n, r_2) & \cdots & d(l_n, r_m) \end{pmatrix} \tag{4.8}$$

Aus allen möglichen Korrespondenzen in diesen Matrizen $\mathbf{E_{SSD}}$ und $\mathbf{D}$ sollen die richtigen Zuordnungen mit ihren korrekten Disparitätswerten ermittelt werden. Hierzu werden verschiedene Constraints angewendet und Annahmen gemacht:

- Die SSD muss kleiner als eine Schwelle θ_{SSD} sein (Compatibility Constraint): $SSD(\mathbf{L_i}, \mathbf{R_j}) < t_{SSD}$.
- Es existieren keine negativen Disparitätswerte (abgeleitet aus Standardstereogeometrie; Epipolar Constraint): $d(l_i, r_j) \geq 0$.

- Einschränkung des Disparitätsbereichs entsprechend der Applikation (Disparity Limit Constraint): $d_{min} \leq d(l_i, r_j) \leq d_{max}$.
- Ein Interestpixel kann maximal an einer Korrespondenz beteiligt sein (Uniqueness Constraint).

Die einzelnen Constraints werden in Kap. 2.1.2 näher erläutert.
Durch diese Bedingungen werden $\mathbf{D}$ und $\mathbf{E_{SSD}}$ zu oberen Dreiecksmatrizen. Zudem werden Matrixelemente, welche durch eine der oberen Annahmen entfernt werden, eindeutig gekennzeichnet. Ist weiterhin eine eindeutige Zuordnung nicht möglich, d.h. es gibt mehrere mögliche Korrespondenzen, die o.g. Bedingungen erfüllen, wird die mit der kleinsten SSD gewählt.

Schließlich kann die Zuordnung zweier Interestpixel zu einer Korrespondenz in einer Matrix $\mathbf{C}$ (Gleichung 4.9) codiert werden.

$$\mathbf{C} = \begin{pmatrix} c(l_1, r_1) & c(l_1, r_2) & \cdots & c(l_1, r_m) \\ 0 & c(l_2, r_2) & \cdots & c(l_2, r_m) \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & c(l_n, r_m) \end{pmatrix} \tag{4.9}$$

$$\text{mit } c(l_i, r_j) = \begin{cases} 1, & \text{wenn } l_i \text{ und } r_j \text{ korrespondierend} \\ 0, & \text{sonst} \end{cases}$$

Das Ergebnis der Korrespondenzanalyse sind die Korrespondenzen mit ihren entsprechenden Disparitätswerten und den Parametern der Modellfunktionen beider approximierter Kanten.

Berechnung Szenenfluss: Die Position des betragsgrößten Intensitätsgradient in horizontaler Richtung wird nach der Grauwertmodellierung subpixelgenau beschrieben durch:

$$u_e(v, t) = -\frac{p_3(v, t)}{p_2(v, t)}. \tag{4.10}$$

Der vertikale Verlauf δ des Intensitätsgradienten innerhalb der ROI und dessen Geschwindigkeit μ entlang der Epipolarlinie sind gegeben durch:

$$\delta = \left.\frac{\partial u_e}{\partial v}\right|_{v_c, t_c} \text{ und } \mu = \left.\frac{\partial u_e}{\partial t}\right|_{v_c, t_c}, \tag{4.11}$$

wobei der Index c das Zentrum der raum-zeitlichen ROI beschreibt.

Die subpixelgenaue Disparität ist beschrieben durch:

$$d = [u_i^l + u_e^l(v_c, t_c)] - [u_i^r + u_e^r(v_c, t_c)]\,. \tag{4.12}$$

Die Parameter u_i^l und u_i^r bezeichnen die ganzzahlige horizontale Pixelkoordinate des

linken bzw. des rechten Interestpixels. Die Parameter $u_e^l(v_c, t_c)$ und $u_e^r(v_c, t_c)$ beschreiben die subpixelgenaue Position der Kante relativ zum Interestpixel. Insgesamt wird die subpixelgenaue Position der Kante im Bild durch $u_i^l + u_e^l(v_c, t_c)$ bzw. $u_i^r + u_e^r(v_c, t_c)$ beschrieben.

Die Tiefengeschwindigkeit $\dot{z}$ entlang der z-Achse ist in metrischen Einheiten definiert durch (Wöhler, 2009):

$$\dot{z} = \frac{\partial z}{\partial t} = -\frac{bf\left(\mu^l - \mu^r\right)}{d^2}. \tag{4.13}$$

Die laterale Geschwindigkeitskomponente $\dot{x}$ entlang der Epipolarlinie in metrischen Einheiten wird berechnet durch (Wöhler, 2009):

$$\dot{x} = \frac{\partial x}{\partial t} = b\frac{\mu^r d - \frac{1}{2}\left(u_e^l + u_e^r\right)\frac{\partial d}{\partial t}}{d^2}\ , \tag{4.14}$$

mit

$$\frac{\partial d}{\partial t} = \mu^l - \mu^r\ . \tag{4.15}$$

Die vertikale Geschwindigkeitskomponente $\dot{y} = \partial y/\partial t$ kann wegen des Aperturproblems nicht durch das Spacetime-Stereo ermittelt werden, da nur vertikale Kanten in den Stereobildpaaren erkannt und modelliert werden und diese lediglich eine Aussage über die horizontale Geschwindigkeitskomponente zulassen. Sind jedoch Aufnahmen eines trinokularen Kamerasystems verfügbar, welche zu jedem Zeitschritt ein horizontales und ein vertikales Stereobildpaar beinhalten, so können einmal horizontale und einmal vertikale Kanten zur Berechnungen von Stereokorrespondenzen und Bewegungsinformationen verwendet werden. Abb. 4.4 zeigt die berechneten 3D-Punkte und die Geschwindigkeitskomponenten für Aufnahmen eines trinokularen Kamerasystems.

Abschließend erhält man für die beobachtete Szene eine Punktewolke M_p, wobei für jeden gefundenen 3D-Punkte $w_i \in M_p$ die räumliche Koordinaten x_i, y_i, z_i, sowie die Tiefengeschwindigkeit $\dot{z}_i$ und die laterale Geschwindigkeitskomponente $\dot{x}_i$ verfügbar sind.

Bewertung des Spacetime-Stereos: Die Vorteile des Spacetime-Stereos nach Gövert (2006) liegen in den subpixelgenau bestimmbaren Kantenmittelpunkten und den daraus ermittelten Disparitätswerten. Dies führt zu einer signifikanten Erhöhung der Tiefenauflösung im Vergleich zu pixelgenauen Verfahren (siehe Kap. 2.1.2). Weiter können aufgrund der Betrachtung einer Kante über mehrere Zeitschritte Geschwindigkeitsinformationen ermittelt werden, wodurch z.B. eine Hintergrundeliminierung bis hin zu einer Bewegungsanalyse bzw. einem Tracking möglich sind. Somit stehen spärliche, unvollständige Szenenflussdaten zur Verfügung, die für die vorgesehenen Szenarien zum Einsatz kommen. Die Daten sind spärlich, da nicht zu jedem Pixeln im Bild auch ein Tiefenwert zur Verfügung steht und sie sind unvollständig, da die vertikale Geschwindigkeitskomponente $\dot{y}$ mithilfe des Spacetime-Stereos nicht ermittelt werden kann.

Ein Nachteil des von Gövert (2006) vorgestellten Verfahrens ist der hohe Rechenaufwand, speziell für den Fit der Parameter des Tangens-Hyperbolicus'. Durch die ange-

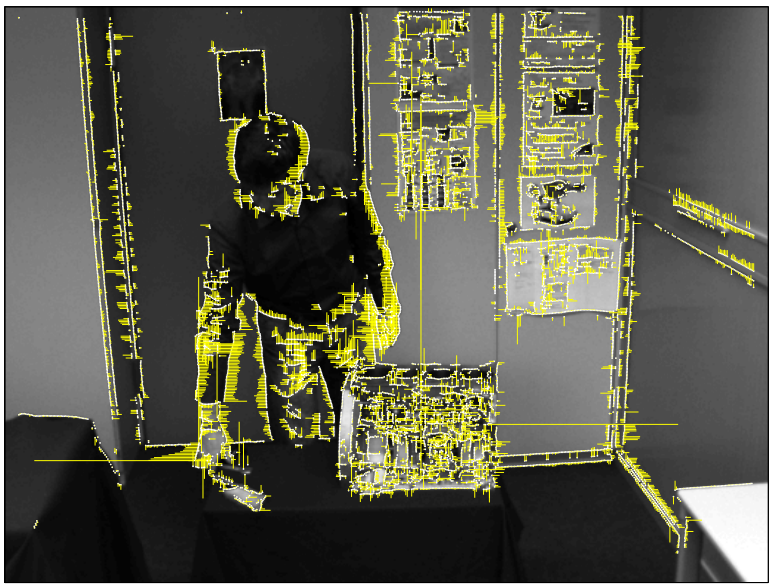


Abbildung 4.4.: Spacetime-Stereo Ergebnisbeispiel mit 3D-Punkten (weiss) und Epipolargeschwindigkeiten (gelb). Hier wurden aus trinokularen Bilddaten vertikal und horizontal Kanten extrahiert, Korrespondenzen etabliert und anschließend kombiniert.

sprochene Linearisierung der Optimierung kann diese in Grenzen gehalten werden, was jedoch Einbuße in der Genauigkeit zur Folge hat.

Außerdem kommt es zu Problemen wenn mehrere Kantenverläufe ineinander übergehen. Beispielsweise entsteht im Produktionsumfeld im Abbild des Arms eines Arbeiters eine sogenannte Shadingkante, d.h. der Intensitätsverlauf auf dem Arm ändert sich orthogonal zur Armachse und führt zu einer zusätzlichen Kante im Bild. Diese wiederum verschmiert mit der Objektkontur, was letztendlich zu einem Grauwertverlauf führt, der durch die beschriebene Modellfunktion nicht zu approximieren ist (siehe Abb. 4.8). Folglich werden nur wenige 3D-Punkte auf der Kontur des Arms des Arbeiters gefunden.

Die Objektgeschwindigkeit im Bild ist bei dem Verfahren beschränkt, da durch die nichtlineare Optimierung der Modellfunktion in der ROI immer nur die nächste Kante im nächsten Zeitschritt gefunden wird. Dadurch entsteht ein Konvergenzradius für reale Szenen von ca. sechs Pixeln horizontaler Verschiebung pro Zeitschritt. Bei schnelleren Bewegungen, wie sie beispielsweise im Straßenverkehrsumfeld oft auftreten, wird keine zeitliche Korrespondenz gefunden, d.h. schnelle Objekte verschwinden aus der Punktewolke.

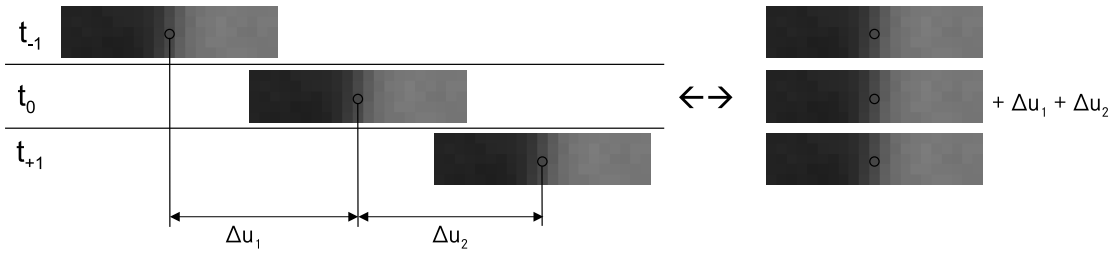


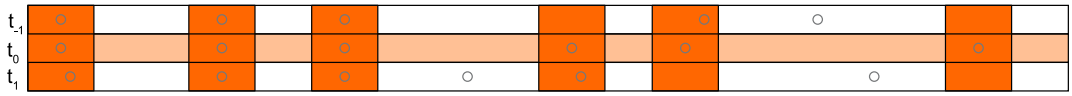
Abbildung 4.5.: Zur Erhöhung des Konvergenzradius der Bewegungsschätzung stehen im erweiterten Spacetime-Stereo zu jeder ROI Versatzinformationen $\Delta\mathbf{u}_{\mathrm{ROI}} = [\Delta u_1, \Delta u_2]^T$ zur Verfügung.

Neuentwickelte Erweiterungen des Spacetime-Stereos

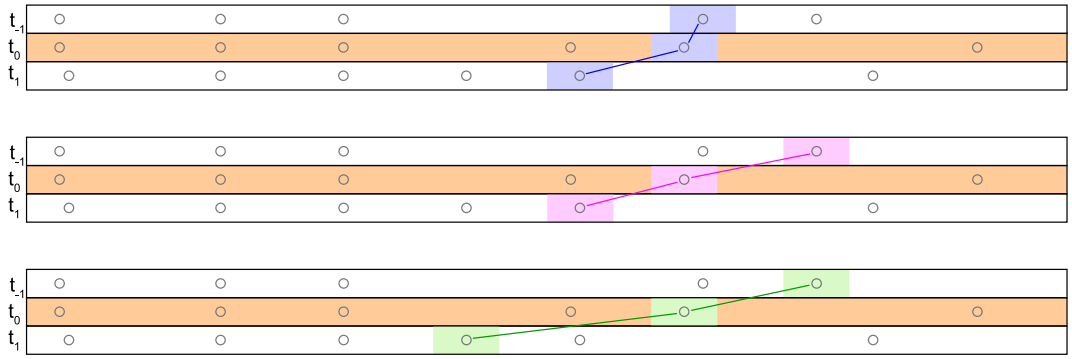
Vergrößerung des Konvergenzradius der Geschwindigkeitsschätzung: Im Rahmen dieser Arbeit wurde das verfügbare Spacetime-Stereo-Verfahren von Gövert (2006) erweitert, um auch schnelle Objektbewegungen im Bild detektieren zu können. Wie bereits beschrieben, ist der Konvergenzradius für die Geschwindigkeitsschätzung im Spacetime-Stereo verhältnismäßig klein, was dazu führt, dass schnelle Objekte aus der für die Szene ermittelten Punktewolke M_p verschwinden ($w_i \in M_p$; $w_i = [x_i, y_i, z_i, \dot{x}_i, \dot{z}_i]^T$). Kanten mit einem großen Versatz in der Bildebene sind in zwei aufeinanderfolgenden Zeitschritten nicht mehr in der raum-zeitlichen ROI enthalten, da die Bewegung im Bild die Breite der ROI überschreitet. Wird die Breite der schnellen Bewegung angepasst, werden mehrere Kanten in der ROI abgebildet und eine richtige Zuordnung ist nicht mehr möglich.

Zur Lösung dieses Problems wird eine neue Art von ROI definiert. In einem einzelnen Zeitschritt werden zwar Bereiche der bisherigen Größe abgedeckt, d.h. angepasst nur an die Breite der Kantenverläufe bzw. an die Geschwindigkeit „langsamer“ Objekte. Im Gegensatz zu den bisherigen ROIs aus dem System von Gövert (2006), welche Bildausschnitte an der gleichen Position in allen Zeitschritten beinhalten, kann sich die Position nun über die Zeit ändern, die Bildausschnitte sind also zeitlich zueinander versetzt. Daher beinhalten die neuen ROIs zusätzlich zu den Grauwertinformationen die Versatzinformation $\Delta\mathbf{u}_{\mathrm{ROI}} = [\Delta u_1, \Delta u_2]^T$, welche die Positionsänderung in den aufeinander folgenden Zeitschritten darstellt (siehe Abbildung 4.5).

Zu einem Bildausschnitt aus dem aktuellen Zeitschritt, zu dem durch die Grauwertmodellierung im verfügbaren System von Gövert (2006) noch keine korrespondierende Kante in den anderen Zeitschritten gefunden wurde, werden passende Bildausschnitte aus den beiden anderen Zeitschritten (t_{-1} und t_{+1}) an den Stellen gesucht, an denen Interestpixel detektiert wurden, d.h. es werden Kombinationen von zeitlich versetzten Bildausschnitten untersucht. Durch eine Annahme über die maximale Geschwindigkeit im Bild, wird die Anzahl an Kombination von vornherein sinnvoll beschränkt. Zur Bewertung einer Kombination dient ein Vergleichmaß, in diesem Fall die SSD, für den Bildinhalt um die Interestpixel herum, was zur Untersuchung aller Interestpixelkombi-



(a) ROIs ohne zeitlichen Versatz



(b) ROIs mit Versatzinformation

Abbildung 4.6.: Im verfügbaren System (a) wurden zu jeden Interestpixel eine fest definierte ROI ausgeschnitten. Bei dem neuen, erweiterten System (b) werden möglich Kombinationen von Interestpixel untersucht.

nationen dient. Definieren zwei Interestpixel dieselbe Kante in verschiedenen Zeitschritten, so führt der Vergleich der Bildinhalte zu einem kleinen SSD-Wert. Sind passende Kombinationen über alle drei Zeitschritte gefunden, wird abschließend auch hier die Modellfunktion angepasst, wobei die Versatzinformation stets beachtet werden muss. Gibt es mehrere zeitliche Korrespondenzen, wird die Kombination ausgewählt, welche bei der Modellanpassung die beste Ähnlichkeit zwischen den Grauwerten in den Bildern $I(u, v, t)$ und den modellierten Grauwerten $I_s(\mathbf{p}, u, v, t)$ liefert (siehe Gl. 4.5). Abb. 4.6 zeigt die zeitliche Korrespondenzsuche im neuen System. Der Berechnungsaufwand für das Verfahren steigt durch die Erweiterung des Konvergenzradius nur leicht, da die Analyse für schnelle Kanten nur dann verwendet wird, wenn über den ursprünglichen Ansatz keine zeitlichen Korrespondenzen gefunden wurden.

Auch bei der abschließenden Berechnung der Szenenflussparameter muss die Versatzinformation zwischen den Interestpixeln berücksichtigt werden, um eine korrekte Berechnung von Epipolargeschwindigkeit und Tiefengeschwindigkeit durchführen zu können.

Abb. 4.7 zeigt das Ergebnis der Erweiterung im neuen System: Kanten, die sich schnell in der Szene bewegen und vorher in der Punktewolke M_p der Szene fehlten, sind nun vorhanden und werden richtig berechnet.

Dynamische Maskierung des raum-zeitlichen Kantenverlaufs: Ein weiteres, bereits angesprochenes Problem, sind ineinander übergehende Kantenverläufe. Beispiel hierfür ist Abbild des Unterarms von Personen (siehe Abb. 4.8). Hier gehen die Kantenverläufe der Konturen in die Shadingkante auf dem Unterarm über. Die Kantenverläufe sind nicht

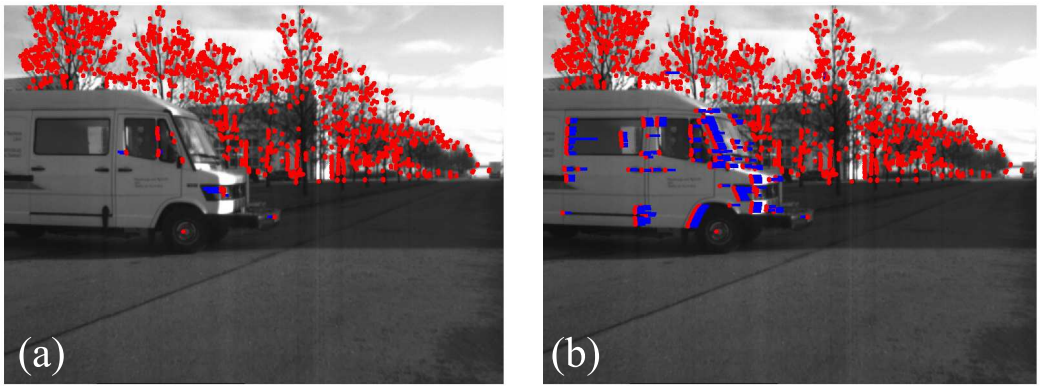


Abbildung 4.7.: Vor der Erweiterung (a) fehlten sich schnell bewegenden Kanten in der Punktewolke. Nach der Erweiterung (b) werden diese schnellen Kanten richtig repräsentiert.

mehr gegeneinander abzugrenzen.

Die bisherige Modellfunktion bedingt genau eine Kante innerhalb der ROI, die möglichst mit einem konstanten Grauwert beginnt und auch wieder endet. Ist dies nicht der Fall, kommt es zu ungenügenden Modellanpassungen (siehe Abb. 4.9a), was über den Restfehler der Modellanpassung erkannt wird und zur Löschung des Pixels als Kandidat für einen Interestpixels führt.

Durch eine Maskierung des Kantenverlaufs könnte dieses Problem behoben werden, da so nur die eigentliche Kante betrachtet wird und Randeffekte ausgeblendet werden. Die Maskierung ist dabei von der Breite des horizontalen Kantenverlaufs, von der Schrägheit δ der Kante in vertikaler Richtung und von der Geschwindigkeit μ der Kante innerhalb der ROI abhängig. Die Parameter δ und μ sind jedoch initial nicht bekannt. D.h. auch wenn die PSF des Kamerasystems bekannt ist, wird die Schrägheit der Kante δ und die Geschwindigkeit μ erst durch die Grauwertmodellierung ermittelt. Daher wird eine Maskierung durch die Gewichtungsfunktion w in die Fehlerfunktion der nichtlinearen Optimierung zur Ermittlung der Parameterfunktionen p_1 bis p_4 mit eingebracht, um so gleichzeitig die passenden Kantenparameter zusammen mit der passenden Maskierung zu erhalten.

Die Fehlerfunktion $e_{\tanh}$ im verfügbaren System von Gövert (2006) für den nichtlinearen Anpassung der Modellfunktion I_s an die Grauwertkante im Bild I lautete (siehe Gl. 4.5):

$$e_{\tanh_1} = I(u,v,t) - I_s(\mathbf{p},u,v,t) \tag{4.16}$$

$$= I(u,v,t) - [p_1(v,t)\cdot\tanh[p_2(v,t)\cdot u + p_3(v,t)] + p_4(v,t)]\,. \tag{4.17}$$

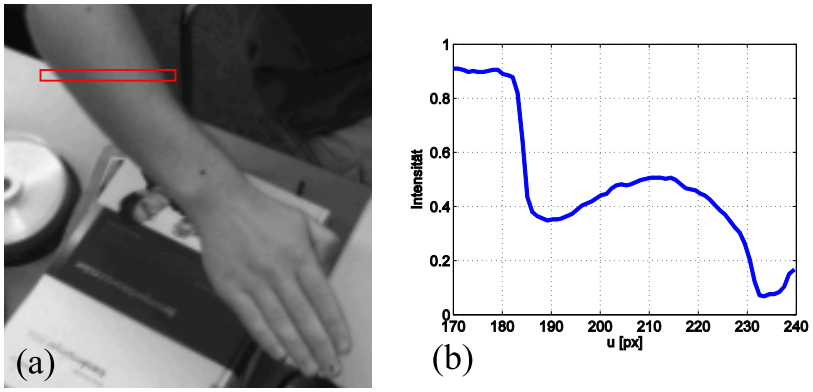


Abbildung 4.8.: Beispiel für das Shading auf dem menschlichen Unterarm (a). Im Bild zeigt sich der charakteristische Grauwertverlauf (b).

Mithilfe der Gewichtungsfunktion $w(\mathbf{p}, u, v, t)$ lässt sich im neuen System der Einfluss der Fehler am Rand unter Berücksichtigung des Kantenverlaufs abschwächen (siehe Gl. 4.18).

$$e_{\tanh_2} = w(\mathbf{p}, u, v, t)\,[I(u, v, t) - I_s(\mathbf{p}, u, v, t)] \tag{4.18}$$

Die Gewichtungsfunktion wird so gewählt, dass an der Stelle des betragsgrößten Gradienten u_e, welche abhängig von der Zeile und dem Zeitschritt innerhalb der ROI ist, das Gewicht 1 beträgt und dann ein stetiger Verlauf zum Rand hin auf 0 erfolgt (siehe Abb. 4.9b), wobei die Stetigkeit wichtig für die Optimierung ist. Die Gewichtungsfunktion ist definiert als:

$$w(\mathbf{p}, u, v, t) = \frac{1}{1 + e^{f_w(u_{dist}(\mathbf{p},u,v,t) - u_{rad})}} \tag{4.19}$$

mit

$$u_{dist}(\mathbf{p}, u, v, t) = |(u - u_e|_{v,t})|. \tag{4.20}$$

Der Parameter u_{rad} korrespondiert dabei mit der Breite der Gewichtsfunktion w und sollte ungefähr dem Radius der PSF der Kamera entsprechen. Der Faktor f_w beeinflusst direkt den Verlauf der Gewichtsfunktion, wobei der Wert ca. bei der Hälfte des PSF-Radius liegen sollte.

Durch diese Erweiterung können nun auch Kanten modelliert werden, die nicht mit einem konstanten Grauwert beginnen und enden (siehe Abb. 4.9c). Experimente haben gezeigt, dass durch diese dynamische Maskierung bei gleichem Berechnungsaufwand das Konvergenzverhalten verbessert wird. Die Qualität der Grauwertmodellierung erhöht sich außerdem, was sich an den Restfehlern der Anpassung zeigt.

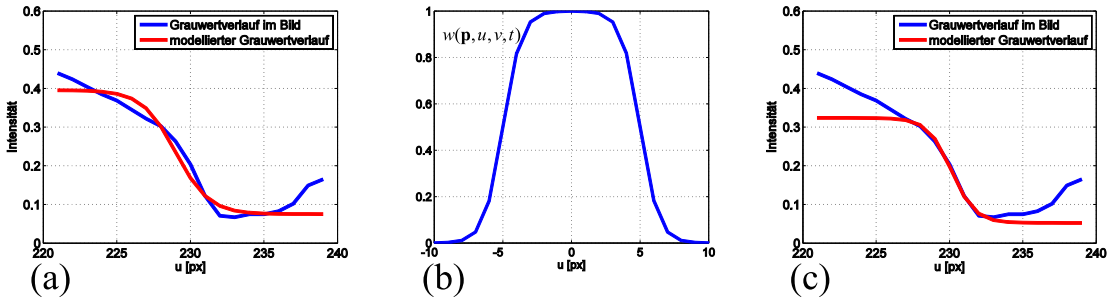


Abbildung 4.9.: Beispiel für die Grauwertmodellierung ohne dynamische Maskierung (a) und mit Maskierung (c). Die Gewichtungsfunktion ist beispielhaft dargestellt (b).

4.2.2. Kombination von 3D-Punktewolke und Verschiebungsvektorfeld

Neben den dichten Szenenflussverfahren, welche einen großen Rechenaufwand bedeuten, und dem beschriebenen Spacetime-Stereo, ist es auch möglich, eine Kombination aus 3D-Punkten, die von einem Stereobildverarbeitungsalgorithmus berechnet wurden, und einem Verschiebungsvektorfeld als unvollständige Szenenrepräsentation zu benutzen. Durch diese Kombination bleibt die Tiefengeschwindigkeit $\dot{z}$, d.h. die Geschwindigkeitskomponente entlang der z-Achse, jedoch unbestimmt. Vorteilhaft ist jedoch der recht geringe Berechnungsaufwand. In der Literatur sind einige Verfahren, sowohl zur Berechnung des optischen Flusses (vgl. Kap. 2.1.3), als auch zur Stereobildverarbeitung (vgl. Kap. 2.1.2) bekannt, die in Echtzeit zu berechnen sind, d.h. im Bezug auf die angedachten Szenarien mehrfach innerhalb einer Sekunde.

In der vorliegenden Arbeit werden verschiedene Kombinationen benutzt:

- **Census-Stereo + Census-Fluss:** Pixelgenaue Korrespondenzbestimmung auf Basis der Census-Transformation, sowohl für Stereo als auch für optischen Fluss (Stein, 2004)
- **Korrelationsstereo + Census-Fluss:** Subpixelgenaue Stereokorrespondenzanalyse auf Blockmatchingbasis (Franke und Joos, 2000) kombiniert mit der pixelgenauen optischen Fluss-Berechnung (Stein, 2004)
- **Korrelationsstereo + TV-L^1 Fluss:** Subpixelgenaues Blockmatching (Franke und Joos, 2000) kombiniert mit dichtem, subpixelgenauem Fluss (Zach et al., 2007)

Census-Stereo + Census-Fluss: Der Vorteil bei dieser Methode liegt in der Berechnungsdauer. Die Korrespondenzsuche über die Hashtabelle (siehe Abschnitt 2.1.3) ist sehr effizient zu berechnen. Allerdings sind die etablierten Korrespondenzen nur pixelgenau, was zu einer Diskretisierung der Tiefenwerte führt.

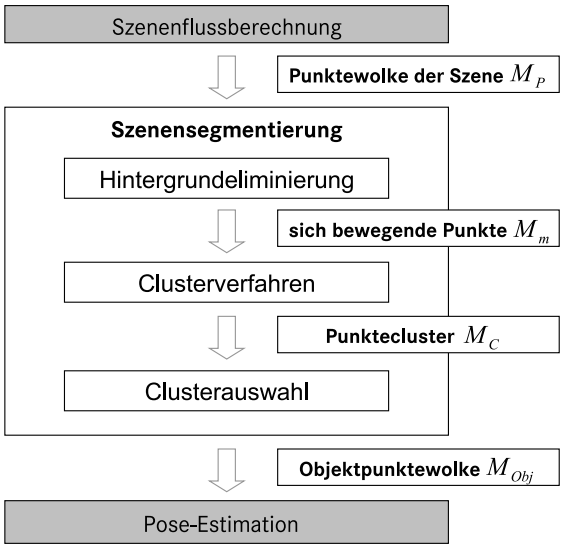


Abbildung 4.10.: Überblick über die Vorsegmentierung.

Korrelationsstereo + Census-Fluss: Durch den Austausch des Census-Stereos durch das Korrelationsstereo werden subpixelgenaue Disparitäten bereitgestellt und die Diskretisierung der Tiefenschätzung aufgehoben.

Korrelationsstereo + TV-L^1 Fluss: Nachteilig bei den beiden vorherigen Kombinationen ist die Bedingung, dass durch den Census-Fluss nur dort Flussvektoren etabliert werden können, wo Ecken im Bild vorhanden sind. Bei der Verfolgung der Hand-Unterarm-Region einer Person sind nur sehr wenige Ecken im Bild verfügbar. Daher ist die Verwendung eines dichten Flussverfahrens, wie hier der TV-L^1 Fluss, sinnvoll.

4.3. Vorsegmentierung

Um die einzelnen Objekte in der Szene voneinander zu trennen, sowie statische von sich bewegenden Elementen zu trennen, wird eine Segmentierungsstufe benötigt. Dabei wird zunächst eine Hintergrundeliminierung durchgeführt, gefolgt von einer Clusteranalyse und abschließend einer Clusterauswahl, welche das Cluster des interessierenden Objekts auswählt (siehe Abb. 4.10).

4.3.1. Hintergrundeliminierung

Liegen für die beobachtete Szene Szenenflussinformationen vor, ist eine Hintergrundeliminierung unkompliziert und effektiv. Durch eine definierte Geschwindigkeitsschwelle θ_{BS} können sich bewegende von unbewegten Punkten getrennt werden. Dabei ist der Parameter θ_{BS} entsprechende des Rauschens des Algorithmus zur optischen Fluss-Berechnung etwas höher als der Rauschanteil der Geschwindigkeitskomponenten der Sze-

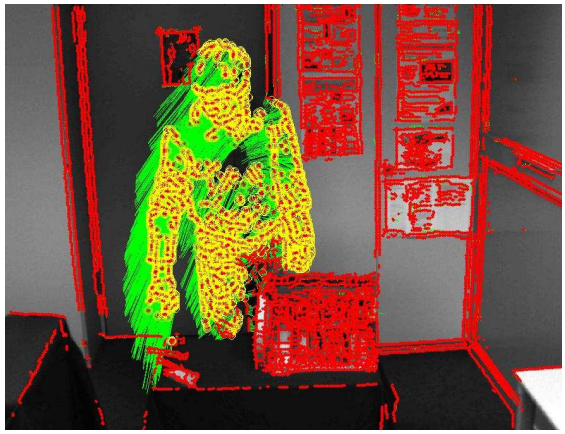


Abbildung 4.11.: Beispiel für die Hintergrundeleminierung im Produktionsszenario. Die 3D-Punkte M_P der Szene sind in rot eingezeichnet, die sich bewegenden Punkte M_m in gelb und die Bewegungsvektoren in grün. Datenbasis ist der Szenenfluss auf Basis von Korrelationsstereo und TV-L^1 Fluss

nenflusspunkte zu wählen. Gl. 4.21 beschreibt wie die Punktewolke M_m der sich bewegenden Punkte aus der Punktewolke aller Szenenflusspunkte M_p berechnet wird, wobei $\dot{u}$ und $\dot{v}$ die horizontale bzw. vertikale Komponente des optischen Flusses ist und P ein Szenenflusspunkt im Raum ist.

$$M_m = M_p \left\{ P | \sqrt{\dot{u}^2 \cdot \dot{v}^2} > \theta_{BS} \right\} \tag{4.21}$$

Die Schwelle wird in Pixeln pro Zeitschritt definiert. Würde man die Schwelle in Metern pro Zeitschritt definieren, würden weit entfernte, stationäre Punkte mit einem geringen Rauschen in der optischen Fluss-Berechnung immer als Vordergrund gezählt werden.

Nach der Hintergrundelimierung sind lediglich die bewegten Vordergrundobjekte, jedoch noch als zusammenhängende Punktewolke M_m, vorhanden (siehe Abb. 4.11).

4.3.2. Clusterverfahren

Zur Segmentierung der Punktewolke M_m der beobachteten Szene in einzelne Objekte M_c wird ein Clusterverfahren verwendet. Dabei muss das Verfahren verschiedene Kriterien erfüllen, um für die angestrebten Applikationen geeignet zu sein:

1. Die vorliegenden Szenenflussdaten bestehen aus den räumlichen Koordinaten x, y, z [m] und den Geschwindigkeitsinformationen $\dot{x}, \dot{y}, \dot{z}$ [m/Zeitschritt]. Anhand dieser soll entschieden werden, welche Teile der Punktewolke M_m zum selben Objekt gehören.

2. Da im Voraus keine Aussage über die Anzahl der interessierenden Objekte in der Szene gemacht werden kann, muss der Algorithmus selbst eine geeignete Anzahl

an Clustern bestimmen.

3. Zwei Punkte können weit entfernt voneinander sein und dennoch zu einem Cluster gehören, z.B. können auf einem Teil des Objekts keine Punkte extrahiert werden, da das Objekt dort keine Textur besitzt. Hingegen ist es auch möglich, dass relativ nahe Objekte nicht demselben Objekt angehören (z.B. Arm eines Arbeiters vor einem Bauteil). Die Geschwindigkeitsinformation ist also wichtig und zu berücksichtigen.

4. Das zu wählende Verfahren soll möglichst wenig Rechenzeit in Anspruch nehmen.

Die klassischen Clusterverfahren verwenden zum Vergleich von Datenpunkten meist ein eindimensionales Merkmal. Um die Vorgabe 1 umsetzen zu können, müßte man die Merkmale der Punkte, d.h. die räumlichen Koordinaten und die Geschwindigkeitsparameter gegeneinander gewichten, um schlussendlich auf einen skalaren Vergleichswert zu kommen. Bei räumlichen Koordinaten und Geschwindigkeitsinformationen ist das schwierig, da unterschiedliche Einheiten vorliegen. Außerdem würde dadurch die Parametersteuerung des Algorithmus erschwert. Daher wird darauf verzichtet und stattdessen räumliche Koordinaten und Geschwindigkeitsinformationen getrennt voneinander behandelt.

Für die beiden Szenenflusspunkte

$$\mathbf{P_1} = \begin{pmatrix} x_1 \\ y_1 \\ z_1 \\ \dot{x}_1 \\ \dot{y}_1 \\ \dot{z}_1 \end{pmatrix} \text{ und } \mathbf{P_2} = \begin{pmatrix} x_2 \\ y_2 \\ z_2 \\ \dot{x}_2 \\ \dot{y}_2 \\ \dot{z}_2 \end{pmatrix} \tag{4.22}$$

besteht das Distanzmaß $\delta_{\mathbf{dist}}$ aus den beiden Komponenten δ_e und δ_v:

$$\delta_{\mathbf{dist}} = \begin{pmatrix} \delta_e \\ \delta_v \end{pmatrix} \tag{4.23}$$

$$\delta_e = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \tag{4.24}$$

$$\delta_v = \sqrt{(\dot{x_1} - \dot{x_2})^2 + (\dot{y_1} - \dot{y_2})^2 + (\dot{z_1} - \dot{z_2})^2}. \tag{4.25}$$

Somit wird eine Trennung zwischen räumlichen Koordinaten und Geschwindigkeitsinformationen beibehalten.

Das verwendete Clusterverfahren (Konrad, 2006) arbeitet mit zwei Schwellen: die Schwelle für die räumliche Distanz θ_e und die Schwelle für den Geschwindigkeitsunterschied θ_v. Beide Schwellen müssen unterschritten werden, damit zwei Punkte dem gleichen Cluster zugeordnet werden können:

$$M_c = M_m \left\{ P | \delta_e < \theta_e \cap \delta_v < \theta_v \right\}. \tag{4.26}$$

Somit kann einfach und direkt Einfluss auf die Ausdehnung der Cluster genommen werden, da keine Bewertung und Zusammenfassung der einzelnen Merkmale (räumliche Koordinaten und Geschwindigkeitskomponenten) zu einem Skalar notwendig ist.

Die Etablierung von Clustern basiert bei dem Verfahren auf einem graphenbasierten Ansatz, wobei die einzelnen Punkte als Knoten und die Distanzen dazwischen als Kanten des Graphen betrachtet werden. Zunächst werden alle Datenpunkte ihres Geschwindigkeitsbetrags nach sortiert. Anschließend wird diese sortierte Liste $s = (s_1, ..., s_n)^T$ komplett durchlaufen. Man betrachtet einen Punkt $s_i = (x_i, y_i, z_i, \dot{x}_i, \dot{y}_i, \dot{z}_i)^T$ $(i = 1, ..., n)$ und die Distanzen zu den folgenden Punkten in der Liste. Sobald der Geschwindigkeitsunterschied zu einem der folgenden Punkte größer als die Schwelle δ_v ist, kann die Suche aufgrund der sortierten Liste abgebrochen werden. Sind bei den gefundenen Datenpunkten beide Schwellen unterschritten, werden die Punkte zu einem Cluster zusammengefasst. Ist die komplette Liste von Datenpunkten durchlaufen, sorgt abschließend ein Rekursionsansatz dafür, dass die Clusternummern richtig vergeben werden.

Vorteilhaft an diesem Algorithmus ist die hohe Effizienz und die einfache Parametersteuerung, was die Anwendbarkeit verbessert. Eine Gewichtung zwischen räumlichen Koordinaten und Geschwindigkeitsinformationen ist nicht notwendig. Außerdem können auch Szenenflussdaten von rotierenden Objekten geclustert werden, was aufgrund der sich kontinuierlich ändernden Geschwindigkeitsvektoren entlang der Objektachse oft problematisch ist. Da hier jedoch immer nur die Geschwindigkeitsdifferenzen benachbarter Punkte betrachtet werden, wird ein Zerfall der Punktewolke eines rotierenden Objekts in mehrere Cluster implizit verhindert.

4.3.3. Clusterauswahl

Um aus den einzelnen Clustern M_c, die aus der Clusteranalyse entstehen, das Cluster M_{Obj} auszuwählen, welches das interessierende Objekt repräsentiert, werden zusätzliche Informationen benötigt. Im Folgenden werden verschiedene Arten beschrieben, wie aus den Punkteclustern das Cluster, welches das interessierende Objekt repräsentiert, ausgewählt werden kann:

1. Produktionsumfeld: Distanz zu Referenzobjekten (z.B. Werkstück)
2. Pose-Estimation für alle Cluster und anschließende Verifikation (siehe Kap. 6.2), wobei die Verifikation dazu dient festzustellen, ob tatsächlich das interessierende Objekt gefunden wurde
3. Straßenverkehrsumfeld: Verfolgung aller Objekte, da alle von Interesse sind
4. Klassifikation des Bildinhaltes nach Clusteranalyse (z.B. Fahrzeugrückfrontenklassifikation oder Fußgängerklassifikation)

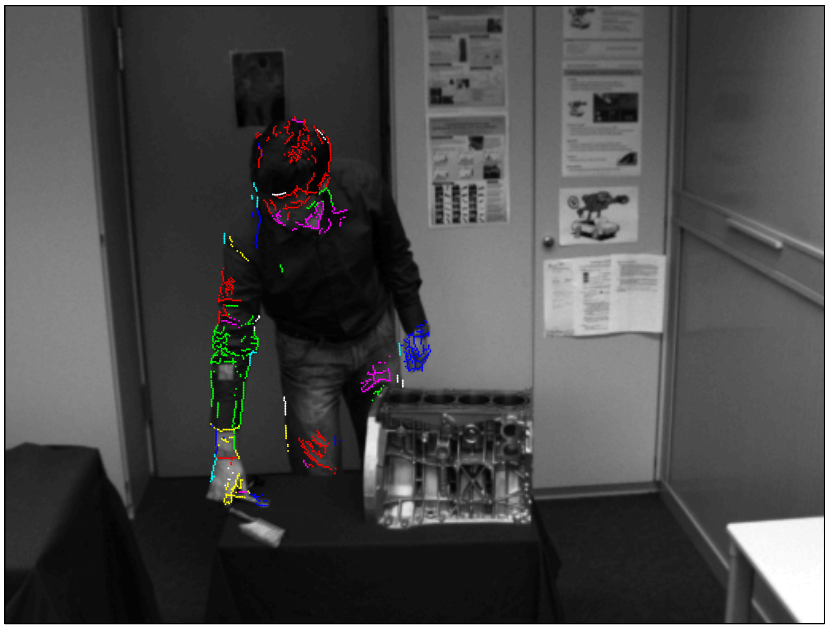


Abbildung 4.12.: Ergebnis des Clusterverfahrens: jede Farbe beschreibt ein Punktecluster (Farben wurden mehrfach verwendet).

Für das Produktionsumfeld werden Variante 1 und 2 benutzt und für das Straßenverkehrsumfeld wird Variante 3 verwendet. Durch die Modularität des Gesamtsystems sind andere Varianten bzw. Ergänzungen jederzeit möglich. Abschließend führt die Vorsegmentierung zu einer Punktewolke M_{Obj}, die das beobachtete Objekt repräsentiert.

4.4. Pose-Estimation

Nachdem ein initiale Objektpunktewolke M_{Obj} aus der Szene extrahiert wurde, wird das Modell des zu erkennenden Objekts (Abschnitt 4.4.1 bzw. 4.4.2), wie in Abschnitt 4.4.3 beschrieben, grob an diese Punkte angepasst, d.h. die Objektpose wird initialisiert. Anschließend wird über einen weiteren Anpassungsschritt diese Poseschätzung verfeinert (Abschnitt 4.4.4). Dabei wird auch die Punkteauswahl verändert, sodass am Ende der Optimierung nicht nur die Pose des Objekts bekannt ist, sondern auch die Zugehörigkeit der Punkte zum interessierenden Objekt. Die Optimierung der Zugehörigkeit der Szenenflusspunkte zum Objekt ist dabei aus zweierlei Gründen wichtig: Zum einen wird dadurch die iterative Poseschätzung verbessert und zum anderen werden diese Szenenflusspunkte später zur Bewegungsanalyse herangezogen, wo dem Objekt falsch zugeordnete Punkte

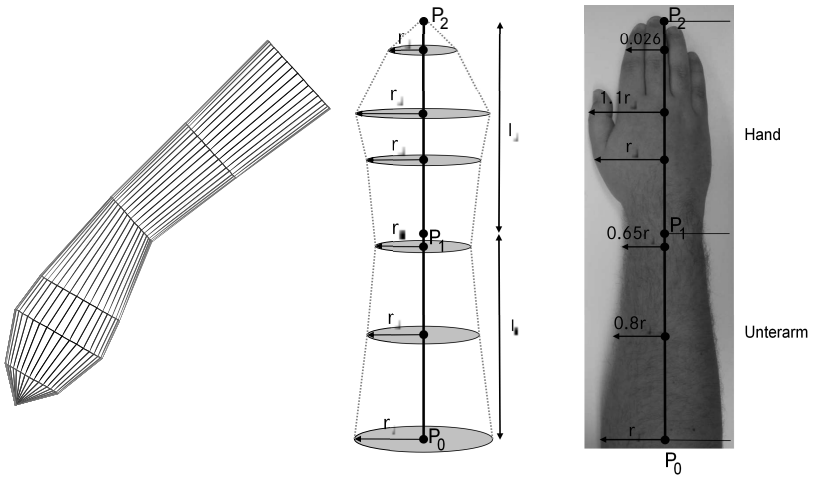


Abbildung 4.13.: Definition des Hand-Unterarm-Modells (Hahn et al., 2007).

das Ergebnis verfälschen würden.

Das 3D-Modell des zu erkennenden Objekts, welches für die Pose-Estimation zum Einsatz kommt, ist anwendungsspezifisch. Es kann mit sogenannten *schwachen Modellen* gearbeitet werden, d.h. das Objektmodell beschreibt die Form des interessierenden Objekts nur grob bzw. mit einem geringen Detaillierungsgrad. Solche Modelle sind zum einen einfach zu beschreiben und umzusetzen. Außerdem ist durch diese grobe Beschreibung des Objekts keine Modelladaption während der Poseoptimierung notwendig, was Rechenzeit spart und zudem die Robustheit des Gesamtsystems erhöht. Entscheidend ist jedoch, dass durch die Verwendung eines groben Modells die Modellparameter nicht für jedes neue Objekt aus der gleichen Objektklasse angepasst werden müssen. Beispielsweise kann im Produktionsumfeld das gleiche Modell mit den gleichen Modellparametern für alle Arbeiter verwendet werden, ohne zunächst deren Körpermaße bestimmen zu müssen. Dadurch verfügt das vorgestellte System über eine Robustheit, die sonst in der Literatur nur selten dargestellt wird. Die punktewolkenbasierte Modellanpassung mittels ICP-Algorithmus arbeitet problemlos mit den groben Modellen, da ein globales Optimum für die Poseschätzung unter Berücksichtigung aller 3D-Punkte gesucht wird. Daher wird im Endeffekt lediglich der Restfehler etwas höher, als bei einem perfekten Objektmodell, d.h. die Fehlerfunktion des verwendeten Pose-Estimation-Algorithmus, hat einen konstanten Offset, der jedoch die Optimierung nicht weiter beeinflußt. Die Poseschätzung zeigt jedoch keinen nennenswerten Qualitätsverlust.

4.4.1. 3D-Modell der Hand-Unterarm-Region

Form des Modells: Für das Produktionsumfeld wird das analytische Hand-Unterarm Modell von Hahn et al. (2007) verwendet, welches ein von der menschlichen Anatomie abgeleitetes, artikuliertes Modell, bestehend aus fünf Kegelstümpfen und einem vollständigen Kegel, darstellt (siehe Abb. 4.13). Das Modell ist eine kinematische Kette (engl.: kinematic chain), ein in der Literatur vielfach verwendeter Ansatz zur Modellierung von Körperpartien, welcher auf Leonardo Da Vinci zurückgeht (siehe Kap. 2.2).

Modellparameter: Hahn et al. (2007) definieren feste Relationen zwischen den einzelnen Radien des Modells, sodass nur zwei Radien (r_1 und r_4) geschätzt werden. Die Längen der Kegelstümpfe sind so definiert, dass nur die Gesamtlänge des Unterarms l_a und die Gesamtlänge der Hand l_h vordefiniert werden müssen. Die Relationen zwischen den Kegelstumpflängen sind im Modell hinterlegt.

Bei der neu entwickelten Anpassung mittels ICP-Algorithmus werden jedoch auch die Radien fest für alle beobachteten Personen definiert. Dafür gibt es zwei Gründe: Zum einen wird die Optimierung der Poseparameter durch die Festsetzung der Radien stabiler und zum anderen haben Experimente gezeigt, dass die Radien nicht zuverlässig aus Punktewolken zu schätzen sind.

Poseparameter: Die Position und Orientierung des rotationssymmetrischen Modells, sowie die internen Freiheitsgarde (zwei Knickwinkel des Handgelenks) werden über die sieben Poseparameter definiert:

$$\mathbf{\Phi} = \left[P_{0_x}, P_{0_y}, P_{0_z}, \alpha_1, \beta_1, \alpha_2, \beta_2\right] . \tag{4.27}$$

Dabei definiert der Punkt $\mathbf{P_0} = [P_{0_x}, P_{0_y}, P_{0_z}]^T$ den Aufpunkt des Modells im 3D-Raum was dem Punkt inmitten des menschlichen Ellbogens entspricht. Durch diesen Punkt ist zunächst die Position des Modells im Raum definiert. Die Winkel $\alpha_1, \beta_1, \alpha_2$ und β_2 definieren die Orientierung des Modells im Raum. Dabei beschreibt der Index 1 die Winkel für den Unterarm und der Index 2 die Winkel der Hand. Die jeweilige Differenz $\Delta\alpha = \alpha_1 - \alpha_2$ bzw. $\Delta\beta = \beta_1 - \beta_2$ beschreibt demnach die Knickwinkel der Hand zum Unterarm.

Zur Berechnung des Handgelenkpunkts P_1 bzw. des Fingerspitzenpunkts P_2 werden die Poseparameter mit den Modellparametern verknüpft:

$$P_1 = P_0 + \mathbf{R_y}(\alpha_1) \cdot \mathbf{R_z}(\beta_1) \cdot [0, 0, l_a]^T \quad , \tag{4.28}$$

$$P_2 = P_1 + \mathbf{R_y}(\alpha_2) \cdot \mathbf{R_z}(\beta_2) \cdot [0, 0, l_h]^T \quad . \tag{4.29}$$

Die Multiplikation mit den Matrizen $\mathbf{R_y}$ und $\mathbf{R_z}$ beschreiben eine Rotation um die y- bzw. z-Achse (Faugeras, 1993):

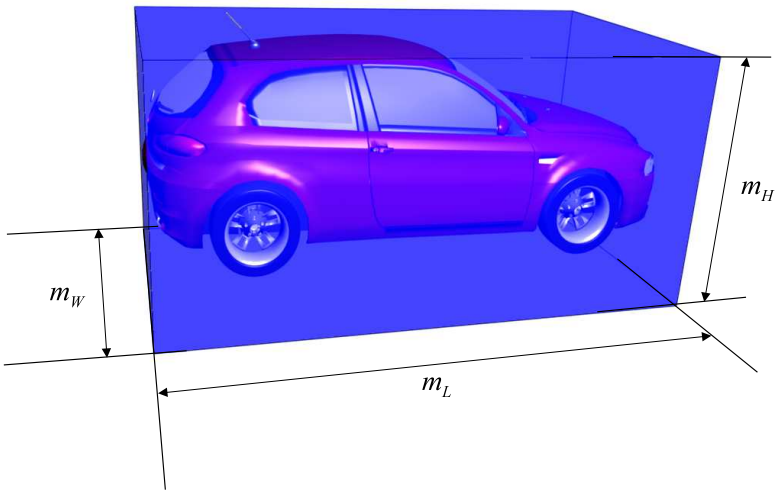


Abbildung 4.14.: Illustration des 3D-Fahrzeug-Modells, wofür ein Quader verwendet wird, mit den Modellparametern m_H, m_W und m_L.

$$\mathbf{R_y}(\alpha) = \begin{pmatrix} \cos\alpha & 0 & \sin\alpha \\ 0 & 1 & 0 \\ -\sin\alpha & 0 & \cos\alpha \end{pmatrix}; \; \mathbf{R_z}(\beta) = \begin{pmatrix} \cos\beta & -\sin\beta & 0 \\ \sin\beta & \cos\beta & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{4.30}$$

4.4.2. 3D-Fahrzeug-Modell

Form des Modells: Um möglichst viele verschiedene Fahrzeugtypen erkennen und in der Bildsequenz verfolgen zu können, wird wiederum ein schwaches Modell in Form eines Quaders verwendet. Bei der Anpassung dieses rigiden Modells an die segmentierte Punktemenge mittels ICP-Algorithmus werden die Abstände nur zu den sichtbaren Seiten des Quaders berechnet. Da der Quader sich in etwa auf der gleichen Höhe wie die Kamera befindet, können entweder eine oder zwei Seiten sichtbar sein. Das führt dazu, dass das Modell eigentlich ein Rechteck bzw. eine „L-Form“ darstellt.

Modellparameter: Die Größe des Quaders wird durch drei Parameter bestimmt: die Höhe m_H, die Breite m_W und die Länge m_L. Im Gegensatz zur Objektverfolgung im Produktionsszenario können hier jedoch nicht alle Modellparameter von vornherein fest definiert werden, da die Größe von Fahrzeugen im Straßenverkehr sehr unterschiedlich sein kann (z.B. Bus vs. Fahrrad). Es wird daher eine einmalige Schätzung der Modellparameter durchgeführt. Es wird die Breite des Fahrzeugs fest definiert, d.h. der Parameter, der in der Punktewolke über die Ausdehnung entlang der optischen Achse

des Kamerasystems geschätzt werden müsste, was durch die Tiefenungenauigkeit der Stereobildverarbeitung jedoch nicht möglich ist. Die Länge hingegen wird direkt aus der lateralen Ausdehnung der Punktewolke geschätzt, was sich in den Experimenten als praktikabel auszeichnete. Dadurch werden auch Randeffekte beispielsweise beim Ein- und Ausfahren aus dem Sichtbereich der Kamera, wobei die Ausdehnung der Punktewolke nur noch gering ist und dadurch eine Poseschätzung ungenau wird, implizit behandelt. Die Höhe des Quaders ist eine unkritische Größe. Da die Translationskomponente t_y entlang der Hochachse des Fahrzeugs unbetrachtet bleibt (siehe nächster Abschnitt), wird bei falscher Modellhöhe lediglich ein höherer Restfehler bei der Poseschätzung bleiben. Einen Einfluss auf die Qualität der Poseschätzung hat dies jedoch nicht.

Poseparameter: Die Pose eines rigiden Objekts im 3D-Raum wird über sechs Poseparameter definiert, drei Translationsparameter und drei Rotationsparameter. Dabei definieren die drei Translationsparameter t_x, t_y und t_z den Aufpunkt und somit die Position des Modells und die drei Rotationsparameter Θ_x, Θ_y und Θ_z die Orientierung des Modells im Raum. Als Aufpunkt wird für das Quadermodell der Mittelpunkt im Inneren des Modells definiert.

Im betrachteten Straßenverkehrsszenario können Annahmen getroffen werden, um die Anzahl der Poseparameter zu reduzieren. So kann davon ausgegangen werden, dass die anderen Fahrzeuge etwa mittig im Kamerabild erscheinen. Trifft diese Annahme nicht zu, kann zumindest davon ausgegangen werden, dass die Lage der Straßenoberfläche bekannt ist. Dadurch kann die vertikale Translationskomponente t_y außer Acht gelassen werden.

Außerdem können die Rotationswinkel um die fahrzeugeigene Längs- und Querachse (Θ_x und Θ_z) festgehalten werden. Die Schätzung dieser Winkel ist aus der Punktewolke nicht möglich, trägt aber auch nicht wesentlich zur Gefahrenabschätzung der vorliegenden Situation bei.

Durch diese Annahmen bleiben schlussendlich drei Poseparameter übrig: die Translation t_x entlang der x-Achse, die Translation t_z entlang der z-Achse und der sogenannte Gierwinkel (Rotation um die Hochachse des Fahrzeugs) Θ_y.

Die Pose $\mathbf{\Phi}$ ist somit beschrieben durch:

$$\mathbf{\Phi} = [t_x, t_z, \Theta_y]. \tag{4.31}$$

Durch die Beschränkung auf drei Parameter ist die Pose hier wesentlich schneller und nochmals robuster zu schätzen als bei einem sechsdimensionalen Posevektor.

4.4.3. Modellinitialisierung

Zur Initialisierung der Poseparameter bei der Modellanpassung wird zunächst der Schwerpunkt der Objektpunktewolke M_{Obj} benutzt, um die Position des Modells zu definieren. Anschließend wird eine Hauptkomponentenanalyse der Punktewolke dazu

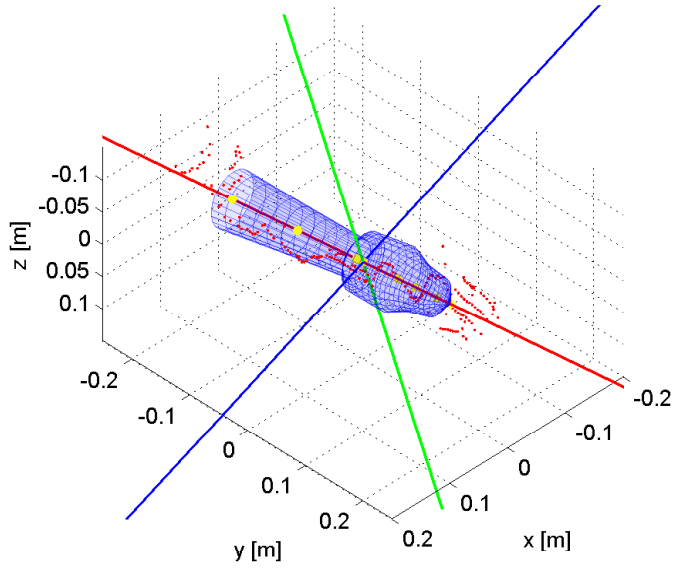


Abbildung 4.15.: Modellinitialisierung im Beispielszenario Produktion unter Verwendung des analytischen Hand-Unterarm-Modells (siehe Kap. 4.4.1) und der ersten Hauptkomponente des Objektpunkteclusters (rot).

benutzt, die Orientierungen des Modells zu bestimmen (siehe Abb. 4.15), wobei die Orientierung der ersten Hauptkomponente die initiale Orientierung des Modells festlegt.

Somit ist eine grobe Initialisierung der Objektpose vorhanden, um darauf aufbauend eine Verfeinerung der Schätzung durchzuführen. Interne Freiheitsgrade werden hier noch nicht geschätzt. Für das Beispiel des Hand-Unterarm Modells gilt also an dieser Stelle $\alpha_1 = \alpha_2$ und $\beta_1 = \beta_2$.

4.4.4. Iterative Closest Point Algorithmus zur Poseschätzung

Zur Anpassung eines 3D-Modells an eine 3D-Objektpunktewolke M_{Obj} ($P_{Obj} = (x_i, y_i, z_i)$, $P_{Obj} \in M_{Obj}$) wird häufig der sogenannte *Iterative Closest Point* Algorithmus (ICP) benutzt, welcher in Kap. 2.1.7 genauer beschrieben ist. Dabei wird ausgehend von einem Objektpunkt P_{Obj} der nächste Punkte auf der Modelloberfläche gesucht, d.h. eine Senskrecht die diesen Punkt mit dem Modell verbindet. Der Punkt auf dem Modell wird als P_m bezeichnet. Die Fehlerfunktion E besteht aus den aufsummierten, euklidischen Abständen zwischen den Objektpunkten und den korrespondierenden Punkte auf der Modelloberfläche (siehe Gl. 4.32).

$$E = \sum_{i=1}^{n} ||P_{Obj_i} - P_{m_i}||^2 \tag{4.32}$$

Bei der Verwendung dieser Methode in Verbindung mit 3D-Punkten aus der Stereobildverarbeitung kommt es jedoch zu deutlichen Problemen. Zum einen muss die Segmentierung initial sehr gut sein und zum anderen sollte das Objekt nur im Nahbereich vor dem Kamerasystem betrachtet werden. Eine schlechte Segmentierung generiert in der Fehlerfunktion der Modellanpassung viele Nebenminima, wodurch die Suche nach der richtigen Pose unmöglich wird. Werden Punkte mit einem zu hohen Rauschanteil verwendet, wird das Ergebnis ebenfalls verfälscht, da die Punktabstände in der ursprünglichen Version von Besl und McKay (1992) euklidisch zum Modell berechnet werden. Somit ist diese Version des ICP-Algorithmus für die angestrebten Applikationen ungeeigent.

In der vorliegenden Arbeit wird grundsätzlich der ICP-Algorithmus benutzt, jedoch in einer robusten, verbesserten Art. Hauptaugenmerk liegt hierbei auf der Ausreißerbehandlung. Unabhängig vom verwendeten Clusteransatz ist eine perfekte Segmentierung der Punktewolke in der Realität nicht verfügbar. D.h. es muss immer damit gerechnet werden, dass Punkte, die nicht zum Objekt gehören, in der Objektpunktewolke auftauchen. Umgekehrt sind auch Objektpunkte evtl. einem anderen Cluster zugeordnet, sprich sie fehlen in der Objektpunktewolke. Außerdem sind natürlich auch immer Ausreißer im Sinne von Messfehlern vorhanden. Um eine robuste Modellanpassung auf realen Daten durchzuführen, müssen all diese Fälle betrachtet und behandelt werden.

Gewichtung der Abstände: Um eine Robustheit gegenüber Messfehlern und fälschlicherweise zugeordneten Punkten zu erhalten, wird in der Abstandsberechnung ein sogenannter M-Schätzer (engl.: M-Estimator) verwendet (Rey, 1983). Diese Gewichtungsfunktion führt dazu, dass Punkte mit einem kleinen Abstand zum Modell normal gewichtet werden, Punkte die jedoch sehr weit weg vom Modell auftauchen, geringer gewichtet werden. Würde man keine Gewichtung einführen, würde bereits ein Ausreißer die Poseschätzung unbrauchbar machen. Die verwendete Gewichtungsfunktion wird als *Fair*-Gewichtungsfunktion bezeichnet:

$$w_f(E) = \frac{1}{1 + \frac{|E|}{c_M}}. \tag{4.33}$$

Das Gewicht w_f wird dadurch direkt aus dem ursprünglichen Fehler E (siehe Gl. 4.32) bestimmt. Der Parameter c_M ist eine definierbare Konstante, die Einfluss auf die Gewichtung hat.

Korrespondenzbildung: Wie der Name Iterative Closest Point bereits aussagt, wird in der ursprünglichen Version von Besl und McKay (1992) iterativ nach dem nächsten Punkt gesucht und dabei die Pose immer weiter angepasst. In unserem Fall besteht das Objektmodell jedoch nicht aus einzelnen Punkten, sondern aus Flächen bzw. aus dreidimensionalen Körpern. Daher muss zunächst auf eine andere Art der Korrespondenzbildung umgestellt werden. Die Suche nach dem nächsten Punkt wird in der Literatur als *Point-to-Point Matching* bezeichnet. In unserem Fall wird hingegen ein sogenanntes *Point-to-Plane Matching* verwendet. Dabei wird der nächste Punkt auf dem Modell ge-

sucht, sprich eine Senkrechte, die den 3D-Punkt und das Modell verbindet. Diese Art der Korrespondenzsuche wird entsprechend der Grundidee auch während jeder Iteration neu berechnet. Dadurch werden Fehlzuordnungen während der Pose-Estimation korrigiert.

Korrektur der Punkteauswahl: Durch die Verwendung des M-Schätzers wird zwar der Einfluss der Ausreißer auf die Poseschätzung verringert, jedoch nicht ganz entfernt. Daher wird die Punktewolke in ihrer Zusammensetzung ebenfalls während der Optimierung verändert. Zunächst wird eine bestimmte Anzahl an Iterationen durchlaufen. Danach kann ein Histogramm über die Punktabstände zum Modell erstellt werden. Aus diesem Histogramm ist nun ersichtlich, welche Punkte offensichtlich zum Modell gehören und welche Fehlzuordnungen aus einer falschen Segmentierung sind. In der Praxis hat sich gezeigt, dass eine Schwelle $D_{\max}$ bestehend aus Mittelwert $\overline{dist}$ und Standardabweichung $\sigma(dist)$ der Punktabstände sinnvoll ist:

$$D_{\max} = \overline{dist} + f_{\text{std}}\sigma(dist), \tag{4.34}$$

wobei über den Faktor f_{std} die Schwelle noch individuell gesteuert werden kann.

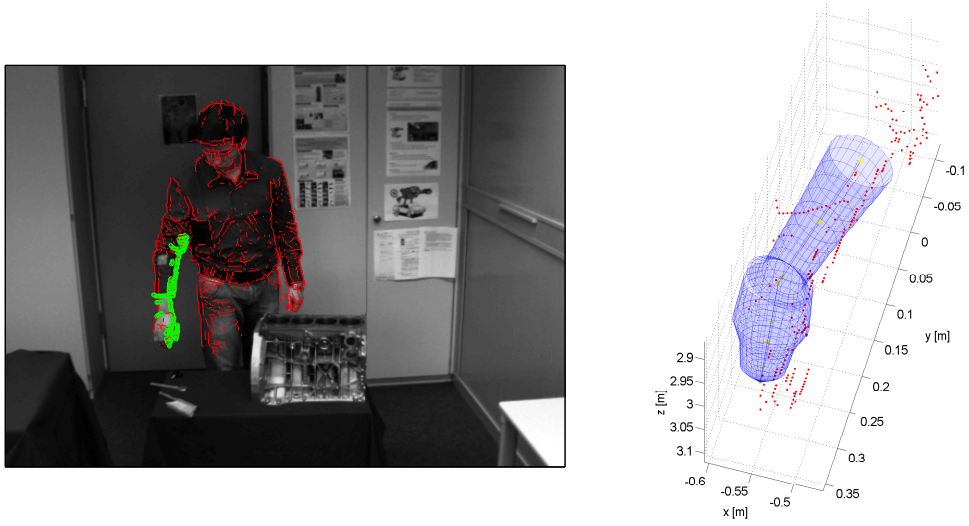
Anders als in der Literatur (Zhang, 1992) üblich wird diese Schwelle jedoch nicht nur dazu benutzt, objektfremde Punkte aus der Punktewolke auszuschließen, sondern auch um Objektpunkte, die nicht korrekt der Punktewolke zugewiesen wurden wieder in die Menge aufzunehmen. Somit ergibt sich folgende Vorschrift für die Objektpunktewolke:

$$M_{Obj} = M_P \left\{P | dist(P, \mathbf{\Phi}) < D_{\max}\right\}. \tag{4.35}$$

4.4.5. Hand-Unterarm ICP

Ein Problem, welches durch die längliche Hand-Unterarm-Region entsteht, ist die Ungenauigkeit bei der Schätzung der Translation entlang der Hand-Unterarm-Achse. Die Fehlerfunktion ist entlang dieser Achse sehr flach, da hier, bei Verwendung der klassischen ICP-Fehlerfunktion (Besl und McKay, 1992), keine signifikante Änderung des Abstandsfehlers eintritt. Außerdem findet eine ungewollte Verschiebung entlang dieser Achse statt, wenn lediglich ein Teil der Hand-Unterarm-Region in der Punktewolke sichtbar ist. In Abb. 4.18 ist die ursprüngliche Fehlerfunktion für die Verschiebung des Hand-Unterarm-Modells entlang und orthogonal zur Hand-Unterarm-Achse zu sehen. Dabei zeigt sich, dass keine Signifikanz entlang der Achse besteht, wohingegen orthogonal dazu sehr eindeutig ein Minimum auszumachen ist. Hierbei wird das Point-to-Plane-Matching zur Etablierung von Punktkorrespondenzen verwendet, wobei die euklidische Distanz zum nächsten Punkt auf der Modelloberfläche gesucht wird.

Durch die Einführung eines zusätzlichen Fehlerterms $||p - P_3||$ in der Fehlerfunktion des ICP-Algorithmus e_{ICP} (siehe Gl. 4.36), lässt sich das Problem beheben. Diese Distanz $||p - P_3||$ beschreibt den Abstand eines segmentierten Objektpunkts p zum Punkt P_3 des Modells, d.h. zur Spitze der Hand. Dieser zusätzliche Fehlerterm $||p - P_3||$ wird über den Gewichtungsparameter λ_{ht} mit dem euklidischen Distanzmaß zwischen Modell und Objektpunkt $||p - p_m||$ verknüpft.



(a) Ergebnis der Clusterauswahl

(b) Ergebnis der Modellanpassung

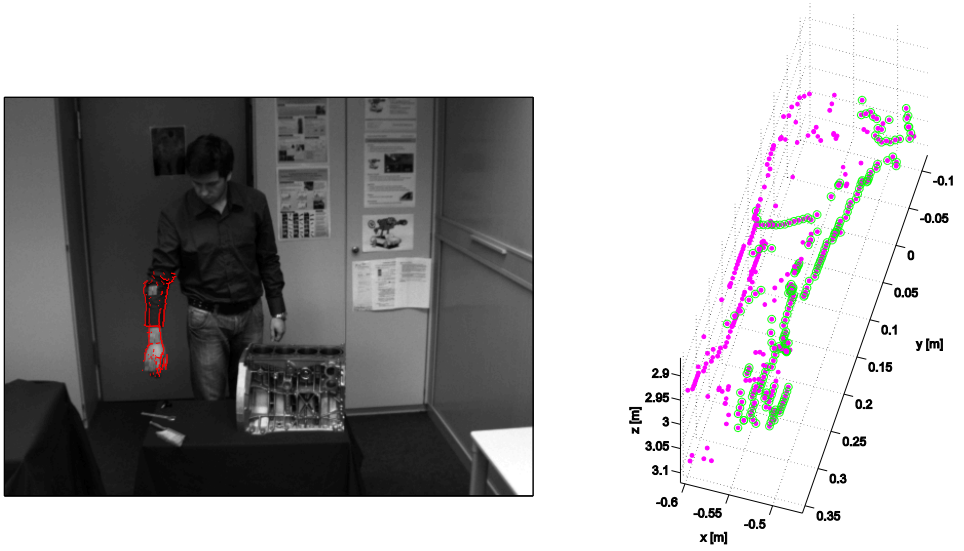
Abbildung 4.16.: (a) Punktewolke des menschlichen Körpers (rot) und das initiale Objektcluster M_{Obj} der Hand-Unterarm-Region (grün). (b) 3D Hand-Unterarm-Modell (blau) an die 3D Punktewolke (rot) angepasst.

$$e_{\text{ICP}} = ||p - p_m|| + \lambda_{ht}||p - P_3|| \tag{4.36}$$

Nach der Erweiterung der Fehlerfunktion zeigt sich ein ähnliches Verhalten für die Verschiebung entlang der Hand-Unterarm-Achse und die Verschiebung orthogonal dazu. Durch den Gewichtungsparameter kann direkt Einfluss auf die longitudinale Verschiebung genommen werden und so die Ungenauigkeit bei der Poseschätzung deutlich verringert werden. Nach wie vor werden alle Poseparameter inklusive der internen Freiheitsgrade in Form der beiden Knickwinkel des Handgelenks geschätzt.

4.4.6. Iterative Closest Point-Algorithmus mit problemspezifischer Fehlermetrik

Die ursprüngliche, euklidische Berechnung der Abstände zwischen Objektpunkte und Modell (Punkte zu Modelloberfläche, siehe Kap. 4.4.4) hat einen Nachteil: Sie ist nicht an das Rauschverhalten der extrahierten 3D-Punkte angepasst. Die Fehlerfunktion E (siehe Gl. 4.32), welche einen mittleren quadratischen Fehler aller Abstände berechnet, liefert ein Optimum für gaußverteilte Fehler, jedoch rauschen die Koordinaten der 3D-Objektpunkte nicht gaußverteilt. Im Folgenden wird diese Problematik genauer beschrieben und der entwickelte Lösungsansatz vorgestellt.



(a) Abschließendes Segmentierungsergebnis

(b) Initiale und endgültige Punktewolke

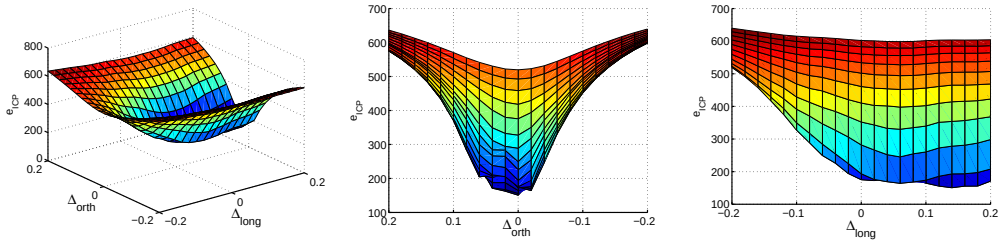
Abbildung 4.17.: (a) Verfeinerte Punkteauswahl für die Hand-Unterarm-Region. (b) 3D Punktewolke mit den initialen Punkten (grün) und allen Objektpunkten (violett).

Charakteristik der Stereomessung: Zur Berechnung eines 3D-Punkts ${}^{W}\mathbf{w}$ werden die korrespondierenden Bildpunkte ${}^{J}\mathbf{m}_l$ und ${}^{J}\mathbf{m}_r$ zusammen mit der Kamerakonstante f und der Basisbreite b verwendet. Dabei sind die beiden Abbildungen ${}^{J}\mathbf{m}_l$ und ${}^{J}\mathbf{m}_r$ eines exakten Punkts im 3D-Raums, Messgrößen, welche systembedingt gaußverteilt rauschen, was zu einer ebenso gaußverteilten Streuung der Disparität d führt (Stein et al., 2006). Wie in Gl. 4.37 zu sehen, besteht bei der Berechnung eines 3D-Punkts zwischen der Disparität und den Weltkoordinaten x, y und z eine umgekehrt proportionale Beziehung.

$$x = \frac{u_r \cdot f}{d} \qquad y = \frac{v_r \cdot f}{d} \qquad z = \frac{f \cdot b}{d} \tag{4.37}$$

Wie in Abb. 4.19 zu sehen, führt der nichtlineare Zusammenhang zwischen d und z dazu, dass nahe Punkte (d.h. große Disparität) genau bestimmt werden können, wobei die Unsicherheit Δz für weit entfernte Punkte stark zunimmt. Bei gleichem Messfehler für die Disparität Δd führt dies zu unterschiedlichen, entfernungsabhängigen Fehlern bei der Tiefenschätzung (Δz_1, Δz_2).

Durch die Definition, dass der Ursprung des Weltkoordinatensystems im Zentrum der rechten Kamera liegt (siehe Kap. 2.1.1), beschreibt die Bildkoordinaten des rechten Punktes ${}^{J}\mathbf{m}_r = (u_r, v_r)^T$ den Sehstrahl, auf dem der beobachtete 3D-Punkt liegt. An dieser Stelle ist die Messung sehr genau, da ein direkter linearer Zusammenhang zwi-



(a) 2D Fehlergebirge (b) Verschiebung orthogonal zur Hand-Unterarm-Achse (c) Verschiebung parallel zur Hand-Unterarm-Achse

Abbildung 4.18.: Verhalten der ursprünglichen Fehlerfunktion E aus Gl. 4.32 um das Optimum herum.

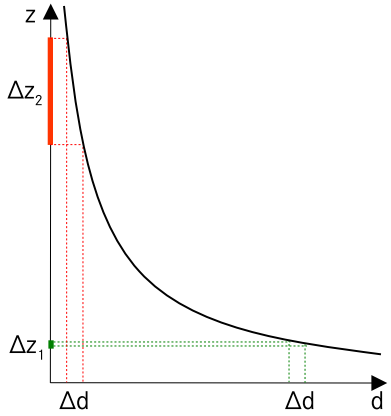


Abbildung 4.19.: Die Abhängigkeit zwischen dem Tiefenwert z und der Disparität d.

schen gemessenem Bildpunkt und Sehstrahl besteht. Zur Bestimmung des 3D-Punkts auf dem Sehstrahl wird die gemessene Disparität verwendet. Es kommt bei weit entfernten Punkten zu einer großen Unsicherheit bei der Bestimmung der 3D-Koordinaten, da die Triangulation, auf welcher die Bestimmung basiert, zu einem nichtlinearen Zusammenhang zwischen Disparität und Abstand auf dem Sehstrahl führt. Die Unsicherheit dieses Abstandes nimmt mit der Entfernung zu. Des Weiteren führt der nichtlineare Zusammenhang zwischen d und z zu einem nicht gaußverteilten, nicht symmetrischen Rauschen der Tiefenschätzung. Durch den limitierten Öffnungswinkel der Kameras ist das auftretende Rauschen maßgeblich in der z-Komponente enthalten.

Wird der Ansatz nach Besl und McKay (1992) zur Modellanpassung mittels ICP-Algorithmus verwendet, so führt diese Charakteristik zu ungenauen Poseschätzungen, insbesondere bei weit entfernten Objekten ($z \gg b$). Grund dafür ist die Vorgehensweise bei der Poseschätzung, die implizit nach der Methode der kleinsten Quadrate arbeitet (Zhang, 1992). Das führt zu der Annahme, dass die Messpunkte im dreidimensionalen

Raum gaußverteilt rauschen oder zumindest symmetrisch rauschen. Wird diese Bedingung nicht erfüllt, kommt es zu einem systematischen Fehler.

Die Problemstellung der Modellanpassung muss auf gaußverteilte oder zumindest symmetrische Unsicherheiten aufbauen, wenn die Lösung auf Basis der Methode der kleinsten Quadrate verwendet wird. Eine Variante, dies für die Stereodaten zu erreichen, wäre eine Modellanpassung in einem Koordinatensystem, welches durch u,v und d aufgespannt wird. Hierzu müsste allerdings auch das Modell in diesen Raum transformiert werden, was modellabhängig sehr komplex sein kann. Daher wird im Folgenden die Modellanpassung durch die Anpassung der Fehlermetrik auf gaußverteilte Fehler zurückgeführt, um weiterhin beliebige Modelle verwenden zu können.

Rückführung auf gaußverteilte Fehlermaße: Aufgrund der beschriebenen Charakteristik der Stereomessung wird nun jeder 3D-Punkt in Polarkoordinaten beschrieben, d.h. ein 3D-Punkt ist über die Winkel zur optischen Achse und dem Abstand zum Koordinatenursprung definiert. Durch diese Darstellung lässt sich die geringe Unsicherheit in der Winkelschätzung von der hohen Unsicherheit bei der Abstandsschätzung trennen.

Für die Berechnung des Abstands r (Gl. 4.38) zwischen Kamera und gemessenem 3D-Punkt soll der Fehler auf eine gaußverteilte Fehlergröße, auf die Disparität d, zurückgeführt werden. Der Abstand r berechnet sich aus den gemessenen Größen wie folgt:

$$r = \sqrt{x^2 + y^2 + z^2} = \sqrt{\left\{\frac{ub}{d}\right\}^2 + \left\{\frac{vb}{d}\right\}^2 + \left\{\frac{fb}{d}\right\}^2}, \tag{4.38}$$

wobei u und v die Bildkoordinaten im Bild der rechten Kamera sind.

Laut Fehlerfortpflanzungsgesetz (Bronshtein und Semendyayev, 1997) wird der Fehler einer nicht direkt messbaren Größe (in diesem Fall der Abstand zum Koordinatenursprung r) aus den Fehlern der messbaren Größen (in diesem Fall u, v und d) anhand des totalen Differentials bestimmt. In Gl. 4.39 ist der Zusammenhang zwischen dem Abstandsfehler Δr und den Messfehlern Δu, Δv und Δd gezeigt:

$$\Delta r = \frac{\partial r}{\partial u}\Delta u + \frac{\partial r}{\partial v}\Delta v + \frac{\partial\, r}{\partial d}\Delta d\ . \tag{4.39}$$

Die partiellen Ableitungen $\partial r/\partial u$ in Gl. 4.40, $\partial r/\partial v$ in Gl. 4.41 und $\partial r/\partial d$ in Gl. 4.42 werden anhand der Gl. 4.37 und Gl. 4.38 berechnet.

$$\frac{\partial r}{\partial u} = \frac{1}{2}\cdot \underbrace{\left[\left(\frac{ub}{d}\right)^2 + \left(\frac{vb}{d}\right)^2 + \left(\frac{fb}{d}\right)^2\right]^{-\frac{1}{2}}}_{=1/r} 2\frac{ub^2}{d^2}$$

$$
\begin{aligned}
&= \frac{1}{r} \cdot \underbrace{\frac{ub}{d}}_{=x} \cdot \underbrace{\frac{b}{d}}_{=z/f} \\
&= \frac{xz}{rf} \qquad (4.40)
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial r}{\partial v} &= \frac{1}{2} \cdot \underbrace{\left[\left(\frac{ub}{d}\right)^2 + \left(\frac{vb}{d}\right)^2 + \left(\frac{fb}{d}\right)^2\right]^{-\frac{1}{2}}}_{=1/r} \cdot 2\frac{vb^2}{d^2} \\
&= \frac{1}{r} \cdot \underbrace{\frac{vb}{d}}_{=y} \cdot \underbrace{\frac{b}{d}}_{=z/f} \\
&= \frac{yz}{rf} \qquad (4.41)
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial r}{\partial d} &= \frac{1}{2} \cdot \frac{1}{r} \cdot \left[-2\frac{u^2b^2}{d^3} - 2\frac{v^2b^2}{d^3} - 2\frac{b^2f^2}{d^3}\right] \\
&= \frac{1}{2} \cdot \frac{1}{r} \cdot \left(-\frac{2}{d}\right) \cdot \underbrace{\left[\left(\frac{ub}{d}\right)^2 + \left(\frac{vb}{d}\right)^2 + \left(\frac{fb}{d}\right)^2\right]}_{=r^2} \\
&= -\frac{rz}{bf} \qquad (4.42)
\end{aligned}
$$

Durch Abschätzung der Größenordnungen der Parameter wird gezeigt, dass der Fehler Δd der Disparität die Abstandrechnung am stärksten beeinflusst. Im Kreuzungsszenario liegt der Wert der Weltkoordinate z im Bereich bis etwa 50 m, was einer Größenordnung von 10^1 m entspricht. Bei dieser Entfernung hat bei dem verwendeten Stereokamerasystem die Weltkoordinate x einen Wert von ca. ± 10 bis ± 30 m, was ebenso einer Größenordnung von 10^1 m entspricht. Die Weltkoordinate y liegt durch die Annahme, dass die Straße parallel zur x-z-Ebene des Kamerakoordinatensystems ist, im Bereich ± 2 m, was der Größenordnung 10^0 m entspricht. Der Abstand r vom optischen Zentrum der Kamera zum 3D-Punkt ist von der gleichen Größenordnung wie die z-Koordinate, also 10^1 m. Die Basisbreite b, welche etwa zwischen 10 und 35 cm liegt, ist von der Größenordnung 10^{-1} m. Die Kamerakonstante f, welche einen Wert von ca. 1000 bis 1400 Pixel hat, ist von der Größenordnung 10^3 Pixel. Demzufolge gelten bei diesem Szenario, d.h. unter Verwendung eines Stereokamerasystems, welches eine um mindestens eine Größenordnung kleinere Basisbreite b als die gemessenen Abstände r hat, folgende Abschätzungen:

$$
\left|\frac{\partial r}{\partial u}\right| = \left|\frac{xz}{rf}\right| \rightarrow \text{ Größenabschätzung: } \frac{10^1 m \cdot 10^1 m}{10^1 m \cdot 10^3 px} = 10^{-2}\frac{m}{px} \qquad (4.43)
$$

$$\left|\frac{\partial r}{\partial v}\right| = \left|\frac{yz}{rf}\right| \rightarrow \text{ Größenabschätzung: } \frac{10^0 m \cdot 10^1 m}{10^1 m \cdot 10^3 px} = 10^{-3}\frac{m}{px} \tag{4.44}$$

$$\left|\frac{\partial r}{\partial d}\right| = \left|\frac{rz}{bf}\right| \rightarrow \text{ Größenabschätzung: } \frac{10^1 m \cdot 10^1 m}{10^{-1} m \cdot 10^3 px} = 1\frac{m}{px} \tag{4.45}$$

Da der Unterschied in den Größenordnungen der Fehlerauswirkungen so groß ist, können die Messfehler der Bildkoordinaten Δu bzw. Δv, welche mit den Faktoren $\partial r/\partial u$ (siehe Gl. 4.40) und $\partial r/\partial v$ (siehe Gl. 4.41) die Abstandrechnung beeinflussen, bei der Rückführung der Abstandrechnung auf gaußverteilte Messgrößen vernachlässigt werden. Daher gilt in sehr guter Näherung folgender Zusammenhang:

$$\Delta r \approx \frac{rZ}{bf}\Delta d \tag{4.46}$$

Die Kameraparameter b und f sind konstant und Δd ist gaußverteilt, weswegen der Term rZ aus Gl. 4.46 zur Normierung der Abstände in der Fehlerfunktion benutzt wird, um das gaußverteilte Fehlermaß $\Delta r/rZ$ zu erhalten.

Neue Fehlermetrik: Wie gezeigt wurde, ist es sinnvoll die Fehlermaße für den Abstandsfehler E_r (siehe Gl. 4.48) und für den Winkelfehler E_φ (siehe Gl. 4.49) getrennt voneinander zu betrachten. Im Gesamtfehler E werden beide Terme als gewichtete Summe über den Parameter λ miteinander verknüpft:

$$E^2 = E_r^2 + \lambda E_\varphi^2. \tag{4.47}$$

Der Abstandsfehler E_r hängt nur von r_i, r_{m_i} und z_i ab, wobei r_i die Distanz zwischen Kamera und gemessenem 3D-Punkt ist, r_{m_i} die Distanz zwischen Kamera und Modelloberfläche auf dem gleichen Sehstrahl beschreibt und z_i die z-Koordinate des gemessenen 3D-Punkts ist. Durch die Rückführung auf gaußverteilte Fehlermaße erhält man:

$$E_r(r_i, z_i) = \frac{r_i - r_{m_i}}{z_i r_i}. \tag{4.48}$$

Der Winkelfehler E_φ hängt von der Differenz zwischen den Polarwinkeln φ_i des 3D-Punkts i und φ_m des Mittelpunkts des Modells ab:

$$E_\varphi(\varphi_i) = \frac{\varphi_i - \varphi_m}{2}(1 + \tanh|\alpha(|\varphi_i - \varphi_m| - \beta)|). \tag{4.49}$$

Der Tangens Hyperbolicus sorgt für die Stetigkeit der Fehlerfunktion und wird für das Konvergenzverhalten der Optimierung benötigt. Der Parameter β korrespondiert mit ca. dem halben Winkel der Breite des Modells projiziert auf die Bildebene. Sowohl λ als auch α sind benutzerdefinierte Parameter, die das Konvergenzverhalten der Optimierung beeinflussen. Der Parameter λ dient außerdem zum Angleichen der Maßeinheiten zwischen Abstands- und Winkelfehler. Abb. 4.20 zeigt die Fehlerfunktion E_φ, die durch die

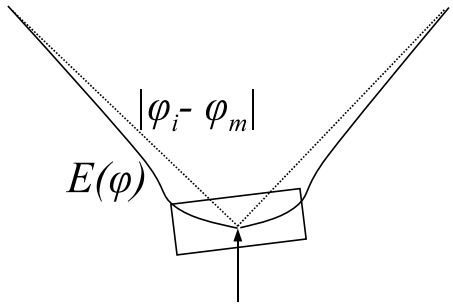


Abbildung 4.20.: Darstellung der Winkeldifferenz $|\varphi_i - \varphi_m|$ (gestrichelt) und dem Fehlerterm E_φ (durchgezogen). Die Winkeldifferenz ist an der Nullstelle unstetig, im Unterschied zum Fehlerterm, der durch das Einfügen der Tangens-Hyperbolicus-Funktion an dieser Stelle stetig bleibt.

Verwendung des Tangens Hyperbolicus entsteht und wodurch die Fehlerfunktion stetig und differenzierbar wird.

Durch die Anpassung der Fehlermetrik steht nun eine Form des ICP-Algorithmus zur Verfügung, die die Charakertistik der Stereomessung berücksichtig und beliebige Modelle zulässt. Der Aufwand zur Berechnung der Fehlerfunktion ist ähnlich der des klassischen Ansatzes.

4.4.7. Quaternionen-ICP

Geschlossene Lösungsverfahren für das Problem der relativen Orientierungsschätzung zwischen zwei Punktewolken mit bekannten Korrespondenzen werden beispielsweise von Arun et al. (1987), Zhang (1992) oder von Horn (1987b) beschrieben. Horn (1987b) benutzt Quaternionen für die direkte Lösung, wobei diese Lösung sehr effizient zu berechnen ist. Für rauschbehaftete Daten ist die Robustheit des Verfahrens nicht ausreichend, da implizit die Methode der kleinsten Quadrate auf die euklidischen Punktabstände angewendet wird, welche anfällig für Ausreißer ist. Außerdem können Modelle, die durch Flächen beschrieben sind, nicht mit dieser Methode verarbeitet werden. Es wird also vorausgesetzt, dass das Modell ebenfalls aus einzelnen Punkten besteht und korrekte Punktzuordnungen zwischen aufgenommener Punktewolke und Modell bekannt sind.

Eines der Ziele dieser Arbeit ist es einen Algorithmus zu entwickeln, der einerseits schneller und effizienter durchführbar ist, als der ursprüngliche ICP-Algorithmus mit Ausreißerbehandlung (siehe Abschnitt 4.4.4), wobei die hohe Robustheit des Verfahrens gewahrt werden soll. Die iterative Ermittlung von Punktkorrespondenzen wie im ICP-Algorithmus muss erhalten bleiben, da feste Zuordnungen zu Beginn nicht bereitgestellt werden können.

Bei diesem neuentwickelten Verfahren wird zunächst eine Initialpose wieder über den Clusterschwerpunkt und die erste Hauptkomponente bestimmt (siehe Abschnitt 4.4.3). Anschließend werden, wie bei der iterativen Modellanpassung, Korrespondenzen zwi-

schen Modell und Punktewolke über einen Point-to-Plane-Ansatz etabliert. So wird zu einem 3D-Punkt $\mathbf{w_{p_i}}$ $(i = 1, ..., n)$ der nächstgelegene Modellpunkt $\mathbf{w_{m_i}}$ $(i = 1, ..., n)$ bestimmt, wobei n die Anzahl der gemessenen 3D-Punkte beschreibt.

Zunächst werden die Schwerpunkte der beiden Punktemengen bestimmt:

$$\overline{\mathbf{w_p}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{w_{p_i}} = \frac{1}{n}\sum_{i=1}^{n}\begin{pmatrix} x_{p_i} \\ y_{p_i} \\ z_{p_i} \end{pmatrix}, \quad \overline{\mathbf{w_m}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{w_{m_i}} = \frac{1}{n}\sum_{i=1}^{n}\begin{pmatrix} x_{m_i} \\ y_{m_i} \\ z_{m_i} \end{pmatrix}. \tag{4.50}$$

Anschließend werden diese Mittelpunkte von der jeweiligen Punktemenge subtrahiert, was zu den neuen Punktemengen $\mathbf{w'_{p_i}}$ und $\mathbf{w'_{m_i}}$ führt:

$$\mathbf{w'_{p_i}} = \mathbf{w_{p_i}} - \overline{\mathbf{w_p}} = \begin{pmatrix} x'_{p_i} \\ y'_{p_i} \\ z'_{p_i} \end{pmatrix}, \quad \mathbf{w'_{m_i}} = \mathbf{w_{m_i}} - \overline{\mathbf{w_m}} = \begin{pmatrix} x'_{m_i} \\ y'_{m_i} \\ z'_{m_i} \end{pmatrix}. \tag{4.51}$$

Die von Horn (1987b) vorgestellte Lösung basiert auf Quaternionen, welche eine Erweiterung der reellen Zahlen, ganz ähnlich zu den komplexen Zahlen, darstellt. Ein Quaternion lässt sich über vier reelle Zahlen q_a, q_b, q_c, q_d beschreiben:

$$\dot{\mathbf{q}} = q_a + q_b \cdot i + q_c \cdot j + q_d \cdot k, \tag{4.52}$$

wobei ähnlich zu den komplexen Zahlen gilt: $i^2 = j^2 = k^2 = -1$. Einheitsquaternionen haben zudem die Eigenschaft, ähnlich zu Einheitsvektoren, dass der Betrag gleich 1 ist: $||\dot{\mathbf{q}}|| = 1$.

Quaternionen werden oft für Drehungen im 3D-Raum genutzt, da sie hier eine sehr elegante Beschreibung darstellen, die gegenüber anderen Darstellungen verschiedene Vorteile aufzeigt, z.b. wenn mehrere Drehungen hintereinander erfolgen, sind weit weniger Rechenoperationen als bei herkömmlichen Rotationsmatrizen notwendig, weshalb gerade in der Computergrafik Quaternionen häufig eingesetzt werden. Für Quaternionen gelten besondere Rechenregeln, die beispielsweise von Hamilton (1866) beschrieben werden.

Horn (1987b) zeigt, dass zur Ermittlung der Transformation zwischen den beiden Punktewolken $\mathbf{w_{p_i}}$ und $\mathbf{w_{m_i}}$ ein Einheitsquaternion $\dot{\mathbf{q}}$ gefunden werden muss, welches folgenden Ausdruck maximiert:

$$\sum_{i=1}^{n}(\dot{\mathbf{q}}\dot{\mathbf{w}}'_{\mathbf{p_i}}\dot{\mathbf{q}}^*)\dot{\mathbf{w}}'_{\mathbf{m_i}} \ . \tag{4.53}$$

Durch weitere Umformungen, unter Berücksichtigung der Rechenregeln für Quaternionen, erhält man folgenden zu maximierenden Ausdruck:

$$\dot{\mathbf{q}}^T\mathbf{N}\dot{\mathbf{q}} \ . \tag{4.54}$$

Die Matrix $\mathbf{N}$ stellt dabei die Summe der Punktbeziehungen zueinander dar:

$$\mathbf{N} = \sum_{i=1}^{n} \mathbf{N_i} = \sum_{i=1}^{n} \mathbf{R_{p_i}^T} \mathbf{R_{m_i}} \tag{4.55}$$

$$\mathbf{R_{p_i}} = \begin{pmatrix} 0 & -x'_{p_i} & -y'_{p_i} & -z'_{p_i} \\ x'_{p_i} & 0 & z'_{p_i} & -y'_{p_i} \\ y'_{p_i} & -z'_{p_i} & 0 & x'_{p_i} \\ z'_{p_i} & y'_{p_i} & -x'_{p_i} & 0 \end{pmatrix} \quad \mathbf{R_{m_i}} = \begin{pmatrix} 0 & -x'_{m_i} & -y'_{m_i} & -z'_{m_i} \\ x'_{m_i} & 0 & -z'_{m_i} & y'_{m_i} \\ y'_{m_i} & z'_{m_i} & 0 & -x'_{m_i} \\ z'_{m_i} & -y'_{m_i} & x'_{m_i} & 0 \end{pmatrix}. \tag{4.56}$$

Die Elemente der Matrix $\mathbf{N}$ lassen sich über die Punktkoordinaten berechnen:

$$\mathbf{N} = \begin{pmatrix} S_{xx} + S_{yy} + S_{zz} & S_{yz} - S_{zy} & S_{zx} - S_{xz} & S_{xy} - S_{yx} \\ S_{yz} - S_{zy} & S_{xx} - S_{yy} - S_{zz} & S_{xy} + S_{yx} & S_{zx} + S_{xz} \\ S_{zx} - S_{xz} & S_{xy} + S_{yx} & -S_{xx} + S_{yy} - S_{zz} & S_{yz} + S_{zy} \\ S_{xy} - S_{yx} & S_{zx} + S_{xz} & S_{yz} + S_{zy} & -S_{xx} - S_{yy} + S_{zz} \end{pmatrix} \tag{4.57}$$

$$S_{xx} = \sum_{i=1}^{n} x'_{p_i} x'_{m_i}, \quad S_{xy} = \sum_{i=1}^{n} x'_{p_i} y'_{m_i}, \ \ldots \ . \tag{4.58}$$

Das gesuchte Einheitsquaternion entspricht dem größten Eigenvektor $\dot{\mathbf{e}}_{\max}$, korrespondierend zu dem größten Eigenwert $\lambda_{\max}$:

$$[\mathbf{N} - \lambda_{\max} \mathbf{E}]\, \dot{\mathbf{e}}_{\max} = 0 \tag{4.59}$$

$$\dot{\mathbf{e}}_{\max} = \begin{pmatrix} q_0 \\ q_x \\ q_y \\ q_z \end{pmatrix}, \tag{4.60}$$

wobei E die Einheitsmatrix der Größe 4×4 repräsentiert. Für die Lösung des charakteristischen Polynoms 4. Grades sind geschlossene Lösungen verfügbar, z.B. die Methode nach Lodovico Ferrari (1522-1565). Der Eigenvektor $\dot{\mathbf{e}}_{\max}$ wird anschließend über die Lösung des homogenen Gleichungssystems aus Gleichung 4.59 ermittelt.

Mithilfe dieses Eigenvektors lässt sich die Rotationsmatrix R zwischen den Punktemengen $\mathbf{w'_{p_i}}$ und $\mathbf{w'_{m_i}}$ wie folgt bestimmen:

$$\mathbf{R} = \begin{pmatrix} q_0^2 + q_x^2 + q_y^2 - q_z^2 & 2(q_x q_y - q_0 q_z) & 2(q_x q_z + q_0 q_y) \\ 2(q_y q_x + q_0 q_z) & q_0^2 - q_x^2 + q_y^2 - q_z^2 & 2(q_y q_z - q_0 q_x) \\ 2(q_z q_x - q_0 q_y) & 2(q_z q_y + q_0 q_x) & q_0^2 - q_x^2 - q_y^2 + q_z^2 \end{pmatrix}. \tag{4.61}$$

Diese Vorgehensweise führt insgesamt zu einer geschlossenen Lösung des Gesamtproblems.

Zunächst ist die Methode noch sehr anfällig für Ausreißer, was auch mit der einmaligen Korrespondenzsuche zusammenhängt. Daher wird zunächst über eine Gewichtungs-

funktion das Verhalten eines M-Schätzers (engl.: M-Estimator) (Rey, 1983) mit einer *Fair*-Gewichtsfunktion nachgebildet, um den Einfluss von sehr weit entfernten Punkten auf die Optimierung zu reduzieren. Die Gewichtsfunktion $w_f(d^{(0)})$ bezieht sich dabei auf den ursprünglichen, euklidischen Abstand $d_i^{(0)}$ zwischen den Punkten $\mathbf{w_{p_i}}$ und $\mathbf{w_{m_i}}$:

$$w_f(d_i^{(0)}) = \frac{1}{1 + \frac{d_i^{(0)}}{c}}. \tag{4.62}$$

Diese Gewichtungsfunktion wird in die geschlossene Lösung nach Horn (1987b) eingebracht, wodurch sich die aufsummierte, paarweise Beziehung der mittelwertfreien Punkte wie folgt ergibt:

$$S_{xx} = \sum_{i=1}^{n} w_f(d_i^{(0)}) x'_{p_i} x'_{m_i}, \quad S_{xy} = \sum_{i=1}^{n} w_f(d_i^{(0)}) x'_{p_i} y'_{m_i}, \ \ldots \ . \tag{4.63}$$

Nachdem die Transformation in dieser neuen Form berechnet wurde, wird die Modellpose aktualisiert. Anschließend werden Punkte, die zu weit vom Modell entfernt sind aus der Punktewolke ausgeschlossen, um Ausreißer zu löschen (vgl. Kap. 4.4.4). Danach wird die Korrespondenzsuche erneut durchgeführt und die analytische Lösung nochmals bestimmt. Nach zwei bis drei Iterationen erhält man eine optimierte Punktewolke und eine verbesserte Objektpose. Die Gewichtung der Punktkorrespondenzen werden in jedem Iterationsschritt erneut berechnet. Dieses Vorgehen ist im Kontext des Bündelausgleichs in der Literatur als *Robust Reweighting* bekannt (Triggs et al., 2000).

Speziell für die Rotationswinkelschätzung ist die Gewichtungsfunktion unerlässlich, um genaue Ergebnisse zu erhalten. Die Robustheit insgesamt ist ähnlich hoch wie beim klassischen ICP mit Ausreißerbehandlung (siehe Abschnitt 4.4.4). Die Verarbeitung ist allerdings rund 40-mal schneller als die nichtlineare Lösung mittels ICP.

4.4.8. Modellbasierte Stereobildverarbeitung

In der Literatur werden zur Schätzung der 3D-Pose eines Objekts, definiert durch drei Translations- und drei Rotationsparameter, aus Stereobildsequenzen meist punktbasierte Verfahren wie der ICP-Algorithmus (siehe Kap. 2.1.7) verwendet, nachdem die Struktur der Szene mithilfe eines Stereobildverarbeitungsalgorithmus durch die Berechnung der 3D-Koordinaten einer Vielzahl von Punkten rekonstruiert wurden. Vorteilhaft bei diesen punktbasierten Verfahren ist die effiziente Berechnung der Pose. Außerdem können *schwache* Objektmodelle verwendet werden. Nachteilig ist jedoch die Entkopplung von den ursprünglichen Eingangsdaten (Bilddaten), wodurch nicht alle verfügbaren Informationen genutzt werden. Liefert der zuvor verwendete Stereoalgorithmus nur ungenaue, falsche oder wenige Punkte, so beeinflusst das die Genauigkeit der Poseschätzung direkt.

Neben den punktbasierten Pose-Estimation-Methoden wird daher in dieser Arbeit auch ein rein bildbasiertes Verfahren verwendet. Außerdem wird somit eine Redundanz bei der Poseschätzung dargestellt, was gerade bei Sicherheitsapplikationen wichtig ist.

In der Literatur sind sogenannte modellbasierte Stereobildverarbeitungsalgorithmen

(engl.: *Model-based Stereo*) seit mehreren Jahren bekannt. Sie unterscheiden sich grundlegend von der traditionellen Stereobildverarbeitung, da keine explizite Korrespondenzanalyse benötigt wird, um die beobachtete Szene zu rekonstruieren.

Ein früher Ansatz von Tonko und Nagel (2000) basiert auf einer direkten Assoziation von Kantenelementen im Bild mit Kantensegmenten auf einem detaillierten Objektmodell. Dieser Ansatz bedingt ein genaues Objektmodell und ausreichende Kanteninformationen im Bild, um eine eindeutige Poseschätzung durchführen zu können.

Der überwiegende Teil der Literatur beschäftigt sich nicht alleine mit dem Problem der Poseschätzung, sondern eher mit der Rekonstruktion bzw. der Modellanpassung, was eine implizite Poseschätzung bedeutet. So wird von Lee und Kunii (1993) ein Ansatz vorgestellt, um die menschliche Hand dreidimensional zu rekonstruieren, wobei gleichzeitig auch die Pose der Hand ermittelt wird. Das Modell ist dabei artikuliert, von der menschlichen Anatomie abgeleitet und besteht aus mehreren planaren, rigiden Elementen die zusammen eine kinematische Kette darstellen. Die Parameter des Modells werden unter Betrachtung der Ähnlichkeit zwischen der modellierten und der gesehenen Objektansicht in den Stereobildern angepasst. Alternativ dazu repräsentieren Heap und Hogg (1996) die gesamte Oberfläche der Hand durch ein deformierbares Modell. Solch ein deformierbares Modell wird auch im Applikationsszenario der 3D Gesichtsrekonstruktion von Amberg et al. (2007) verwendet. Adaptiert wird das Modell entsprechend der Objektansicht in den Stereobildpaaren, basierend auf Silhouetten, Farbunterschieden und manuell definierten Landmarken.

Die aktuell in der Literatur verfügbaren Verfahren zeichnen sich dadurch aus, dass sie entweder ein sehr detailliertes Modell bedingen oder ein zunächst schwaches Modell verwenden, welches über die Optimierung an die Form des realen Objekts angepasst wird. Für das Szenario der Fahrerassistenz für Kreuzungen im Straßenverkehr ist jedoch ausschließlich die Pose des Fahrzeugs wichtig, die genaue Form des Fahrzeugs kann außer Acht gelassen werden. Es ist sogar hinderlich, wenn die Form mit rekonstruiert wird, da dadurch der Optimierungsprozess wesentlich komplexer wird und auch an Robustheit einbüßt. Demnach sind in der Literatur derzeit keine Verfahren bekannt, die eine Poseschätzung mittels schwachem Modell auf Basis der modellbasierten Stereobildbverarbeitung erlauben.

Grundprinzip des neuen Verfahrens: Die Grundidee des Verfahrens ist, die Oberfläche des Objektmodells zusammen mit der Objektpose dazu zu verwenden, Homographien (Hartley und Zisserman, 2004) zwischen den beiden Stereobildern zu ermitteln, die korrespondierende Bildausschnitte beschreiben. Die Objektpose stellt also eine direkte Verknüpfung der beiden Stereobilder dar, wobei die Güte der Pose über den Vergleich der korrespondierenden Bildausschnitte ermittelt werden kann.

Um den Rechenaufwand für das Verfahren zu begrenzen, wird zunächst eine initiale Pose ermittelt (siehe Kap. 4.4.4). Anschließend werden auf der Oberfläche des 3D-Modells n virtuelle, äquidistante 3D-Punkte verteilt. Diese Punkte werden nun in das Bild der Masterkamera (im Folgenden wird die rechte Kamera als Masterkamera angenommen) I_r projiziert, in deren Umgebung kleine Bildteile ausgeschnitten und anschließend af-

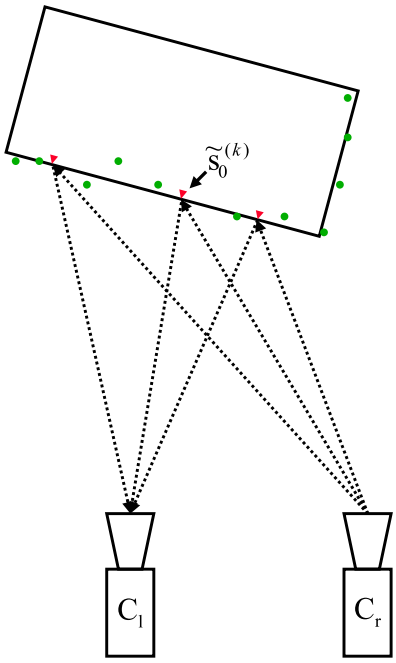


Abbildung 4.21.: Prinzip der modellbasierten Stereobildverarbeitung. Sehstrahlen der rechten Kamera C_r schneiden die Objektoberfläche und diese Schnittpunkte werden in das Bild der linken Kamera C_l projiziert.

fin transformiert, um eine frontoparallele Ansicht auf den korrespondierenden Teil der Objektoberfläche zu erhalten. Dabei wird angenommen, dass das Objekt aus einzelnen Flächen besteht, was für tessellierte Objekte immer zutrifft. Diese Bildausschnitte $s_r^{(i)}$ bleiben für das Bild der rechten Kamera über den gesamten Optimierungsprozess gleich, damit eine stabile Optimierung durchgeführt werden kann.

Um eine genauere Aussage über die Pose des Objekts zu erhalten, werden nun Sehstrahlen ausgehend von den berechneten Bildausschnitten berechnet, mit der Modelloberfläche geschnitten und diese Schnittpunkte in das zweite Kamerabild projiziert (siehe Abb. 4.21). Dort werden ebenfalls Bildteile um die projizierten Punkte ausgeschnitten und affin transformiert. Vergleicht man nun die beiden Bildausschnitte eines 3D-Punkts, erhält man eine Aussage über die Korrektheit der Modellpose. Optimiert man ein beliebiges Ähnlichkeitsmaß (z.B. Kreuzkorrelationskoeffizient, SSD oder SAD) zwischen den korrespondierenden Bildausschnitten über die Modellpose, so erhält man eine verbesserte Poseschätzung (Pose-Refinement). Es wird also eine Art modellbasierte, *inverse* Stereoberechnung durchgeführt. Zur Optimierung kann dabei auf einen beliebigen nichtlinearen Optimierungsansatz zurückgegriffen werden. Aufgrund experimenteller Untersuchungen wird das Gauß-Newton Verfahren verwendet.

Berechnung der Fehlerfunktion: Der projektive Vektor $\tilde{o}_r^{(i)}$ der Pixelkoordinaten des Bildausschnitts i im rechten Bild wird mit dem korrespondierenden projektiven Vektor $\tilde{o}_l^{(i)}$ im linken Bild durch die Homographie ${}_r^lH^{(i)}$ verknüpft. Diese Homographie wird durch die Modellebene definiert, auf welcher sich das Zentrum des Fensters k befindet.

Dadurch ergibt sich:

$$\tilde{o}_l^{(i)} = {}_r^lH^{(i)}\tilde{o}_r^{(i)} \tag{4.64}$$

mit $i = 1, \ldots, n$. Hartley und Zisserman (2004) erklären das Konzept der Homographien, welche durch Ebenen definiert sind, genauer. Zusammengefasst wird die Verknüpfung zwischen linkem und rechtem Bild über die Homographie wie folgt berechnet: Zunächst wird vom festgehaltenen Bildausschnitt $s_r^{(i)}$ im rechten Bild der Schnittpunkt zwischen dem korrespondierenden Sehstrahl vom Mittelpunkt $\tilde{o}_r^{(i)}$ aus mit der Modellebene berechnet. Dieser 3D-Punkt wird anschließend in das linke Kamerbild projiziert und um diesen Punkt $\tilde{o}_l^{(i)}$ herum wird wieder ein Bildausschnitt $s_l^{(i)}$ der gleichen, vordefinierten Größe extrahiert.

Die Fehlerfunktion E_{MBS}, die in dem modellbasierten Stereobildverarbeitungsalgorithmus minimiert wird, stellt den Vergleich zwischen den n Bildausschnitten aus dem rechten Kamerabild mit den n Bildausschnitten aus dem linken Kamerabild dar:

$$E_{\mathrm{MBS}} = \sum_{i=1}^{n} S\left(I_r(\tilde{o}_r^{(i)}),\, I_l(\tilde{o}_l^{(i)})\right), \tag{4.65}$$

wobei S das Vergleichsmaß definiert, $I_r(\tilde{o}_r^{(i)})$ den Bildausschnitt um den Punkt $\tilde{o}_r^{(k)}$ im rechten Bild und $I_l(\tilde{o}_l^{(i)})$ den korrespondierenden Bildausschnitt im linken Bild, ermittelt über die Homographie ${}_r^lH^{(i)}$. Dabei muss zusätzlich beachtet werden, dass jeweils eine frontoparallel Darstellung der Bildausschnitte verwendet wird. Zur Berechnung der Pose werden nur diejenigen Bildausschnitte benutzt, welche einen ausreichenden Kontrast beinhalten, was mithilfe eines Sobel Kantendetektors überprüft wird.

Vorteilhaft an diesem neu entwickelten Verfahren ist, dass es lediglich eine grobe Initialisierung der Pose benötigt, wodurch die Verwendung von „ungenauen“, aber schnellen Stereoverfahren ermöglicht wird. Das Verfahren bietet eine direkte Verknüpfung von 3D-Punkten und 2D-Bildinformationen, was eine Stabilisierung der Poseschätzung unterstützt. Das vorgestellte Verfahren basiert auf der affinen Transformation und Interpolation von kleinen Bildausschnitten (z.B. 15x15 Pixel) und dem Vergleich dieser Ausschnitte, was einen geringen Berechnungsaufwand mit sich bringt.

4.4.9. Fusion mit konturbasiertem Pose-Estimation-Ansatz

Eine Fusion mehrerer Algorithmen ist aus verschiedenen Gründen sinnvoll. Zum einen wird eine Redundanz geschaffen, wodurch der Ausfall eines Algorithmus abgefangen werden kann. Zum anderen werden bei richtiger Kombination die Nachteile eines Algorithmus durch die Eigenschaften des anderen ausgeglichen.

Die Algorithmen, die bei der Fusion eingesetzt werden, sollten möglichst orthogonale Eigenschaften haben. Daher wird für das Produktionsszenario eine Kombination aus

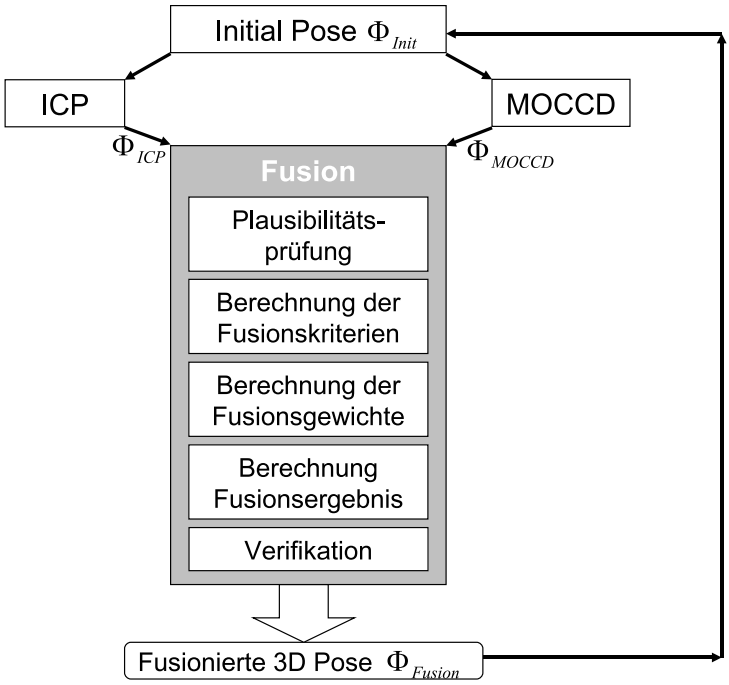


Abbildung 4.22.: Überblick über die Fusion.

punktbasierter Pose-Estimation und einem konturbasiertem Ansatz verwendet. Dabei hat die punktbasierte Poseschätzung grundsätzlich die Eigenschaft eine hohe Genauigkeit für die Entfernungsschätzung des Objekts von der Kamera bereitzustellen. Der konturbasierte Ansatz hingegen zeigt eine hohe Genauigkeit bei der Schätzung der Poseparameter parallel der Bildebene.

Die Struktur der Fusion ist in Abb. 4.22 dargestellt.

Prinzip der konturbasierten Pose-Estimation: Zur konturbasierten Pose-Estimation wird in dieser Arbeit der sogenannte *Multiocular Contracting Curve Density*-Algorithmus (MOCCD) verwendet, die multiokulare Erweiterung des *Contracting Curve Density*-Algorithmus (CCD).

Die multiokulare Erweiterung des von Hanek (2004) entwickelten Algorithmus wurde erstmals von Krüger (2007) erwähnt. Dort wurde der Algorithmus für Anwendungen in der industriellen Bildverarbeitung eingesetzt. Hahn et al. (2007) verwendeten den MOCCD erstmals für die Verfolgung der Hand-Unterarm-Region des Menschen in Bildsequenzen.

Der Algorithmus verwendet direkt die von der Stereokamera aufgenommenen Bilder in Verbindung mit den Kameraparametern. Die Kontur des hinterlegten 3D-Modells wird in die Bilder aller Kameras projiziert. Senkrecht dieser Kontur werden an mehreren Stellen die innere und äußere Grauwertstatistiken ermittelt, um anschließend unter Optimierung der 3D-Modellpose die bestmögliche Trennung der inneren und äußeren Pixelstatistik zu

suchen. Bei größtmöglichem Unterschied der Statistiken stimmen die Kante im Bild und die projizierte Objektkontur bestmöglich überein.

Der Vorteil dieses Algorithmus gegenüber ähnlichen Methoden (z.B. dem Chamfer-Matching von von Bank et al. (2003)) ist, dass die Kanten, an die die Kontur angepasst wird sehr robust gefunden werden, indem die Statistik der Grauwertverteilung in der Umgebung mit einbezogen wird. Bei anderen Verfahren werden oft feste Schwellen angenommen, die hier entfallen, wodurch hier auch sehr schwache Kanten im Bild als Merkmal verwendet werden können.

Funktionsweise der Fusion: Zunächst werden die intialen Poseparameter (siehe Kap. 4.4.3) für beide Algorithmen zum Start der jeweiligen Optimierung genutzt. Zur Verifikation und zur Bewertung der berechneten Pose werden drei verschiedene Qualitätskriterien benutzt: (i) die punktweise Distanz zum Modell, (ii) die Orientierungsähnlichkeit und (iii) die Ansichtsähnlichkeit.

Das erste Fusionskriterium ist die punktweise Distanz der segmentierten Punktewolke zum Modell in der jeweiligen Pose. Dazu wird eine dünne Hülle um das Objektmodell benutzt, um die Anzahl der Punkte innerhalb der Objekthülle zu ermitteln. Der Gewichtswert σ_p ist dann der Quotient aus den Punkten innerhalb der Hülle und allen segmentierten Punkten.

Die Orientierungsähnlichkeit σ_o wird berechnet, indem die Kontur des 3D-Modells extrahiert und in die Bilder des Kamerasystems projiziert wird. Um die Qualität der Kontur zu berechnen, läuft der Algorithmus entlang kleiner Senkrechten an jedem Kurvenpunkt. Für jedes Pixel auf der Senkrechten wird der Grauwertgradient berechnet. Die Orientierung des Gradienten, welche zwischen 0° und 180° liegt, wird dann mit der Orientierung der Modellkontur an dieser Stelle verglichen. Anschließend werden die Kurvenpunkte gezählt, welche mit einer kleinen Toleranz mit der jeweiligen Gradientenorientierung übereinstimmen. Die Anzahl wird über alle Kamerabilder und alle Kurvenpunkte normalisiert, sodass ein Qualitätsmaß im Bereich $[0 \dots 1]$ entsteht.

Die Ansichtsähnlichkeit σ_c für eine 3D-Pose zum Zeitpunkt t basiert auf dem Vergleich der aktuellen Objektansicht mit den Objektansichten zum Zeitpunkt t_{-1}. Unter Verwendung der aktuellen 3D-Pose und den bekannten Kameraparametern wird das Abbild des Unterarms in allen Kameras ausgeschnitten und in eine definierte Größe und Orientierung transformiert. Die endgültige Poseschätzung wird dazu benutzt, ein Referenzabbild des Unterarms in eine Datenbank abzulegen (siehe Abb. 4.23). Diese Datenbank beschreibt die Ansicht des zu verfolgenden Objekts in den letzten Zeitschritten. Die Ansichtähnlichkeit σ_c einer neuen Poseschätzung wird auf Basis der normierten Kreuzkorrelation zwischen der aktuellen Objektansicht zum Zeitpunkt t und zum Zeitpunkt t_{-1} berechnet. Da mehrere Kameras verwendet werden, wird der schlechteste Korrelationskoeffizient als Ähnlichkeitsmaß verwendet.

Das Fusionsergebnis wird unter Verwendung der beiden Poseschätzungen $\mathbf{\Phi}_{\mathrm{ICP}}$ und $\mathbf{\Phi}_{\mathrm{MOCCD}}$ und den drei beschriebenen Fusionskriterien berechnet. Die Fusionskriterien werden für beide Poseergebnisse berechnet. Anschließend werden die Kriterien zu einem Gewichtungsfaktor w_{ICP} für den ICP-Algorithmus und w_{MOCCD} für den MOCCD-

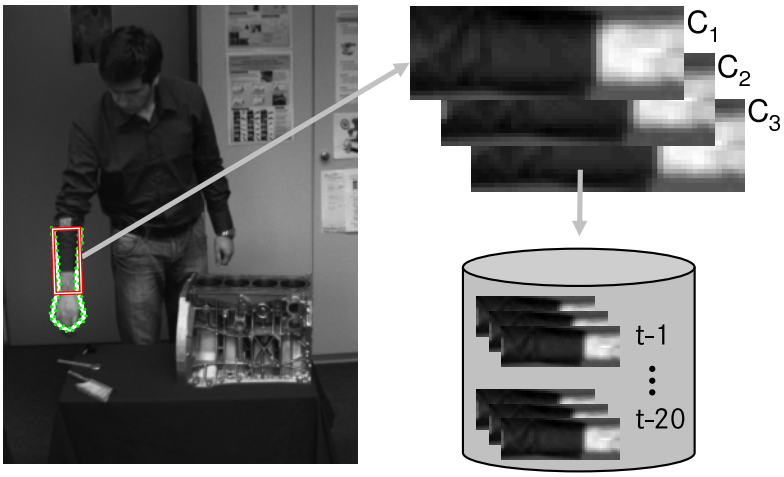


Abbildung 4.23.: Generierung der Referenzabbilder. Das Unterarm-Rechteck (rot) wird in allen Kameras ausgeschnitten und auf eine definierte Größe und Orientierung transformiert und interpoliert.

Algorithmus verrechnet:

$$w_{\text{ICP}} = \sigma_{p|\mathbf{\Phi}_{\text{ICP}}} + \sigma_{o|\mathbf{\Phi}_{\text{ICP}}} + \sigma_{c|\mathbf{\Phi}_{\text{ICP}}} \tag{4.66}$$

$$w_{\text{MOCCD}} = \sigma_{p|\mathbf{\Phi}_{\text{MOCCD}}} + \sigma_{o|\mathbf{\Phi}_{\text{MOCCD}}} + \sigma_{c|\mathbf{\Phi}_{\text{MOCCD}}} \cdot \tag{4.67}$$

Die beiden Poseupdates $\Delta\mathbf{\Phi}_{\text{ICP}}$ und $\Delta\mathbf{\Phi}_{\text{MOCCD}}$ der Pose-Estimation-Algorithmen werden dann zusammen mit den berechneten Gewichtungsfaktoren benutzt, um den fusionierten Poseparametervektor $\mathbf{\Phi}_{\text{fusion}}$ zu berechnen:

$$\mathbf{\Phi}_{\text{fusion}} = \mathbf{\Phi}_{\text{init}} + \frac{w_{\text{ICP}}}{w_{\text{ICP}} + w_{\text{MOCCD}}} \cdot \Delta\mathbf{\Phi}_{\text{ICP}} + \frac{w_{\text{MOCCD}}}{w_{\text{ICP}} + w_{\text{MOCCD}}} \cdot \Delta\mathbf{\Phi}_{\text{MOCCD}} \cdot \tag{4.68}$$

Der fusionierte Poseparametervektor wird abschließend unter Verwendung der Referenzabbilder der letzten Zeitschritte verifiziert, um Fehler auszuschließen und sicher zu stellen, dass weiterhin das interessierende Objekt verfolgt wird.

Dieser gesamte Ablauf wird mehrfach durchgeführt, indem das Fusionsergebnis jeweils zur Initialisierung einer erneuten Poseschätzung mittels beider Algorithmen verwendet wird, bis schließlich das Ergebnis der Fusion konvergiert (siehe Abb. 4.24).

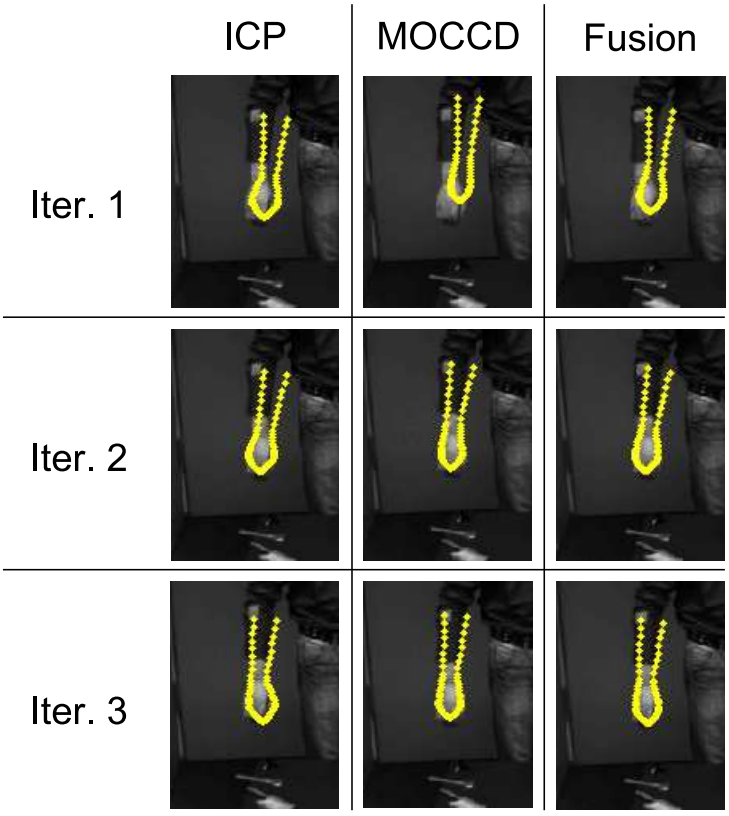


Abbildung 4.24.: Konvergenzverhalten der ICP-MOCCD Fusion. Die erste Spalte zeigt das ICP-Ergebnis für jede Iteration und die zweite Spalte das jeweilige MOCCD-Ergebnis. In der letzten Spalte wird das Fusionsergebnis für jede Iteration veranschaulicht, was wiederum auch die Initialisierung für die nächste Iteration ist.

4.5. Bewegungsanalyse

4.5.1. Vollständige Bewegungschätzung auf Basis von Verschiebungsvektorfeldern

In der vorliegenden Arbeit werden verschiedene Arten der optischen Flussberechnung verwendet. Dabei werden wahlweise nur eindimensionale Flussvektoren (Spacetime-Stereo), zweidimensionale Flussvektoren aus spärlicher Flussberechnung (Census-Fluss) oder zweidimensionale Flussvektoren aus dichter Flussberechnung (TV-L^1-Fluss) betrachtet (siehe Kap. 4.2). Ziel ist es, unabhängig vom verwendeten Flussverfahren eine robuste und genaue Schätzung für die vollständige Objektbewegung zu erhalten.

Hauptaugenmerk liegt dabei auf den Mehrdeutigkeiten, die durch das Apertur-Problem verursacht werden (siehe Kap. 2.1.2). Dieses Problem tritt meist bei spärlichen Verfahren auf, ist aber auch bei dichten, global arbeitenden Verfahren nicht ausgeschlos-

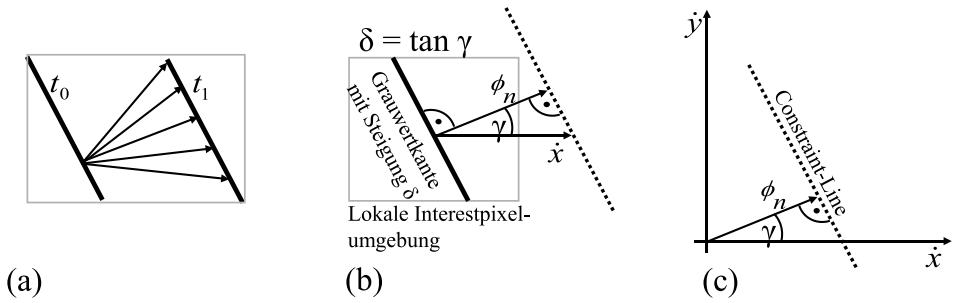


Abbildung 4.25.: (a) Grauwertkante in zwei aufeinanderfolgenden Zeitschritten und die möglichen Bewegungsvektoren für diese Verschiebung. Die Constraint-Line stellt alle möglichen Bewegungsvektoren in einer Linie dar. (b) Beziehung zwischen Kantenrichtung δ und Normalfluss ϕ_n. (b) Definition der Constrain-Line im $\dot{x}\dot{y}$-Raum entsprechend der Konfiguration $(\dot{x}, \dot{y})$ die konsistent mit dem ermittelten Normalfluss ϕ_n ist.

sen. Im Produktionsumfeld tritt zudem das Problem auf, dass spärliche Flussverfahren meist auf Basis von Ecken im Bild arbeiten. Die menschliche Hand-Unterarm-Region weist jedoch nur sehr wenige Ecken im Bild auf. Daher ist hier nur das Spacetime-Stereo oder ein globales Verfahren geeignet.

Ziel ist es, ein Verfahren zur Bestimmung der Objektbewegung zu entwickeln, dass unabhängig vom darunterliegenden Fluss-Verfahren arbeitet und unter Berücksichtigung des Apertur-Problems die vollständige Objektbewegung ermittelt.

In der Literatur sind dafür verschiedene Verfahren bekannt, die jedoch meist auf zweidimensionalen Flussvektoren aufbauen. Für die Verwendung mit eindimensionalen Flussvektoren in Verbindung mit der Kantenorientierung lässt sich mithilfe des Constraint-Line-Ansatzes nach Schunck (1986) die Objektbewegung ermitteln. Nachteilig an diesem Verfahren ist jedoch, dass mit dem ursprünglichen Verfahren lediglich translatorische Bewegungen geschätzt werden können.

Beide Bewegungskomponenten eines 3D-Punkts parallel der Bildebene können nur ermittelt werden, wenn die Umgebung eine Ecke im Bild darstellt. An Kanten hingegen kann nur eine Geschwindigkeitskomponente sicher ermittelt werden, wie beispielsweise beim Spacetime-Stereo.

Grundprinzip: Aus der berechneten Geschwindigkeitskomponente und der Kantenorientierung im Bild lässt sich ein sogenannter Normalfluss ϕ_n berechnen (siehe Abb. 4.25). Der Winkel γ zwischen der Richtung der horizontalen Epipolarlinie und der Richtung des Normalflusses is gegeben durch $\delta = \tan\gamma$ mit der Steigung δ wie in Abschnitt 4.2.1 definiert.

Nachfolgend werden die translatorischen Geschwindigkeitskomponenten parallel der x, y und z Achse als $\dot{x}_{\text{obj}}$, $\dot{y}_{\text{obj}}$ und $\dot{z}_{\text{obj}}$ in Metern pro Sekunde bezeichnet. Bei gegebenem Normalfluss ϕ_n werden alle möglichen Bewegungen $(\dot{x}, \dot{y})$ durch die korrespondierende Constraint-Line im $\dot{x}\dot{y}$-Raum beschrieben (siehe Abb. 4.25b). Für ein Objekt, welches

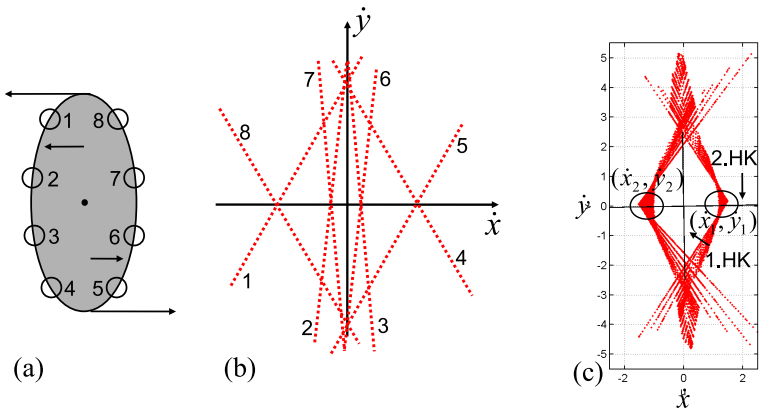


Abbildung 4.26.: (a) Um die Hochachse rotierende Ellipse mit Referenzpunkten auf der Kontur. (b) Resultierende Constraint-Lines der Rotation der Ellipse. (c) Typische Verteilung der Schnittpunkte im $\dot{x}\dot{y}$-Raum für eine reale Testsequenz. Der Mittelwert der Verteilung wurde bereits von allen Punkten subtrahiert, die erste (1.HK) und zweite (2.HK) Hauptkomponente der Verteilung sind als schwarze Linien gekennzeichnet.

sich rein translatorisch bewegt, schneiden sich alle Constraint-Lines von Punkten die zu dem Objekt gehören in einem einzigen Punkt im $\dot{x}\dot{y}$-Raum. Somit ist die Translation eindeutig bestimmt. Für rotatorische Bewegungen bzw. für kombinierte rotatorische und translatorische Bewegungen, entsteht eine ausgedehnte Menge von Schnittpunkten der Constraint-Lines im $\dot{x}\dot{y}$-Raum. Dieser Fall wird jedoch von Schunck (1986) nicht weiter behandelt. Ein Beispiel für eine sich gegen den Uhrzeigersinn, um die Hochachse rotierende Ellipse wird in Abb. 4.26 gezeigt.

Die $\dot{x}$-Koordinaten der Schnittpunkte der Constraint-Lines sind in diesem Beispiel ein Maß für die mittlere horizontale Geschwindigkeit des korrespondierenden Paars von Bildpunkten. Die $\dot{y}$-Koordinaten hingegen haben keine physikalische Bedeutung. Die Schnittpunkteverteilung ist vertikal ausgedehnt, da ein vertikaler Kantendetektor verwendet wurde. Außerdem wurden nur Kanten mit einer Steigung von $|\delta| < \delta_{\max}$ mit $\delta_{\max}$ typischerweise zwischen 1 und 2 betrachtet. Dadurch werden horizontale Kanten außer Acht gelassen.

Abb. 4.26c zeigt die Verteilung der Schnittpunkte aus einer Hand-Unterarm-Testsequenz. Diese Verteilung ist typisch für ein rotationssymmetrisches, längliches Objekt. Die Punkte im $\dot{x}\dot{y}$-Raum werden entlang der Hand-Unterarm-Achse gewichtet, um möglichst jeden Bereich in den weiteren Berechnungen gleich zu gewichten. Die mittlere, translatorische Bewegung $(\dot{x}_{\text{obj}}, \dot{y}_{\text{obj}})$ wurde bereits von den Schnittpunkten in Abb. 4.26c subtrahiert.

In dem Beispiel in Abb. 4.26c bewegen sich Punkte am Handgelenk schneller im Bild als die Punkte am Ellbogen. Die resultierenden Schnittpunkte konzentrieren sich stark an den Punkten $(\dot{x}_1, \dot{y}_1)$ und $(\dot{x}_2, \dot{y}_2)$, welche die Bewegung des Handgelenks und des Ell-

bogens repräsentieren (siehe Abb. 4.26c). Werden beispielsweise im Produktionsszenario zwei kreisrunde Marker am oberen und unteren Ende des Unterarms angebracht, würden zwei Cluster von Schnittpunkten im $\dot{x}\dot{y}$-Raum an den Stellen $(\dot{x}_1, \dot{y}_1)$ und $(\dot{x}_2, \dot{y}_2)$ entstehen. Punkte, die zwischen diesen Markerpositionen liegen, liefern eine Verteilung die symmetrisch ist unter Berücksichtigung der Linie, die die Punkte $(\dot{x}_1, \dot{y}_1)$ und $(\dot{x}_2, \dot{y}_2)$ verbindet. Die Information über den rotatorischen Bewegungsanteil ist damit in dem Bereich Δv enthalten. Dieser Bereich wird beschrieben über die Projektion der Schnittpunkte auf die Hauptkomponente senkrecht der longitudinalen Achse des Objekts (die zweite Hauptkomponente in Abbilldung 4.26c). Der Wert von Δg korrespondiert dann mit der Geschwindigkeitsdispersion über das Objekt, welche durch die rotatorische Bewegung in der Bildebene verursacht wird. Im vorgestellten System wird Δg robust über das 10% und 90% Quantil der Verteilung der projizierten Schnittpunkte ermittelt. Die Winkelgeschwindigkeit ω_p der Objektrotation parallel der Bildebene wird dann durch $\omega_p = \Delta g / \Delta l$ mit Δl als Längenintervall parallel zur longitudinalen Objektachse beschrieben.

Tiefengeschwindigkeitskomponente: Für Szenenpunkte, die mittels Spacetime-Stereo berechnet wurden, sind zusätzliche Informationen über die Tiefengeschwindigkeit verfügbar. Diese können ebenfalls ausgewertet werden, um eine vollständige Bewegung des Objekts zu erhalten.

Die translatorische Bewegungskomponente $\dot{z}_{\text{obj}}$ parallel der z Achse ist gegeben durch den Median der (deutlich rauschbehafteten) Werte von $\partial z / \partial t$ aller dem Objekt zugewiesenen Szenenpunkte.

Die Rotation orthogonale zur Bildebene wird ebenfalls über die Werte $\partial z / \partial t$ bestimmt, wobei jeder rigide Objektteil einzeln betrachtet wird. Dazu wird zunächst die Projektion $w_z^{(i)}$ des 3D-Punkts w_i auf die longitudinale Objektachse berechnet und eine Regressionslinie an die $\left(w_z^{(i)}, \partial z / \partial t^{(i)}\right)$ Datenpunkte angepasst. Die Steigung der Regressionslinie liefert direkt die Geschwindigkeitsdispersion $\Delta \dot{z}$ in z-Richtung und damit die Winkelgeschwindigkeit ω_o der Objektrotation orthogonal zur Bildebene. Bei rotationssymmetrischen Objektmodellen ist damit die vollständige, rotatorische Bewegung durch die beiden Komponenten ω_p und ω_o bestimmt.

Die in diesem Abschnitt beschriebene Methode zur Bewegungsschätzung liefert direkt die vollständige zeitliche Ableitung $\dot{\mathbf{\Phi}}$ der Objektpose $\mathbf{\Phi}$, ohne dazu eine zeitliche Filterung benutzen zu müssen (siehe Kap. 2.1.8).

4.5.2. Ergänzung der Bewegungsanalyse durch den ShapeFlow-Algorithmus

Wird die beobachtete Szene durch ein Stereoverfahren in Kombination mit einem traditionellen Flussverfahren raum-zeitlich rekonstruiert, so stehen keine Informationen über die Tiefengeschwindigkeit zur Verfügung. Abhilfe schafft das erwähnte Spacetime-Stereo, wobei dieses sich nicht für alle Szenen eignet, da die Rekonstruktion im Vergleich zu anderen Stereobildverarbeitungsmethoden eher spärlich ist.

Eine andere Möglichkeit besteht in der zusätzlichen Verwendung des ShapeFlow-Algorithmus (Hahn et al., 2008b). Dieser konturbasierte Ansatz stellt die zeitliche Erweiterung des MOCCD-Algorithmus dar (siehe Abschnitt 4.4.9). Dabei wird ein raumzeitliches Konturmodell an Bilder aus verschiedenen Zeitschritten des multiokularen Kamerasystems angepasst. Dabei wird, wie auch beim MOCCD-Algorithmus versucht, die Statistiken der Grauwertverteilungen innerhalb und außerhalb der Modellkontur entlang kurzer Senkrechten bestmöglich zu trennen, um die Modellkontur so an die Grauwertkontur in den Bildern anzupassen.

Da diese Anpassung in allen Bildern des Kamerasystems und über mehrere Zeitschritte erfolgt, ist die Schätzung aller Bewegungsparameter inklusive der Tiefengeschwindigkeit, trotz der Verwendung traditioneller Verfahren zur optischen Fluss-Berechnung, möglich. Der Konvergenzradius des Verfahrens ist begrenzt, weshalb eine gute Initilisierung wichtig ist. Bei ausschließlicher Verwendung des ShapeFlow-Alorithmus wird daher ein Multihypothesenansatz auf Basis von Kalman-Filtern mit verschiedenen Bewegungsmodellen benutzt. Wird jedoch die in Abschnitt 4.5.1 erwähnte Bewegungsanalyse auf Basis des erweiterten Constraint-Line-Ansatzes zur Initialisierung verwendet, kann dieser Multihypothesenansatz ausbleiben. Die Bewegungsanalyse initialisiert den ShapeFlow sehr gut, nur dass die Tiefengeschwindigkeitskomponente bei Verwendung traditioneller Verfahren zur optischen Fluss-Berechnung mit Null initialisiert wird.

4.5.3. Modelbasierter Szenenfluss

Verfahren zur Analyse der Bewegung von Objekten basieren in der Literatur meist auf zeitlicher Filterung. Nachteilig ist dabei, wie in Kap. 2.1.8 erwähnt, die Trägheit eines solchen Filters und die Bedingung, dass für das beobachtete Objekt ein möglichst realisitisches Bewegungsmodell zur Verfügung stehen muss. Außerdem sind auch Verfahren bekannt, die instantan die Objektbewegung schätzen und daher nicht diese Nachteile aufweisen. Hahn et al. (2008b) stellen ein konturbasiertes Verfahren vor, welches die Objektbewegung aus drei Zeitschritten extrahiert. Dieses Verfahren bedingt ein gutes Vorwissen über die zu erwartende 2D-Kontur des 3D-Objekts im Bild. Ist dieses Vorwissen vorhanden, können Bewegungen in allen Richtungen erkannt werden, auch die anspruchsvolle Schätzung der Tiefengeschwindigkeit gelingt mit diesem Verfahren. Ist die Kontur des Objekts nicht ausreichend genau bekannt und kein Bewegungsmodell vorhanden, wie beispielsweise wenn beliebige Fahrzeuge im Kreuzungsbereich beobachtet werden, so bietet die Literatur derzeit kein Verfahren, mit dem man die vollständige Objektbewegung instantan schätzen kann.

Ein weiteres Problem stellt sich bei der Ermittlung der Bewegungsinformation parallel der optischen Achse des Kamerasystems. Bekannte Verfahren benutzen nur die 3D-Punktinformationen aus den beiden zeitgleich aufgenommenen Stereobildern und zusätzlich optische Fluss-Informationen als 2D-Bewegungsvektoren parallel zur Bildebene der entsprechenden Kamera, ermittelt aus zwei aufeinanderfolgenden Bildern einer Kamera. Aus diesen Informationen können maximal fünfdimensionale Informationen gewonnen werden, dreidimensionale Koordianten des Punkts und zweidimensionale Bewegungsinformationen orthogonal der optischen Achse des Kamerasystems. Die Be-

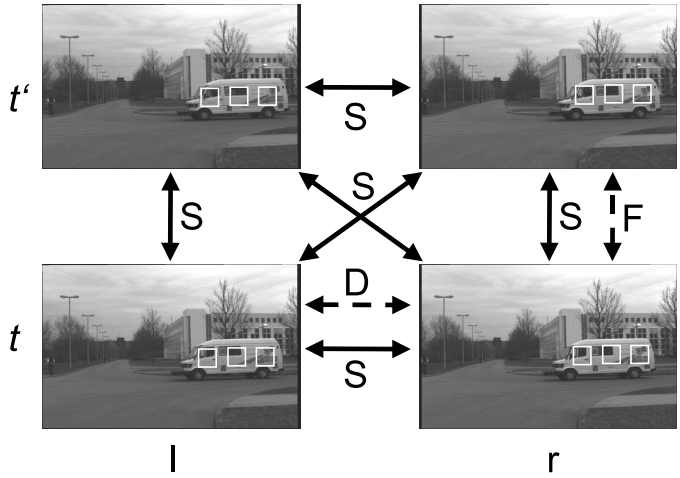


Abbildung 4.27.: Zusammenhang zwischen den vier Bildern einer Stereokamera in zwei Zeitschritten. Ermittlung des optischen Flusses (F), der Disparität (D), und des Szenenflusses (S).

wegungsinformation parallel zur optischen Achse des Kamerasystems ist so jedoch nicht ermittelbar. Franke et al. (2005) ermitteln mittels einer Vielzahl von Kalman-Filtern diese Geschwindigkeitskomponente über zeitliche Filterung der 3D-Position, was zu einem gewissen Verzögerungsverhalten führt. Außerdem werden die vorhandene Eingangsinformationen, d.h. vier Bilder die zu zwei Zeitpunkten mit einem binokularen Kamerasystem aufgenommen werden, nicht vollständig genutzt, da Stereo und optischer Fluss jeweils zwei Bilder benötigen, wobei eines von beiden Algorithmen benutzt wird, in Summe werden also drei benutzt. Es ist grundsätzlich möglich, die komplette 3D-Position und die 3D-Bewegung aus den vier gegebenen Bildern zu berechnen.

Verfahren, die alle vier Bilder benutzen und für jeden Punkt im Bild einen 3D-Punkt mit dazugehörigem 3D-Bewegungsvektor ermitteln, werden in der Literatur als Szenenflussalgorithmen bezeichnet (vgl. Kap. 2.1.4). Diese Algorithmen basieren auf einer globalen Optimierung über die ganze Szene, wobei iterativ die gewünschten Informationen extrahiert werden. Da dieser Vorgang für alle Bildpunkte unter Berücksichtigung aller vier Bilder durchgeführt wird, ist er rechenzeitintensiv und eignet sich somit nicht für die angestrebten Applikationen.

Grundprinzip: In diesem Abschnitt wird die neuartige zeitliche Erweiterung des Konzepts der modellbasierten Stereobildverarbeitung (siehe Kap. 4.4.8) beschrieben, der sogenannte modellbasierte Szenenfluss. Durch die Erweiterung soll es möglich sein, zusätzlich zur Pose auch die Objektbewegung instantan zu schätzen. Dafür werden vier Bilder, die Stereobilder des aktuellen und des letzten Zeitschritts benutzt, wobei Abb. 4.27 den Zusammenhang zwischen den beiden Stereobildpaaren zeigt. Eine zeitliche Filterung durch die Verwendung von Daten aus mehr als zwei Zeitschritten, wird nicht verwendet. Zur Verknüpfung der vier Bilder wird die Objektpose und deren zeitliche Ab-

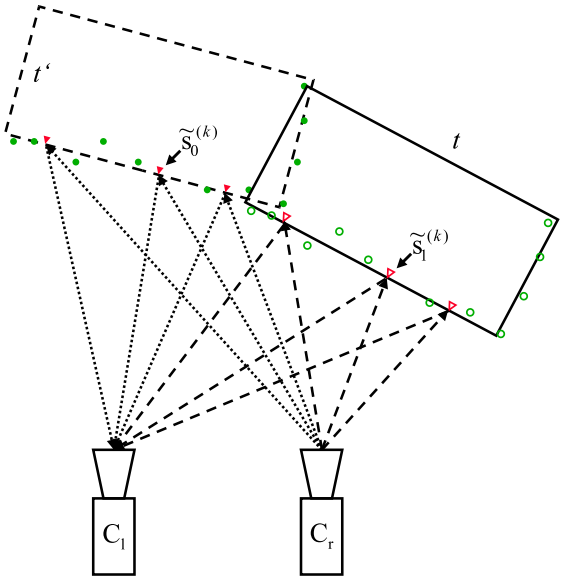


Abbildung 4.28.: Prinizp des modellbasierten Szenenflusses. Kreise: 3D-Punkte, Dreiecke: korrespondierende Punkte auf der Modelloberfläche.

leitung, die Objektbewegung zusammen mit der Modelloberfläche benutzt, um daraus Homographien zu berechnen. Diese Homographien beschreiben bei korrekter Pose- bzw. Bewegungsschätzung korrespondierende Bildpunkte in allen vier Bildern.

Initialisierung: Zur Detektion des Objekts und zur groben Initialisierung der Objektpose werden zunächst Szenenflussdaten verwendet, wie beispielsweise die Daten aus der Kombination Census-Stereo und Census-Fluss (siehe Abschnitt 4.2.2), die jedoch keine hohe Genauigkeit aufweisen müssen, d.h. die pixelgenaue Bestimmung des optischen Flusses ist ausreichend. Nach der Poseinitialisierung (siehe Kap. 4.4.4), wird der Median der zum Objekt gehörenden optischen Fluss-Vektoren dazu verwendet, eine Initialisierung für die Objektbewegungsschätzung zu ermitteln. Die Geschwindigkeitskomponente $\dot{z}$ entlang der Tiefenachse ist aus den vorhandenen Szenenflussdaten nicht abzuleiten und wird daher mit null initialisiert.

Berechnung der Fehlerfunktion: Wie bereits erwähnt, werden die Stereobilder (I_l und I_r) aus zwei aufeinanderfolgenden Zeitschritten (aktuell t und vorheriger Zeitschritt t') verwendet, wodurch vier Bilder zur Verfügung stehen: $I_{lt'}$, $I_{rt'}$, I_{lt} und I_{rt}. Die Homographie ${}^{lt'}_{rt}H^{(k)}$ verknüpft die beiden Bildausschnitte $s_{rt}^{(k)}$ und $s_{lt'}^{(k)}$, und hängt von den intrinsischen und extrinsischen Kameraparametern ab, welche als bekannt angenommen werden (Hartley und Zisserman, 2004). Außerdem ist die Homographie abhängig von der Orientierung der entsprechenden Modellebene zum Zeitpunkt t und t' auf der sich der Bildausschnitt k befindet, was direkt von den 3D-Poseparametern des Objekts und dessen Bewegung beeinflusst wird.

Die Ermittlung der Homographie geschieht nach folgendem Schema, ähnlich wie bei der modellbasierten Stereobildverarbeitung (Abschnitt 4.4.8): Ausgehend von den k festgehaltenen Bildausschnitten im rechten Bild des aktuellen Zeitschritts werden die Schnittpunkte des entprechenden Sehstrahls mit der Modellebene berechnet, wobei die Annahme, dass das Modell aus Ebenen besteht für tesselierte Objekte immer zutrifft. Wird die Homographie für den Zusammenhang zwischen zwei Bildern desselben Zeitschritts berechnet, wird dieser ermittelte 3D-Schnittpunkt einfach in das andere Bild projiziert. Andernfalls muss zunächst der 3D-Schnittpunkt entsprechend der aktuell angenommenen Objektbewegung von einem in den anderen Zeitschritt transformiert werden. Nach dieser Transformation wird der 3D-Schnittpunkt ebenfalls in das andere Bild projiziert. Dadurch kann zu jedem Bildausschnitt ein korrespondierender Bildausschnitt in einem beliebigen anderen Bild gefunden werden. Der Zusammenhang stellt sich über die ermittelte Homographie dann wie folgt dar:

$$\tilde{o}_{i't'}^{(k)} = {}_{it}^{i't'}H^{(k)}\tilde{o}_{it}^{(k)}, \tag{4.69}$$

wobei $\tilde{o}_{it}^{(k)}$ den projektiven Vektor des Mittelpunkts des Bildausschnitts $s_{it}^{(k)}$ beschreibt.

Über die Homographien $H^{(k)}$ lassen sich in allen vier Bildern k Bildausschnitte definierter Größe (z.B. 15×15 Pixel) generieren, die anschließend miteinander verglichen werden. Somit werden zusätzlich zum Vergleich im aktuellen Zeitschritt (siehe modellbasierte Stereobildverarbeitung) auch die anderen fünf möglichen Bildkombinationen berücksichtigt. Demnach erhält man für die modellbasierte Schätzung des Szenenflusses folgende Fehlerfunktion:

$$\begin{aligned} E_{\mathrm{MSF}} = \quad & S \quad \left(I_{lt}(\tilde{o}_{lt}^{(k)}), I_{rt}(\tilde{o}_{rt}^{(k)})\right) + S\left(I_{lt'}(\tilde{o}_{lt'}^{(k)}), I_{rt'}(\tilde{o}_{rt'}^{(k)})\right) + S\left(I_{lt}(\tilde{o}_{lt}^{(k)}), I_{lt'}(\tilde{o}_{lt'}^{(k)})\right) + \\ & S \quad \left(I_{rt}(\tilde{o}_{rt}^{(k)}), I_{rt'}(\tilde{o}_{rt'}^{(k)})\right) + S\left(I_{lt}(\tilde{o}_{lt}^{(k)}), I_{rt'}(\tilde{o}_{rt'}^{(k)})\right) + S\left(I_{lt'}(\tilde{o}_{lt'}^{(k)}), I_{rt}(\tilde{o}_{rt}^{(k)})\right), \end{aligned}$$

wobei S das Ähnlichkeitsmaß definiert und $I_{rt}(\tilde{o}_{rt}^{(k)})$ den Bildausschnitt um den Punkt $\tilde{o}_{rt}^{(k)}$ im rechten Bild zum Zeitpunkt t. Eine affine Transformation wird, wie bei der modellbasierten Stereobildverarbeitung, wegen der verschiedenen Perspektive der Kameras benötigt.

Zur Berechnung der Ähnlichkeit zwischen den Bildausschnitten können veschiedene in der Literatur bekannte Verfahren eingesetzt werden. Auch hier wird wieder ein Sobel Kantendetektor benutzt, um nur Bildausschnitte mit einem hinreichend hohen Kontrast in die Berechnung einzubeziehen. Auch die Verwendung des Gauss-Newton-Algorithmus zur Minimierung der Fehlerfunktion E_{MSF} erfolgt wie bei der modellbasierten Stereobildverarbeitung.

Theoretisch ist es möglich, die Pose und die Bewegung eines Objekts gleichzeitig innerhalb einer Optimierung über die Fehlerfunktion E_{MSF} zu schätzen. In Experimenten hat sich jedoch gezeigt, dass eine konsekutive Optimierung hier sinnvoller ist. Es wird also zunächst die Objektpose im Zeitschritt t optimiert und danach getrennt deren zeitliche Ableitung.

Eigenschaften des Verfahrens: Gegenüber Methoden aus dem Stand der Forschung werden alle vier Bilder (jeweils zwei zum Zeitpunkt t und zwei zum Zeitpunkt t') genutzt, um möglichst gut Pose und Bewegung bestimmen zu können. Ergebnis ist dann sowohl die verfeinerte Pose des Objekts, als auch die vollständige Bewegungsinformation des Objekts.

Vorteilhaft an dem Verfahren ist der Wegfall der Korrespondenzanalyse wie bei traditionellen Stereobildverarbeitungsalgorithmen. Außerdem ist zur Schätzung der Bewegung keine zeitliche Filterung notwendig, wodurch auch hochdynamische Szenen analysiert werden können.

Nachteilig ist hingegen, dass das beobachtete Objekt eine bestimmte Größe in den Bildern haben muss, da sonst die Poseschätzung zu fehleranfällig ist. Durch diese Bedingung ist ein Einsatz des Verfahrens im Produktionsszenario, wo das beobachtete Objekt nur wenige Pixel breit ist, nicht möglich.

Aus der ermittelten Pose und Bewegung des Objekts bzw. dessen Modell kann abschließend ein dichter Szenenfluss generiert werden, der für jeden beliebigen Punkt auf dem Modell eine Aussage über die 3D-Position und die 3D-Bewegung im Raum bereitstellt.

5. Anwendungsbezogene Systemausprägungen

Aus den im vorhergehenden Kapitel beschriebenen Komponenten wird je nach Anforderung, die sich aus der Aufgabenstellung und dem jeweiligen Szenario ergibt, ein Gesamtsystem zusammengestellt. Dabei sind die beiden Szenarien Produktion und Straßenverkehr sehr unterschiedlich und bedingen daher verschiedene Systemausprägungen, um eine bestmögliche Genauigkeit bei der Analyse der Position, Orientierung und Bewegung von Objekten zu erreichen.

Die Systeme sind stets derart aufgebaut, dass zwischen einzelnen Verarbeitungsstufen Rückkopplungen eingesetzt werden können. Dadurch erhöht sich die Gesamtrobustheit des Systems, da aufeinanderfolgende Algorithmen die Vorhergehenden mit wichtigen Informationen versorgen.

Gemein haben alle vier Systemausprägungen, dass sie auf Szenenflusspunkten basieren, die anschließend geclustert werden und dann die Objektpose unter Verwendung des ICP-Algorithmus ermittelt wird. Wie auch in den Experimenten in Teil III gezeigt wird, ist dies ein genereller Ansatz, der für die beiden sehr unterschiedlichen und gleichzeitig anspruchsvollen Szenarien Produktion und Straßenverkehr gleichermassen geeignet ist.

5.1. Produktionsszenario

Das Produktionsszenario zeichnet sich dadurch aus, dass das beobachtete Objekt (die Hand-Unterarm-Region des Arbeiters) im Bild relativ klein abgebildet wird, jedoch eine hohe Geschwindigkeit verbunden mit schnellen Bewegungsänderungen erreicht. Die Hintergründe in den Szenen sind meist stark strukturiert. Eine zusätzliche Herausforderung besteht darin, lediglich den Hand-Unterarm-Bereich einer Person zu verfolgen, da dies der in erster Linie gefährdete und daher interessierende Bereich des Arbeiters ist. Ein komplettes Oberkörpermodell ist hingegen wesentlich komplexer mit einer weitaus höheren Anzahl interner Freiheitsgrade (Schmidt et al., 2006) und daher nicht zielführend. Jedoch ist bei der Betrachtung lediglich eines Teils der Person die Segmentierung auf Basis von 3D-Punktewolken recht anspruchsvoll.

5.1.1. Systemausprägung 1: Raum-zeitliche Poseschätzung auf Basis von 3D-Punktewolken und Verschiebungsvektorfeldern

Zur Verfolgung der Hand-Unterarm-Region des Arbeiters in multiokularen Bildsequenzen wird ein System bestehend aus Spacetime-Stereo (Kap. 4.2.1), graphenbasiertem Clustering (Kap. 4.3), ICP-Algorithmus (Kap. 4.4.4) und Bewegungsanalyse auf Basis

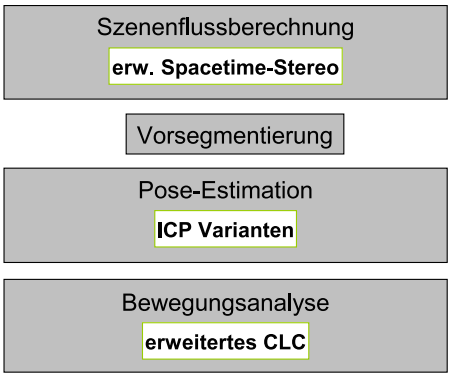


Abbildung 5.1.: Komponenten der Systemausprägung 1.

von Verschiebungsvektorfeldern (Kap. 4.5.1) verwendet. Abb. 5.1 zeigt die Komponenten der Systemausprägung 1.

Vorteilhaft an dieser Systemkonfiguration ist, dass die einzelnen Komponenten von den Ergebnissen sowohl der vorhergehenden als auch der nachfolgenden Komponenten profitieren. So wird die Auswahl der objektspezifischen 3D-Punkte, die durch das Clusterverfahren und die Clusterauswahl generiert wird, nachträglich in der modellbasierten Pose-Estimation modifiziert und somit verfeinert. Die Ergebnisse der Bewegungsschätzung werden wiederum für die Poseinitialisierung und somit als Eingang für die Poseschätzung des nächsten Zeitschritts benutzt. Auch wird die Clusterauswahl über die Pose und die Bewegung, die im letzten Zeitschritt ermittelt wurde gesteuert. Dadurch ergibt sich ein sehr robustes System zur Poseschätzung, wie die Experimente in Abschnitt 8.1 zeigen.

Weiterhin ist die vollständige Schätzung der Pose und deren zeitliche Ableitung, die Bewegung, hervorzuheben. Andere Verfahren können auf Basis eines Verschiebungsvektorfeldes nur Bewegungen orthogonal zur optischen Achse ermitteln. Hier sorgt jedoch das Spacetime-Stereo dafür, dass auch Bewegungsinformationen parallel zur optischen Achse verfügbar sind und somit für rotationssymmetrische Objekte die vollständige Bewegungsinformation ermittelt werden kann. Dazu werden für die beiden rigiden Objektteile Hand und Unterarm die 5 Translations- und Rotationsparameter, die die Pose eines rotationssymmetrischen Objekts im 3D-Raum repräsentieren, zunächst unabhängig voneinander bestimmt. Anschließend werden die Objektteile durch Anpassung der Translationsparameter beider wieder miteinander verknüpft. Dabei gilt die Bedingung, dass die beiden Objektteile am Punkt P_1 verbunden sein müssen (siehe Kap. 4.4.1).

Das Resultat der Berechnungen eines Zeitschritts sind somit die Pose:

$$\mathbf{\Phi} = [P_{0_x}, P_{0_y}, P_{0_z}, \alpha_1, \beta_1, \alpha_2, \beta_2] \tag{5.1}$$

und die Bewegung als zeitliche Ableitung der Pose:

$$\dot{\mathbf{\Phi}} = \frac{\Delta \mathbf{\Phi}}{\Delta t} \ . \tag{5.2}$$

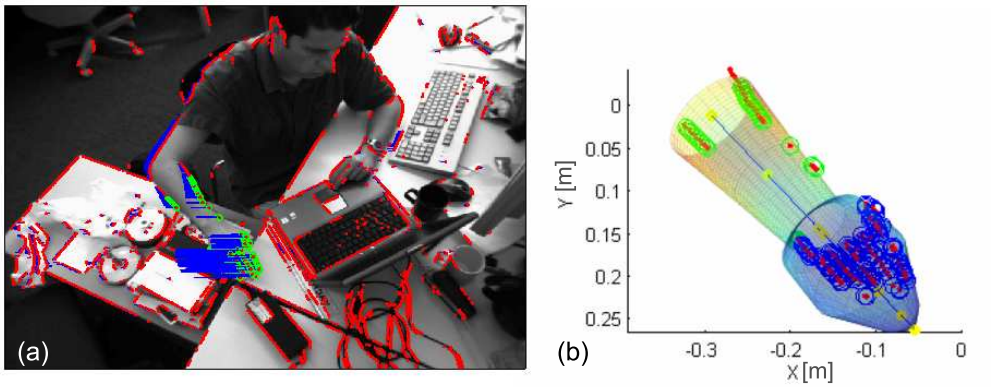


Abbildung 5.2.: Beispielszene für Systemausprägung 1. (a) Spacetime Stereo Daten (rote Punkte mit blauen Bewegungsvektoren) und segmentierte Punkte der Hand-Unterarm-Region (grün). (b) 3D-Modelladaption.

Abb. 5.2 zeigt an einem Beispiel die Eingangsdaten und die Modellanpassung für Systemausprägung 1. Für die Evaluation wird das System als „Tracking by Detection"-Ansatz betrachtet, d.h. dass die Pose $\mathbf{\Phi}(t)$ und die Poseableitung $\dot{\mathbf{\Phi}}(t)$ dazu benutzt werden, die Pose $\mathbf{\Phi}_{\text{init}}(t + n\,\Delta t) = \mathbf{\Phi}(t) + \dot{\mathbf{\Phi}}(t) \cdot (n\,\Delta t)$ für den nächsten Zeitschritt $t + n\,\Delta t$ zu berechnen, bei dem eine neue Modelladaption durchgeführt wird. Die Pose $\mathbf{\Phi}_{\text{init}}(t+n\,\Delta t)$ wird dabei für die Initialisierung der Modelladaption, wie sie in Kap. 4.4.4 beschrieben ist, verwendet.

5.1.2. Systemausprägung 2: Fusion von konturbasierter und punktbasierter Pose- und Bewegungsschätzung

Um nicht ausschließlich auf den 3D-Szenenflusspunkten aus einem der in Kap. 4.2.2 beschriebenen Verfahren aufzubauen, wird in dieser Systemausprägung eine Fusion von sehr unterschiedlichen Pose-Estimation-Verfahren vorgestellt. Dabei wird ein Szenenflussverfahren basierend auf Korrelationsstereo und TV-L^1 Fluss (siehe Kap. 4.2) verwendet, worauf dann wiederum das graphenbasierte Clusterverfahren aufsetzt. Die resultierende Punktewolke wird zur 3D-Poseinitialisierung verwendet. Anschließend wird sowohl die punktbasierte Modellanpassung mittels ICP-Algorithmus (siehe Kap. 4.4.4) durchgeführt, als auch die konturbasierte Poseschätzung auf Grundlage des MOCCD-Algorithmus (Hahn et al., 2008a). Die Ergebnisse dieser beiden Verfahren werden anschließend, wie in Kap. 4.4.9 beschrieben, fusioniert, wobei das fusionierte Ergebnis rückgekoppelt wird, um die beiden Pose-Estimation-Algorithmen neu zu initialisieren und nochmals durchzuführen. Diese Rückkopplung verfeinert die Poseschätzung und konvergiert meist nach wenigen Iterationen. Nach drei bis vier Iterationen ist in der Regel keine signifikante weitere Verbesserung des Poseergebnisses erkennbar, wie sich in Experimenten gezeigt hat. Abb. 5.3 zeigt die Komponenten der Systemausprägung 2.

Bei der Bewegungsanalyse wird eine sukzessive Vorgehensweise verwendet. Eine Initia-

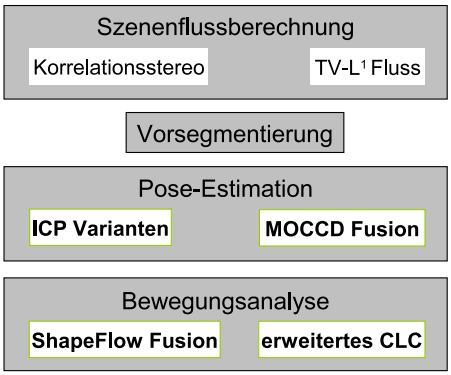


Abbildung 5.3.: Komponenten der Systemausprägung 2.

lisierung der Bewegungsschätzung wird über das Verfahren aus Kap. 4.5.1 durchgeführt. Da hier jedoch nur laterale Bewegungsinformationen verfügbar sind, wird anschließend der sogenannte ShapeFlow-Algorithmus angewendet (siehe Kap. 4.5.2). Dadurch wird konturbasiert die Bewegungsschätzung verfeinert und außerdem auch die Bewegungsinformation entlang der optischen Achse ermittelt.

Grundsätzlich wird durch eine kombinierte Verwendung von 3D-Punkten und Konturinformationen aus multiokularen Bildsequenzen eine robustere Verarbeitung und eine höhere Genauigkeit sowohl bei der Poseschätzung, als auch bei der Bewegungsanalyse erreicht. Dies wird auch in den Experimenten in Kap. 8.2 festgestellt. Dabei zeigt sich, dass die konturbasierten Algorithmen eine sehr genaue Schätzung von Pose bzw. Bewegung zulassen, jedoch einen geringen Konvergenzradius besitzen. Die punktbasierten Ansätze hingegen zeichnen sich durch einen hohen Konvergenzradius aus, sind jedoch was die Genauigkeit angeht etwas schlechter. Daher bietet die Fusion beider Ansätze ein gutes Ergebnis, sowohl was die Genauigkeit als auch die Robustheit angeht.

In diesem Szenario ist die Verwendung einer angepassten Fehlermetrik, wie in Kap. 4.4.6 beschrieben, nicht notwendig, da der Abstand zwischen Objekt und Kamera von 1 – 3 m, was bei dem verwendeten Kamerasystem einer Disparität von ca. 40 Pixeln entspricht, im Vergleich zum Straßenverkehrsszenario (10 – 50 m und ca. 5 – 10 Pixel Disparität) eher gering ist. Die Verwendung des in Kap. 4.4.8 erläuterten modellbasierten Stereos bzw. des in Kap. 4.5.3 erwähnten modellbasierten Szenenflussverfahrens ist nicht möglich, da die Größe des Abbildes des beobachteten Objekts zu gering ist.

5.2. Straßenverkehrsszenario

Im Vergleich zum Produktionsszenario sind die Objekte im Straßenverkehr bei den verwendeten Kamerasystemen meist größer im Bild zu erkennen. Die lateralen Geschwindigkeiten sind ähnlich hoch bzw. etwas höher, jedoch erfolgen Bewegungsänderungen meist nicht so abrupt wie bei der Hand-Unterarm-Region des Arbeiters im Produktionsszenario. Die Hintergründe variieren in diesem Szenario ebenso stark wie in der Produktion.

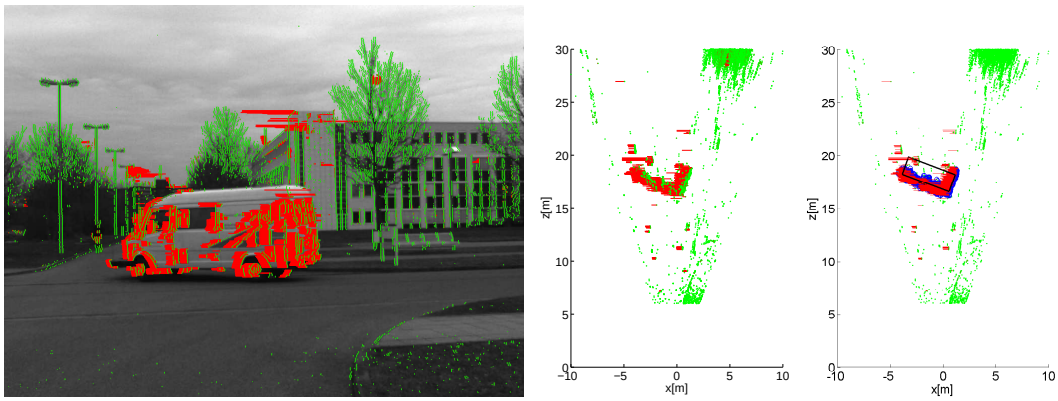


Abbildung 5.4.: (Links) Eingangsbild mit projizierten 3D-Punkten (grün) und assoziierten Bewegungsvektoren (rot) für die horizontale Bewegungskomponente. (Mitte) Sicht von oben auf die Szene. (Rechts) Segementierte Punktewolke des Fahrzeugs (blau) und das angepasste Modell (schwarz).

Insbesondere werden hier Szenen im Kreuzungsbereich untersucht, wobei Objekte mit hoher Geschwindigkeit am Rande des Sichtbereichs auftauchen und gegebenenfalls ihre Richtung ändern. Der Abstand der Objekte zur Kamera (10 – 50 m) ist um etwa eine Größenordnung größer als im Produktionsszenario (1 – 3 m).

5.2.1. Systemausprägung 3: Poseschätzung mittels problemspezifischer Fehlermetrik

Diese Systemausprägung hat ausschließlich die Poseschätzung zum Ziel. Dabei können verschiedene Arten der Szenenflussberechnung verwendet werden (siehe Kap. 4.2). Unabhängig von der Szenenflussberechnung kommt anschließend das graphenbasierte Clusterverfahren nach Kap. 4.3 zum Einsatz. Zur Posebestimmung wird ein ICP-Algorithmus verwendet, jedoch mit einer problemspezifischen Fehlermetrik (siehe Kap. 4.4.6). Dabei sind verschiedene Algorithmen zur Minimierung der Fehlerfunktion denkbar. In den Experimenten in Kap. 9.1 wurde allerdings gezeigt, dass der Levenberg-Marquardt-Algorithmus die höchste Genauigkeit liefert. Abb. 5.5 zeigt die Komponenten der Systemausprägung 3.

Durch die angepasste Fehlermetrik wird dem großen Abstand zwischen Objekt und Kamera Rechnung getragen. Der ICP-Algorithmus nach Besl und McKay (1992) arbeitet mit einer euklidischen Abstandsberechnung, was bei 3D-Punkten, die auf Basis eines Stereokamerasystems berechnet wurden, bei dem die Basisbreite deutlich kleiner ist als der Abstand zum Objekt, nicht sinnvoll ist, da die Fehler, die bei dieser Messung gemacht werden, nicht symmetrisch in alle Raumrichtungen sind. Die Poseschätzungen der einzelnen Zeitschritte werden zeitlich nicht gefiltert, sondern separat betrachtet, was einen "Tracking by Detection"-Ansatz darstellt.

Der Vergleich der euklidischen Metrik gemäß dem Stand der Technik mit der an-

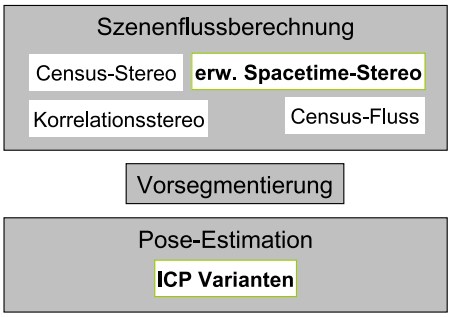


Abbildung 5.5.: Komponenten der Systemausprägung 3.

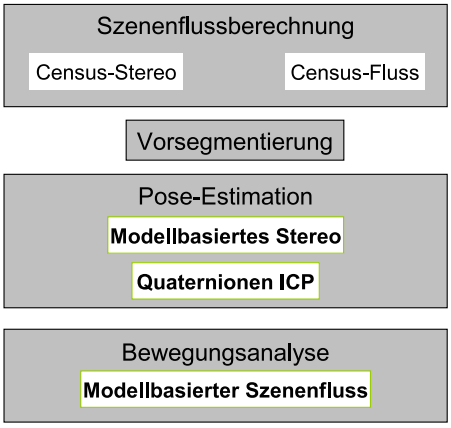


Abbildung 5.6.: Komponenten der Systemausprägung 4.

gepassten Metrik zeigt, dass diese mathematische Anpassung die Genauigkeit der Poseschätzung erhöhen kann (siehe Experimente Kap. 9.1). Abb. 5.4 zeigt beispielhaft die Verarbeitungsschritte bei der Systemausprägung.

5.2.2. Systemausprägung 4: Modellbasiertes Stereo und modellbasierter Szenenfluss

Um nach der punktbasierten Posebestimmung die Genauigkeit der Ergebnisse zu verbessern, wird in dieser Systemausprägung das sogenannte modellbasierte Stereo verwendet. Dabei wird die Objektpose direkt aus den beiden Stereobildern ermittelt, wobei eine klassische 3D-Punkteberechnung lediglich zur Initialisierung der Pose dient. Nach dieser Initialisierung wird die Poseinformation rückgekoppelt um auf Basis der Eingangsbilder direkt die Pose zu ermitteln. Man verwendet also keine explizite Korrespondenzbildung, sondern arbeitet direkt auf den Eingangsbildern, wodurch alle zur Verfügung stehenden Informationen genutzt werden. Abb. 5.6 zeigt die Komponenten der Systemausprägung 4.

Zur Schätzung der dreidimensionalen Bewegung wird der modellbasierte Szenenfluss verwendet. Dabei wird der gleiche Ansatz wie auch beim modellbasierten Stereo benutzt,

um nun jedoch vier Bilder, jeweils die beiden Stereobilder zu zwei aufeinanderfolgenden Zeitschritten, miteinander zu verknüpfen. Hierdurch wird die komplette Bewegung geschätzt, auch die Bewegungskomponente entlang der optischen Achse.

Im Verhältnis zu dichten Szenenflussverfahren aus der Literatur wird bei dem vorgestellten, neuen Ansatz Rechenzeit eingespart, da nur dichter Szenenfluss für das interessierende Objekt bestimmt wird. Die Szenensegmentierung und die Poseinitialisierung werden mittels schneller Stereo- und optischer Flussverfahren berechnet. Dabei kommt auch der sogenannte Quaternionen-ICP (siehe Kap. 4.4.7) zum Einsatz, um nochmals um ca. Faktor 40 schneller eine robuste Aussage über die Objektpose geben zu können als mit dem ICP-Algorithmus nach Besl und McKay (1992). Hier zeigen sich zwar auch Einbußen in Bezug auf die Posegenauigkeit, jedoch wird dies durch die anschließende Berechnung der Pose über das modellbasierte Stereo ausgeglichen.

Die Experimente in Kap. 9.2 zeigen, dass es auch mit einem schwachen Modell (Quader) möglich ist, sowohl die Pose als auch die Bewegung eines Fahrzeugs auf Basis von modellbasierter Stereo- bzw. Szenenflussberechnung zu bestimmmen.

Durch die beiden Systemausprägungen 3 und 4 ist somit eine robuste und genaue Schätzung der Pose und der Bewegung von Fahrzeugen im Kreuzungsbereich möglich. Ein Einsatz der Bewegungsanalyse auf Basis von Verschiebungsvektorfeldern (siehe Kap. 4.5.1) ist nicht notwendig, da sich die Fahrzeuge lateral nur eindimensional horizontal bewegen können. Eine Verwendung von konturbasierten Pose-Estimation-Methoden ist in diesem Szenario nicht umsetzbar, da unbekannte Fahrzeuge eine nicht vorhersagbare Kontur im Bild darstellen. Daher ist eine Fusion wie bei der Hand-Unterarm-Erkennung in Kap. 4.4.9 beschrieben nicht möglich.

6. Integration von Modellwissen zur Erhöhung der Systemrobustheit

6.1. Korrektur von Fehlkorrespondenzen

Stereobildverarbeitungsalgorithmen erzeugen Fehlkorrespondenzen, wenn repetitive Strukturen in der vorliegenden Szene vorhanden sind, da dadurch Mehrdeutigkeiten in der Korrespondenzanalyse vorliegen. Das Vergleichmaß ist an mehreren Stellen klein und sehr ähnlich und aufgrund des Pixelrauschens beschreibt nicht zwingend der kleinste Vergleichswert die richtige Korrespondenz. Diese Problematik tritt insbesondere dann auf, wenn lokale Stereoverfahren (siehe Kap. 2.1.2) verwendet werden. Jedoch entstehen diese Fehlzuordnungen auch unter Verwendung von dichten, globalen Verfahren. Grundsätzlich ist nur möglich die Entfernung zu repetitiven Strukturen zu schätzen, wenn diese im Bild beginnen oder enden, andernfalls können die Mehrdeutigkeiten nicht aufgelöst werden. Abb. 6.1 zeigt die 3D Rekonstruktion verschiedener Stereoalgorithmen der gleichen Szene, die ein planares Schachbrettmuster zeigt.

In der Stereobildverarbeitung wird durch verschiedene Constraints versucht, dieses Problem zu beheben (siehe wiederum Kap. 2.1.2). Speziell für sich wiederholende Strukturen werden von Di Stefano et al. (2004) die Qualität des Minimums der Kostenfunktion und der korrespondierende Disparitätswert durch die Einführung eines Eindeutigkeits- und Schärfetests bewertet, um Mehrdeutigkeiten aufzulösen. Nedevschi et al. (2004) unterdrücken grundsätzlich Korrespondenzen, sollte eine Mehrdeutigkeit in der Korrespondenzanalyse vorliegen.

Daneben gibt es auch Verfahren, die fehlerhafte Korrespondenzen explizit behandeln. Murray und Little (2004) benutzen den sogenannten RANSAC-Algorithmus (Random Sample Consensus, (Fischler und Bolles, 1981)), um eine Ebene an die 3D-Punkte anzupassen und dann fehlerhafte Punkte erkennen und löschen zu können. Sepehri et al. (2004) benutzt einen ähnlichen Ansatz, jedoch wird die Ebene bzw. die Ebenen mittel M-Schätzer angepasst (Huber, 1981; Rey, 1983).

Das in diesem Abschnitt vorgestellte Verfahren stellt einen neuen Ansatz dar, um die Probleme, die durch sich wiederholende Strukturen entstehen, zu lösen, wobei das Verfahren unabhängig vom verwendeten Stereobildverarbeitungsalgorithmus einsetzbar ist. In einem ersten Schritt wird die vorliegende Szene durch ein konventionelles Stereoverfahren rekonstruiert, was zu korrekten und inkorrekten 3D-Punkten führt. Ein applikationsabhängiges Szenen- oder Objektmodell wird an die initiale 3D-Punktewolke angepasst, was zu einer Modellpose führt. Diese Pose wird für eine erneute Korrespondenzanalyse verwendet, wobei die Distanz des 3D-Punkts zum Modell in die Kostenfunktion der Korrespondenzanalyse mit eingeht und so die Qualität der Rekonstruktion

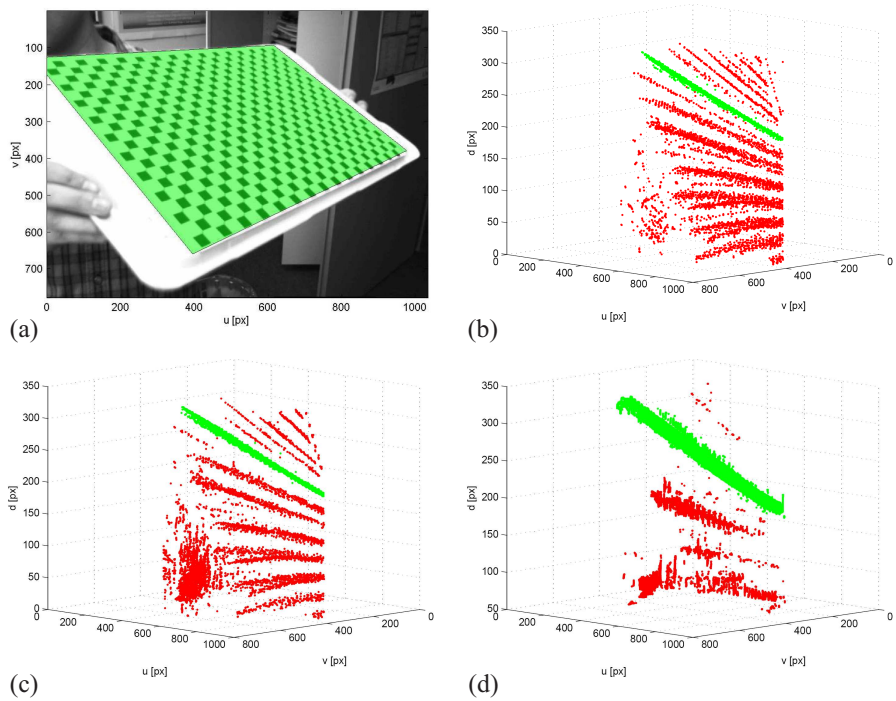


Abbildung 6.1.: Durch verschiedene Stereoverfahren generierte Fehlzuordnungen in der markierten Region. Grüne Punkte: korrekte Disparität; rote Punkte: falsche Disparität. Die 3D-Punkte werden im Disparitätsraum dargestellt, wobei u und v die Pixelkoordinaten im rechten Bild sind und d die assoziierte Disparität darstellt. (a) Rechtes Stereobild; repetitive Struktur in grün gekennzeichnet. (b) Ergebnis des Spacetime Stereo. (c) Korrelationsstereo. (d) Semi-Global-Matching. Zu den verschiedenen Algorithmen siehe Kap. 2.1.2.

erheblich gesteigert werden kann.

6.1.1. Umfeldmodellierung mittels Ebenenmodell

Im Bereich der mobilen Robotik ist die Modellierung der Umwelt oder Teilen daraus durch eine Ebene ein viel verwendeter Ansatz, wodurch eine approximierte Szenendarstellung bereitgestellt wird (Biber et al., 2004). Das Ebenenmodell ist für verschiedene Objekte mit repetitiver Struktur sinnvoll: Fliesen, Zäune oder Gebäudefronten. Die repetitive Struktur auf dem realen Objekt ist für gewöhnlich äquidistant, wobei die Struktur im Bild nicht notwendigerweise äquidistant ist.

Detektion und Charakterisierung von repetitiven Strukturen

Repetitive Strukturen sind durch ein repetitives Grauwertmuster charakterisiert, welches zu einem signifikanten Peak im assoziierten Amplitudenspektrum führt. Horizontale Linien einer festen Länge (z.B. 128 Pixel) werden überlappend aus jeder Bildzeile ausgeschnitten und mittels einer Fast Fourier Transformation (FFT) auf diese Auffälligkeit hin untersucht. Im Amplitudenspektrum wird das Maximum neben dem Gleichanteil extrahiert und mittels einer gegebenen Schwelle, welche von Mittelwert und Standardabweichung des Spektrums abhängt, auf seine Signifikanz überprüft. Anschließend wird die korrespondierende Wellenlänge berechnet. Um eine kontinuierliche Repräsentation der Wellenlängen zu erhalten wird eine Ebene an die einzelnen Wellenlängen angepasst, wobei ein M-Schätzer verwendet wird. Ausreißer werden dabei verworfen und die Bildbereiche, die eine repetitive Struktur enthalten, markiert. Das gleiche Verfahren wird auch auf vertikale Bildlinien angewandt. Das Ergebnis dieser horizontalen und vertikalen Sprektralanalyse sind Bildregionen mit repetitiven Strukturen und Ebenenfunktionen $\lambda_h(u, v)$ und $\lambda_v(u, v)$, welche die Wellenlängen für jede markierte Bildkoordinate (u, v) bereitstellt.

Bestimmung der Modellparameter: Im 3D-Raum wird die Ebene mit der repetitiven Struktur durch die Modellierung von Sehstrahlen unter Zuhilfenahme der ermittelten Wellenlängeninformationen ermittelt. Diese Strahlen beschreiben die Beziehung zwischen repetitiver Struktur in der Szene und deren Abbild in den vorliegenden Bildern. Die Koordinaten $u_{i,j}$ und $v_{i,j}$ der Schnittpunkte zwischen den Sehstrahlen und der Bildebene werden definiert durch

$$u_{i,j} = u_{i,j-1} + \lambda_h \left(u_{i,j-1}, v_{i,j-1}\right) \tag{6.1}$$

$$v_{i,j} = v_{i-1,j} + \lambda_v \left(u_{i-1,j}, v_{i-1,j}\right), \tag{6.2}$$

wobei i den Index des Strahls in einer Spalte beschreibt, j den Index in einer Zeile, $u_{i,0} = 0$, und $v_{0,j} = 0$.

Strahlen, die ausgehend von diesen Schnittpunkten das optische Zentrum schneiden, sind charakteristisch für die repetitive Struktur. Die Annahme, dass die repetitive Struktur im Raum äquidistant ist, führt zu der Bedingung, dass auch die Schnittpunkte zwischen diesen Strahlen und der Modellebene äquidistant sein müssen. Die Distanz zwischen den einzelnen Schnittpunkten wird mit s in Abb. 6.2a bezeichnet. Somit kann der Normalenvektor der Ebene im 3D-Raum bestimmt werden, wobei der Offsetparameter der Ebene zunächst unbestimmt bleibt und über die initiale 3D-Punktewolke ermittelt werden muss.

Wie bereits in Kap. 4.4.6 betrachtet, sind die Meßfehler der durch das Stereoverfahren ermittelten Disparitäten gaußverteilt, wohingegen die 3D-Koordinaten ein unsymmetrisches Rauschverhalten aufweisen. Daher wird die folgende Betrachtung im Disparitätsraum durchgeführt, wo sich die einzelnen Disparitäten unabhängig von der Objektentfernung besser trennen lassen.

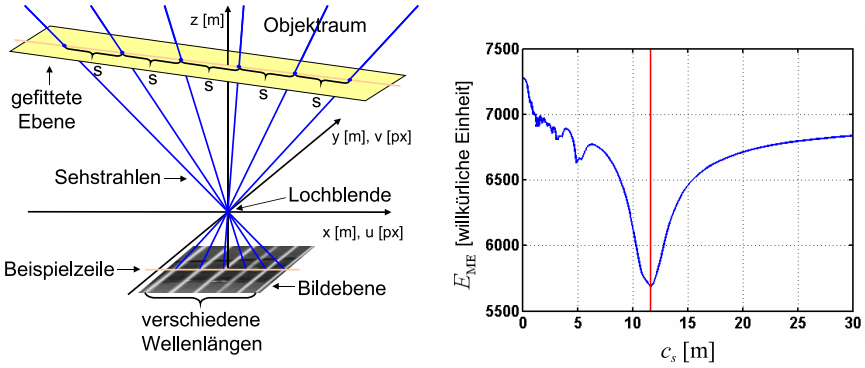


Abbildung 6.2.: Vorgehen bei der Modelladaption des Ebenenmodells bei sich wiederholenden Strukturen. (a) Ebenefit basierend auf der repetitiven Struktur. (b) Abhängigkeit des Disparitätsfehlers E_{ME} entsprechend Gl. 6.6 in Bezug auf den Offsetparameter c_s für Szene 1 (Zaun mit Person) aus den Experimenten (siehe Kap. 10.1).

Die Modellebene ist im 3D-Raum gegeben durch

$$\epsilon_s : \; z(x,y) = a_s x + b_s y + c_s \; . \tag{6.3}$$

Durch die Transformation der Modellebene von 3D-Koordinaten in den Disparitätsraum mithilfe der Gleichungen des Lochkameramodells (siehe Kap. 2.1.1) erhält man folgende Gleichung für die Modellebene:

$$\epsilon_d : \; d(u,v) = a_d u + b_d v + c_d \tag{6.4}$$

mit

$$a_d = -\frac{b}{c_s}\, a_s \; , \qquad b_d = -\frac{b}{c_s}\, b_s \; , \qquad c_d = \frac{bf}{c_s}. \tag{6.5}$$

Der Offsetparameter c_s der Ebene $z(x,y)$ im 3D-Raum, deren Normalenvektor durch die vorher bestimmten Parameter a_s und b_s definiert ist, wird über die Transformation in den Disparitätsraum bestimmt (Gl. 6.4 und 6.5). Hierzu wird die mittlere Distanz zwischen allen 3D-Punkten, die zu der repetitiven Struktur gehören und der Modellebene im Disparitätsraum in Pixeln berechnet. Diese Distanzen werden mittels eines M-Schätzer gewichtet, was zu folgender Fehlerfunktion führt:

$$E_{\mathrm{ME}}(c_s) = \left\langle \left\{ M\left(d_i - \left[-\frac{a_s l}{c_s}\, u_i - \frac{b_s l}{c_s}\, v_i + \frac{lf}{c_s} \right] \right) \right\}^2 \right\rangle_{(u_i,v_i)\, \in\, \epsilon_d} \tag{6.6}$$

mit $M(x)$ als "Fair"-Gewichtsfunktion (Rey, 1983):

$$M(x) = 1 - \frac{1}{1 + \left|\frac{x}{k_{\mathrm{ME}}}\right|}, \tag{6.7}$$

wobei k_{ME} ein benutzerdefinierter Parameter ist und $\langle \ldots \rangle$ den Mittelwert beschreibt. Die Funktion $E_{\mathrm{ME}}(c_s)$ wird über den Offsetparameter c_s der Ebene ϵ_s im 3D-Raum unter Verwendung einer Intervallschachtelung minimiert. Das Verhalten von $E_{\mathrm{ME}}(c_s)$ wird in Abb. 6.2b gezeigt.

6.1.2. Anwendung bei artikulierten Objekten

Um zu zeigen, dass das vorgestellte, neue Verfahren auch bei komplexen Objekten und Szenen Anwendung findet, wird es auch im Produktionsszenario angewendet. In einem Sicherheitssystem sind falsche Korrespondenzen, die zu Phantomobjekten führen kritisch, da sie Notfallstopps des Roboters verursachen. Würde man hingegen die Verfahren von Murray und Little (2004) oder von Sepehri et al. (2004) benutzen, würden Gegenstände, die sich vor der repetitiven Struktur befinden aus der Punktewolke gelöscht, was ebenfalls ein Sicherheitsrisiko bedeutet. Als Modell wird das in Kap. 4.4.1 vorgestellte Hand-Unterarm-Modell verwendet, wobei speziell hier das Modell verlängert wird, um auch den Oberarm mit einzubeziehen. Zur Poseschätzung wird Systemausprägung 1, wie in Abschnitt 5.1.1 vorgestellt, verwendet.

Um die Komplexität für die folgende Analyse sinnvoll zu reduzieren, wird das Modell in drei aneinanderhängende Ebenen gewandelt: die Finger, die Hand und der Arm. Dies ist sinnvoll, da die Ebenen nach wie vor die korrekten Disparitäten für das Objekt wiedergeben. Außerdem ist diese Vereinfachung durchführbar, da die Abweichungen, die sich durch die Approximation ergeben, vernachlässigbar sind im Vergleich zu den Disparitätsfehlern, die durch die repetitive Struktur entstehen (siehe auch Kap. 10.1).

6.1.3. Rückkopplung des Modellwissens

Die initiale Korrespondenzanalyse basiert auf der Matrix $\mathbf{E_{SSD}}$ (siehe Kap. 4.2.1). Sind in der vorliegenden Szene repetitive Strukturen vorhanden, so sind die SSD-Werte in der Matrix klein und sehr ähnlich für alle möglichen Korrespondenzen, was zu einer Vielzahl von falschen Korrespondenzen führt. Basierend auf dem Modellwissen wird ein zusätzlicher Fehlerterm berechnet. Befindet sich ein Punkt im Bereich der repetitiven Struktur, so werden die Disparitäten aller möglicher Korrespondenzen berechnet und mit dem Modell verglichen, wodurch die Matrix $\mathbf{E_d}$ dieser Disparitätsfehler entsteht. Diese beiden Matrizen werden zu einer Gesamtfehlermatrix $\mathbf{E_t}$ mithilfe des Gewichtsfaktors λ_e vereint:

$$\mathbf{E_t} = \mathbf{E_{SSD}} + \lambda_e \mathbf{E_d}. \tag{6.8}$$

Die verfeinerte Korrespondenzanalyse wird basierend auf dieser Matrix $\mathbf{E_t}$ durchgeführt, wobei die gleichen Constraints wie für die initiale Stereoanalyse benutzt werden.

Der Einfluss des Abstands zwischen den 3D-Punkten und dem Modell auf die endgültige Punktewolke steigt mit steigendem Wert von λ_e. Experimente haben gezeigt, dass die 3D-Rekonstruktion nicht stark abhängig vom gewählten Wert λ_e ist und dass eine signifikante Verbesserung des Rekonstruktionsergebnisses für eine Vielzahl von Szenen mit dem gleichen Wert von λ_e erreicht wird.

6.1.4. Wertung

Das neu vorgestellte Verfahren basiert auf einer Modellanpassung und einer erneuten Korrespondenzanalyse um Fehlkorrespondenzen, die durch repetitive Strukturen bei der Stereobildanalyse entstehen, zu korrigieren. Dabei wird bei der erneuten Korrespondenzanalyse nicht alleine der Modellinformation vertraut, sondern weiterhin auch das ursprüngliche Vergleichmaß verwendet, wodurch Objekte, die sich vor der repetitiven Struktur befinden, auch nach der Verbesserung der 3D-Punkte noch vorhanden bleiben.

Das vorgestellte Verfahren ist unabhängig vom verwendeten Stereoalgorithmus. Es können sowohl rigide, einfache Objekte, als auch komplexe, artikulierte Objekte mit sich wiederholenden Strukturen modelliert werden und so die Fehlkorrespondenzen aufgelöst werden. Für das Ebenenmodell wurde ein FFT-basiertes Verfahren zur Bestimmung der Poseparameter vorgestellt, welches gleichzeitig als Detektor für repetitive Strukturen fungiert.

In den Experimenten in Kap. 10.1 wird gezeigt, dass der Gewichtungsparameter λ_e recht unempfindlich ist und eine größenordnungsgenaue Einstellung ausreichend ist, um die Punkte im Bereich der repetitiven Struktur zu korrigieren und gleichzeitig Objekte vor der Struktur zu erhalten.

6.2. Klassifikation artikulierter Objekte

In Kap. 5 werden Systeme zur Verfolgung von Objekten in Stereobildsequenzen beschrieben. Bislang unbeachtet blieb dabei die Frage, ob das beobachtete Objekt auch wirklich jenes ist, welches von Interesse ist. Es erfolgt soweit keine Verifizierung der Objektauswahl, d.h. es wurde angenommen, dass die in Kap. 4.3 beschriebene Segmentierung zuverlässig das interessierende Objekt aus der Punktewolke der Szene extrahiert. Da es sich, vorallem im Produktionsszenario, um eine sicherheitskritische Anwendung handelt, muss sichergestellt werden, dass es sich bei dem beobachteten Objekt um das für die Anwendung relevante handelt.

Im Straßenverkehrsszenario werden, um die verschiedenen Klassen von Objekten zu erkennen, bereits seit längerem Verfahren verwendet, die meist durch eine Kamera gewonnene Bildinhalte analysieren und Objekte klassifizieren (Wöhler und Anlauf, 2001; Wender und Dietmayer, 2007).

Bei der Fußgängererkennung handelt es sich um die Erkennung artikulierter Objekte, wobei der Klassifikator mithilfe einer großen Anzahl von Trainingsbeispiel adaptiert wird. Die Trainingsbeispiele zeigen dabei Personen in den am häufigsten auftretenden Poses. Dies führt jedoch zu einer sehr großen Trainingsdatenbank, um möglichst hohe

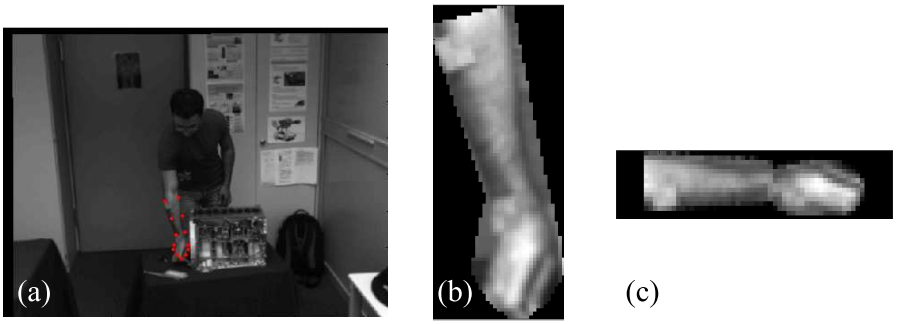


Abbildung 6.3.: Beispiel für die Generierung einer normierten Ansicht der Hand und des Unterarms einer Person. (a) Eingangsbild mit projiziertem Objektmodell. (b) Entlang der Kontur ausgeschnittener Bildinhalt. (c) Bildinhalt entsprechend der Objektpose gedreht, skaliert und dilatiert. Normierte Objektansicht.

Erkennungsraten zu erzielen.

Ein Ansatz bei der Fahrzeugerkennung (Wender und Dietmayer, 2007) ist es, zunächst die Modellpose zu ermitteln und mit deren Hilfe den Bildinhalt zu rektifizieren, um eine normierte Ansicht des Objekts zu erhalten. Durch diese normierte Ansicht wird die Klassifikationsaufgabe wesentlich vereinfacht, jedoch ist diese Technik auf rigide Objekte wie z.B. ein zweiachsiges Fahrzeug beschränkt.

Da es sich bei den Körperteilen der Person im Produktionsszenario um artikulierte Teilobjekte handelt, ist die Klassifikationsaufgabe weitaus komplexer als bei rigiden Objekten. Experimente haben gezeigt, dass ohne Modellwissen bzw. ohne Wissen über die Pose und die internen Freiheitsgrade der jeweiligen Körperregion das Klassifikationsergebnis nicht zufriedenstellend ist.

6.2.1. Generierung einer normierten Objektansicht auf Basis der ermittelten Pose

Auf Basis von Systemausprägung 1 oder 2 (siehe Kap. 5.1.1 bzw. 5.1.2) werden zunächst die Pose der Hand-Unterarm-Region und die Knickwinkel des Handgelenks ermittelt. Anschließend werden 13 Punkte auf dem Umriss des Modells ins Bild projiziert, um dann den Bildinhalt innerhalb dieser Modellkontur auszuschneiden. Des Weiteren wird auf der Basis der Poseinformationen der Bildinhalt so transformiert, dass eine poseunabhängige, normierte Ansicht des Objekts bereitsteht, ähnlich dem Verfahren von Wender und Dietmayer (2007). Für die Transformation wird das ausgeschnittene Bild zuerst gedreht und dann so skaliert, dass eine definierte Bildgröße erreicht wird. Zusätzlich zu dem Verfahren von Wender und Dietmayer (2007) werden dabei auch die internen Freiheitsgrade berücksichtigt, damit die normierte Ansicht auch unabhängig vom Knickwinkel des Handgelenks ist. Abb. 6.3 zeigt den Vorgang an einem Beispiel.

6.2.2. Klassifikation des normierten Hand-Unterarm-Abbilds

Die normierte Ansicht wird anschließend durch einen Klassifikator, in diesem Fall durch einen Polynomklassifikator, klassifiziert, um eine Aussage darüber zu erhalten, ob das vermutete Objekt auch wirklich mit dem Bildinhalt übereinstimmt. Die Pose-Estimation-Verfahren allein sind dazu i.A. nicht in der Lage. Der verwendete Klassifikator wird dabei auch mit normierten Ansichten des Objekts trainiert. Dazu wird die Pose-Estimation auf einer Bildsequenz angewendet und so für jeden Zeitschritt ein normiertes Abbild erzeugt. Zur Erstellung von Negativbeispielen wird eine willkürliche Pose angenommen und damit ein zufälliger Teil der Szene ausgeschnitten, welcher meist den Hintergrund zeigt. Diese Klasse von Bildausschnitten, die keine Hand-Unterarm-Region einer Person repräsentieren, wird als (*kein HU*) bezeichnet, wohingegen Bildausschnitte mit einer Hand-Unterarm-Abbildung der Klasse *HU* zugeordnet werden.

Aus einigen Sequenzen mit unterschiedlicher Kleidung des Arbeiters, unterschiedlicher Beleuchtung, etc. kann so eine Trainingsdatenmenge erzeugt werden. Auf Basis dieser Daten wird ein geeigneter Klassifikationsansatz verwendet, wobei der Klassifikator an die Stichprobe adaptiert wird. Experimente haben gezeigt, dass ein Polynomklassifikator für die vorliegenden Aufgabenstellung im Produktionsszenario geeignet ist. Er wird daher in der vorliegenden Arbeit verwendet (Schürmann, 1996; Kreßel und Schürmann, 1997). Um nicht die Gesamtzahl der Pixel, die nach der Normierung der Objektansicht vorhanden sind, als Merkmalsvektor dem Polynomklassifikator zu übergeben, wird eine Dimensionsreduktion auf Basis einer Hauptkomponentenanalyse (engl.: principal component analysis (PCA)) durchgeführt (Ott, 1977). Dadurch wird die Dimensionalität des Merkmalsvektors sinnvoll reduziert, ohne signifikante Informationen zu verlieren. Ergebnisse der Klassifikation sind geschätzte Klassenzugehörigkeitswahrscheinlichkeiten, die entsprechend der Trainingsdaten aussagen, mit welcher Wahrscheinlichkeit das aktuell präsentierte Beispiel zu einer der beiden Klassen (*HU* oder *kein HU*) gehört. Abb. 6.4 zeigt einen Überblick über das Klassifikationssystem.

6.2.3. Wertung

Die Klassifikationsaufgabe wird durch das beschriebene Verfahren wesentlich vereinfacht, da immer nur eine Ansicht des Objekts nach der Normierung bereitgestellt wird. Die Informationen, die zur Normierung verwendet werden, ergeben sich durch Pose-Estimation-Verfahren, die bei der Szenenanalyse ohnehin ermittelt werden.

Das Verfahren kann beispielsweise bei der Detektion der Hand-Unterarm-Region des Menschen eingesetzt werden. Hierbei werden durch das Handgelenk zwei zusätzliche, interne Freiheitsgrade generiert. Benutzt man das ursprüngliche Bild zur Klassifikation, so führt dies zu einem unzureichenden Klassifikationsergebnis, wie Experimente gezeigt haben. Durch die normierte Ansicht werden die internen Freiheitsgrade und die unterschiedlichen Ansichten des Objekts, die beide zu einer Verringerung der Klassifikationsleistung führen, eliminiert. Die Klassifikationsaufgabe wird so wesentlich vereinfacht und trägt dann dazu bei, Hand und Unterarm einer Person möglichst robust zu erkennen.

Ein weiterer Vorteil des beschriebenen Verfahrens ist, dass eine verhältnismäßig kleine

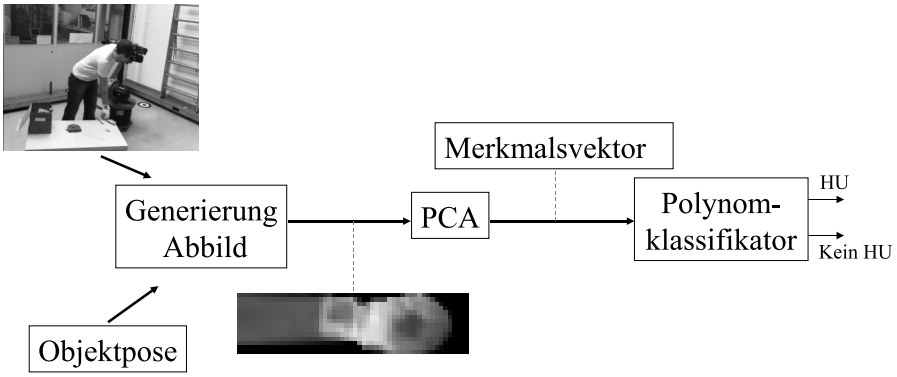


Abbildung 6.4.: Systemdarstellung der Klasifikation artikulierter Objekte. Aus dem Eingangsbild und der Objektpose wird eine normierte Objektansicht generiert. Anschließend wird eine Dimensionsreduktion durchgeführt und mithilfe eines Polynomklassifikators der entstandene Merkmalsvektor klassifiziert. Am Ende steht eine Klassenzugehörigkeitswahrscheinlichkeit zur Verfügung, die aussagt, ob es sich um ein Hand-Unterarm-Abbild handelt (*HU*) oder nicht (*kein HU*).

Menge an Trainingsbeispielen ausreicht, um ein gutes Klassifikationsergebnis zu erhalten, da durch die vorherige Normierung der Ansicht die Varianz in der Pose, der Größe und dem Seitenverhältnis der Ansicht nicht im Klassifikator mit gelernt werden muss.

Teil III.

Experimentelle Untersuchungen

7. Ermittlung von Ground-Truth-Daten durch ein unabhängiges System

Um die Güte der in der vorliegenden Arbeit beschriebenen Verfahren beurteilen zu können, wird eine unabhängige Ground-Truth-Information benötigt, d.h. die Ground-Truth muss durch ein unabhängiges System ermittelt werden. Würde man einen zweiten Sensor neben der Stereokamera hierfür verwenden, z.b. einen Laserscanner, so müssten die beiden Sensorsysteme aufeinander bzw. auf ein gemeinsames Weltkoordinatensystem kalibriert werden, was zusätzliche Ungenauigkeiten verursacht. D.h. beim Beispiel des Laserscanners würden Fehler, die bei der Kalibrierung des Laserscanners auf die Stereokamera gemacht werden, zu einem systematischen Fehler bei den experimentellen Untersuchungen führen. Daher wird davon abgesehen und direkt aus den Bildern der Stereokamera die Ground-Truth-Information in beiden Anwendungsszenarien ermittelt.

Dazu werden bei der Aufnahme der Testsequenzen bestimmte Punkte auf dem interessierenden Objekt markiert. In den Sequenzen werden diese Punkte später in allen Kamerabildern zu jedem Zeitschritt wiedergefunden und über den sogenannten Bündelausgleich (Triggs et al., 2000) zu einer 3D-Information verarbeitet. Das Auffinden dieser Punkte in den Bildern kann zum einen manuell oder auch automatisiert ablaufen. Bei der manuellen Methode werden Punkte auf das Objekt aufgebracht, die im Bild durch eine Person manuell gesucht werden. Anschließend werden die Positionen dieser Punkte in Pixelkoordinaten ganzzahlig bestimmt. Bei der automatisierten Methode werden farbige Schachbrettecken auf Papier gedruckt und anschließend auf das Objekt geklebt. Diese Farbmuster können in der Bildsequenz durch einen Farbfilter vom Hintergrund separiert und anschließend durch einen subpixelgenauen Eckendetektionsalgorithmus (Krüger und Wöhler, 2009) vermessen werden. Die Farbaufnahme wird nur zur Bestimmung der Ground-Truth benutzt. Zur Evaluierung der entwickelten Verfahren werden alle Bilder in Grauwertbilder gewandelt.

Für sehr nahe Objekte ist die Genauigkeit, die man durch die manuelle Methode erhält, ausreichend (siehe Kap. 8.1). Ist das beobachtete Objekt jedoch weiter von der Kamera entfernt, so wird eine subpixelgenaue Vermessung der Referenzpunkte benötigt, d.h. die automatisierte Methode kommt zum Einsatz. Dabei wird eine Genauigkeit für die 2D-Eckenlokalisation von 0.032 Pixel erreicht. Dadurch kann davon ausgegangen werden, dass die Ground-Truth um mindestens eine Größenordnung genauer ist, als die Ergebnisse, die auf Basis eines Stereobildverarbeitungsalgorithmus etabliert wurden (Genauigkeit der Disparitäten typischerweise etwa 0.1 Pixel).

Im Produktionsszenario werden die drei Referenzpunkte des Hand-Unterarm-Modells auf dem Arm der Testperson entweder durch Punkte oder durch die angesprochenen Schachbrettecken markiert. Durch die Berechnung der euklidischen Distanz zwischen

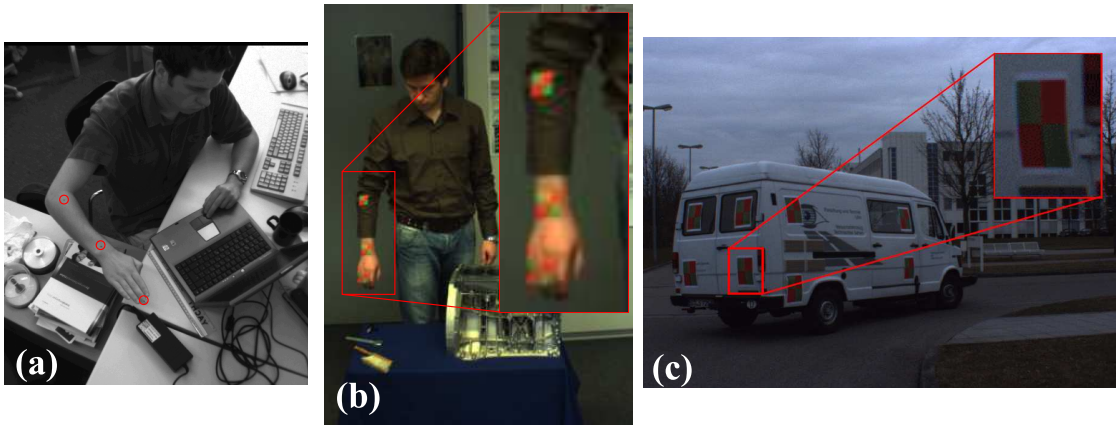


Abbildung 7.1.: Beispiele für die Marker, die in beiden Szenarien zur Ermittlung der Ground-Truth dienen. (a) Pixelgenaue Vermessung durch manuelle Suche der Punktmarker (rot gekennzeichnet). (b) Subpixelgenaue Vermessung im Produktionsszenario auf Basis von Farbklassifikation und Schachbretteckendetektion. (c) Subpixelgenaue Vermessung im Straßenverkehrsszenario, sonst gleiches Verfahren wie in (b).

den somit ermittelten Referenzpunkten und den Punkten des Modells in der geschätzten Pose, kann die Güte der Poseschätzung beurteilt werden.

Im Straßenverkehrsszenario wird aufgrund der höheren Distanz zwischen Kamera und Objekt stets die Methode auf Basis der Schachbrettecken verwendet. Hierbei ist die aktuell beobachtete Seite des Fahrzeugs mit vier dieser Ecken beklebt, um die entsprechende Fläche möglichst genau ermitteln zu können. Durch die vier 3D-Positionen kann der aktuelle Gierwinkel des Fahrzeugs zweifach bestimmt werden und über den Mittelwert der beiden Werte eine zuverlässige Aussage über diesen Winkel gemacht werden.

Abb. 7.1a und Abb. 7.1b zeigen die beiden unterschiedlichen Arten der Ground-Truth-Ermittlung im Produktionsszenario. Abb. 7.1c zeigt die subpixel-genaue Ground-Truth-Bestimmung im Straßenverkehrsszenario.

8. Untersuchungen im Produktionsumfeld

8.1. Experimente zur Systemausprägung 1

Die in Kap. 5.1.1 beschriebene Systemausprägung wird anhand realer Szenen, in denen die Hand-Unterarm-Region einer Person sich ungleichförmig vor einem strukturierten Hintergrund bewegt, evaluiert (siehe Abb. 8.1 und 8.2). Die Distanz zwischen Hand-Unterarm und Kamera liegt dabei zwischen 0.85 und 1.75 m, die Bildauflösung bei 2 bis 3 mm pro Pixel. Für die Aufnahmen wird ein PointGrey Digiclops CCD Kamerasystem benutzt. Ein Zeitschritt zwischen aufeinanderfolgenden Bildern beträgt $\Delta t = 50$ ms. Szenenflussinformationen werden durch das Spacetime Stereo-Verfahren auf Basis von jeweils drei aufeinanderfolgenden Bildpaaren ermittelt, wobei die maximale Schrägheit der detektierten Kanten mit $\delta_{\max} = 2$ definiert wird (siehe Gl. 4.11).

Für jede der drei betrachteten Sequenzen wird die Evaluierung für verschiedene Werte von n durchgeführt, wobei der Parameter n die Anzahl der Zeitschritte definiert, die zwischen zwei Modelladaptionen liegen und über die hinweg die Objektbewegung prädiziert wird. Abb. 8.3 zeigt das Ergebnis. Als Maß für die Genauigkeit der ermittelten Pose wird die mittlere euklidische Distanz zwischen der Ground-Truth Position und der ermittelten Position der drei Punkte P_1 (Kreise), P_2 (Quadrate) und P_3 (Rauten) spaltenweise für die drei Testsequenzen dargestellt. Die zweite Zeile zeigt den mittleren Fehler und die Standardabweichung der translatorischen Bewegungskomponenten U_{obj} (Kreise), V_{obj} (Quadrate) und W_{obj} (Rauten) pro Zeitschritt. Für jeden Wert von n definieren die linken drei Punkte den Unterarm, die rechten drei die Hand. Die dritte Zeile zeigt den mittleren Fehler und die Standardabweichung der rotatorischen Bewegungskomponente ω_p (Kreise) und ω_o (Quadrate). Für jeden Wert von n steht das linke Punktepaar für den Unterarm und das rechte für die Hand.

Die euklidische Distanz zwischen den ermittelten und den wahren Referenzpunkten liegt typischerweise zwischen 40 und 80 mm, für die dritte Sequenz bei 150 mm, wobei hier eine Zeigegeste durchgeführt wird (siehe Abb. 8.2 rechts). Dabei sind die Fehlerwerte unabhängig vom gewählten Wert für n und vergleichbar mit den Ergebnissen von Hahn et al. (2007). Außerdem sind die Abweichungen der Positionsschätzung vergleichbar mit denen von Ziegler et al. (2006). Die Ungenauigkeiten werden teilweise durch die Verschiebung des Hand-Unterarm-Modells entlang der Armachse verursacht (siehe dazu auch Kap. 4.4.9). Trotz sehr unterschiedlicher Probanden wurde eine gleichbleibende Arm- und Handlänge von 220 bzw. 180 mm verwendet. Für die drei Sequenzen liegen die wahren Längen des Unterarms bei 190, 190 und 217 mm und für die Hand bei 203, 193 und 128 mm. Insbesondere in der dritten Sequenz repräsentiert das Modell die Hand nur ungenau, da die Hand hier eine Faust mit einem ausgestrecken Finger bildet. Jedoch wird

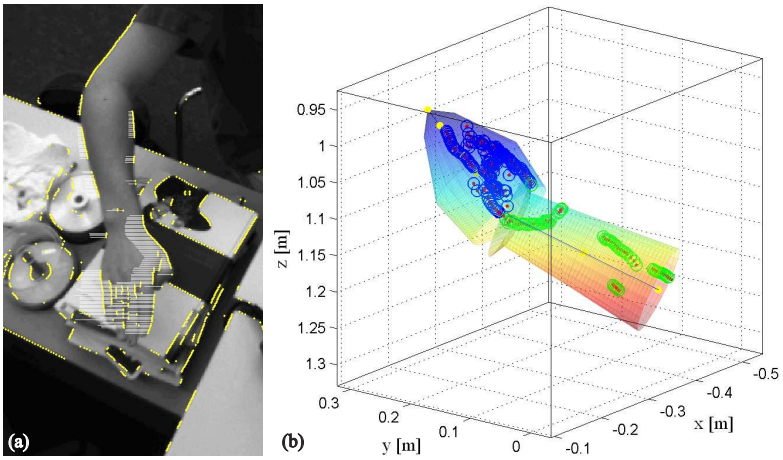


Abbildung 8.1.: Ergebnis des Spacetime Stereos und die Modellanpassung für die erste Testsequenz. (a) Ins Bild projizierte 3D-Punkte (gelb) und Bewegungsvektoren (weiß). (b) 3D-Punkte und 3D-Objektmodell.

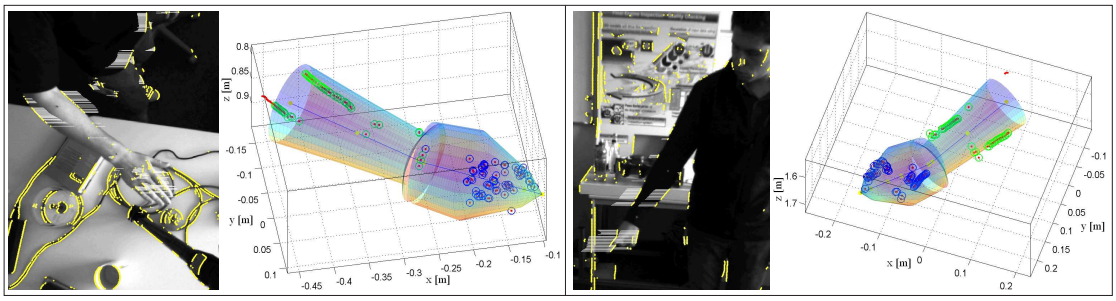


Abbildung 8.2.: Ergebnis des Spacetime Stereos und die Modellanpassung für die zweite (links) und die dritte (rechts) Testsequenz.

dadurch die Robustheit des Systems nicht beeinflusst. Die Evaluierung zeigt weiterhin, dass die Bewegung zwischen zwei aufeinanderfolgenden Bildpaaren mit einem typischen Fehler der translatorischen Genauigkeit von 1–3 mm unabhängig von der gewählten Zeitkonstante geschätzt wird, was vergleichbar mit der Pixelauflösung ist. Der typische Fehler für die Schätzung der rotatorischen Bewegung liegt bei 1–3°, ebenfalls unabhängig von der gewählten Zeitkonstante. Da die Hand im Vergleich zum Unterarm etwas kürzer und eher rund ist, entsteht für die Hand ein größerer Fehler bei der Schätzung der rotatorischen Bewegung, wobei sich die sehr großen Fehler für ω_p in der ersten Sequenz durch sporadische Ausreißer erklären lassen. Eine robuste Detektion und Poseschätzung der menschlichen Hand-Unterarm-Region ist durch das System für Zeitintervalle bis zu 800 ms ($n = 16$) möglich. Die Genauigkeit der Pose- und Bewegungsschätzung ist dabei größenteils unabhängig vom Wert für n.

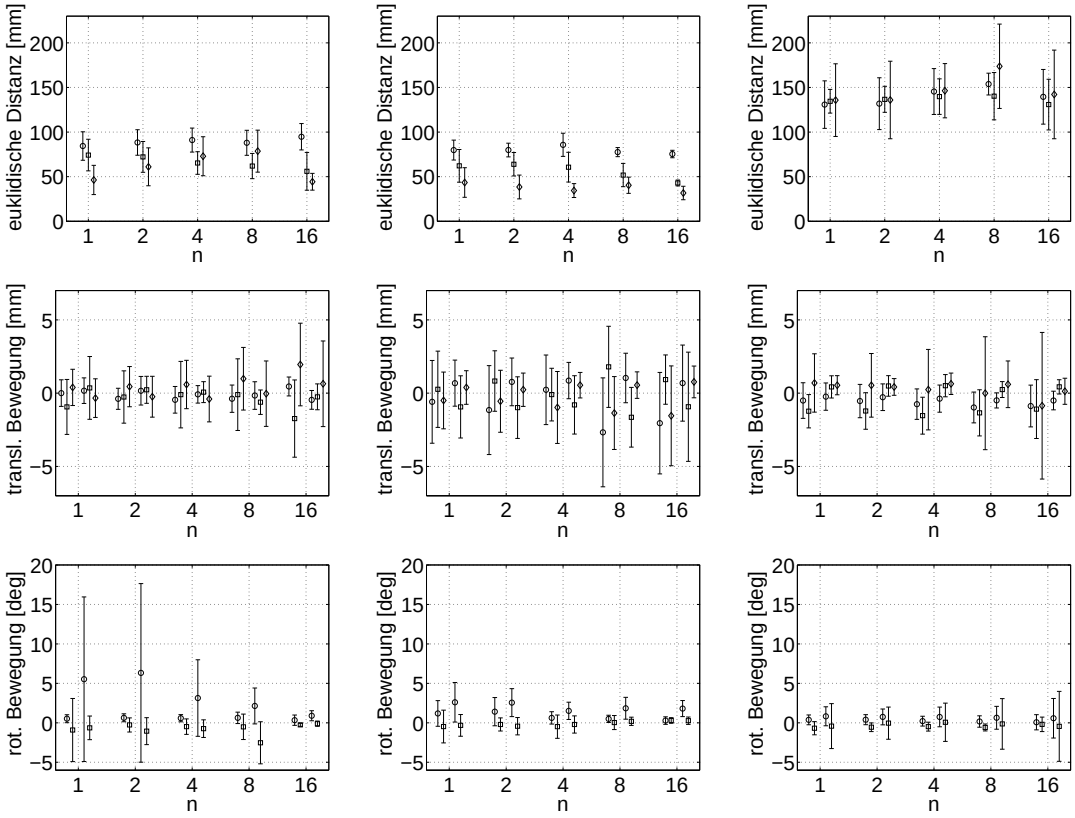


Abbildung 8.3.: Evaluationsergebnisse, spaltenweise dargestellt für die drei Testsequenzen. “Bewegung” bezeichnet hier die Poseänderung zwischen verschiedenen Zeitschritten und ist unabhängig von der gewählten Zeitkonstante und daher in [mm] bzw. [deg] angegeben. Für Details siehe Text Kap. 8.1.

8.2. Experimente zur Systemausprägung 2

In diesem Abschnitt wird quantitativ der Fusionsansatz aus Kap. 4.4.9 evaluiert und mit anderen Pose-Estimation-Algorithmen verglichen, wobei ein öffentlich zugänglicher Datensatz verwendet wird. Die Evaluation fand in Zusammenarbeit mit Markus Hahn statt.

8.2.1. Datensatz

Die Systeme werden anhand von neun Testsequenzen evaluiert, die fünf verschiedene Personen zeigen, die komplexe Bewegungsabläufe durchführen, wie sie im Produktionsszenario typisch sind. Die Sequenzen 1–5 zeigen verschiedene Personen, die in einer Büroumgebung arbeiten, wohingegen die Personen in den Sequenzen 6–9 in einem typischen Industrieumfeld arbeiten. Ein Beispielbild zusammen mit dem korrespondieren-

den raum-zeitlichen Pose-Estimation-Ergebnis aller Sequenzen zeigt Abb. 8.4. In allen Sequenzen ist der Hintergund stark strukturiert und der Kontrast zwischen der Person und dem Hintergund ist eher gering. Die Personen tragen verschiedene Arten von Kleidung mit langen und kurzen Ärmeln und mit oder ohne Handschuhe.

Die Bildsequenzen wurden mit einem trinokularen Farbkamerasystem (siehe Abb. 8.5) aufgenommen, wobei die vertikale und horizontale Basisbreite ca. 150 mm beträgt. Das Zeitintervall zwischen aufeinanderfolgenden Bildtripeln beträgt $\Delta t = 71$ ms. Jede Sequenz besteht aus mindestens 300 Bildtripeln. Die mittlere Distanz zwischen der Testperson und dem Kamerasystem variiert zwischen 2.7 m und 3.3 m. Die Ground-Truth Daten beschreiben die Positionen der Referenzpunkte im Weltkoordinatensystem, welche mit den Modellpunkten P_1, P_2 und P_3 korrespondieren.

8.2.2. Trackingsysteme

In diesem Abschnitt werden die evaluierten Systeme kurz vorgestellt. Abb. 8.6 zeigt die fünf Systeme unterteilt in neu entwickelte Systeme und Systeme die zum Vergleich herangezogen wurden.

System 1 – MOCCD-Ansatz: Das erste System basiert auf dem Ansatz von Hahn et al. (2007). Um die Objektverfolgung zu starten ist eine grobe, manuelle Initialisierung der Poseparameter für den ersten Zeitschritt notwendig. Im Trackingsystem werden drei Instanzen des MOCCD-Algorithmus (siehe Kap. 4.4.9) im Rahmen eines Multi-Hypothesen-Kalmanfilter-Systems verwendet. Jeder Instanz des MOCCD-Verfahrens ist in deren Kalmanfilter ein anderes kinematisches Modell hinterlegt, wobei eine andere Objektbewegung angenommen wird, z.B. konstante Geschwindigkeit oder konstante Beschleunigung. Ein Winner-Takes-All-Ansatz wählt das beste Ergebnis der drei Instanzen unter Beachtung verschiedener Kriterien aus.

System 2 – ShapeFlow-Ansatz: Das zweite Trackingsystem basiert auf dem ShapeFlow-Ansatz von Hahn et al. (2008b). Ebenso wie bei System 1 beginnt die Objektverfolgung mit einer benutzerdefinierten Pose. Anschließend wird zunächst die Pose des Objekts über den in System 1 beschriebenen MOCCD-Ansatz ermittelt und danach durch die raum-zeitliche Erweiterung des MOCCD-Ansatzes, den ShapeFlow-Algorithmus, die zeitliche Ableitung der Pose, die Bewegung ermittelt. Dazu wird eine raum-zeitliche Modellfunktion an die trinokularen Bilder aus drei Zeitschritten angepasst.

System 3 – ICP basierte Modellanpassung: Hier wird die in Kap. 5.1.1 vorgestellte Systemausprägung 1 verwendet.

System 4 – Fusion von ICP und MOCCD: Hier wird der Fusionsansatz wie in Kap. 4.4.9 beschrieben verwendet.

System 5 – Fusion von ICP, MOCCD und ShapeFlow: Hier wird gegenüber System 4 die Bewegungsschätzung noch über den ShapeFlow-Ansatz verfeinert. Dies ist detailliert in Kap. 4.5.2 beschrieben.

8.2.3. Ergebnisse der Evaluierung

Um eine Aussage über die Genauigkeit der einzelnen Verfahren liefern zu können, wird die mittlere euklidische Distanz zwischen der ermittelten 3D-Position der Referenzpunkte und den korrespondierenden Ground-Truth-Daten verwendet, sowie die entsprechende Standardabweichung dieser Distanzen. Die ermittelten zeitlichen Ableitungen der Poseparameter werden durch den mittleren Fehler und die Standardabweichung der drei einzelnen Bewegungskomponenten jedes Referenzpunkts ermittelt, wobei die diskrete zeitliche Ableitung der Ground-Truth-Positionen als Ground-Truth der Bewegungskomponenten herangezogen wird.

Entsprechend Abb. 8.7 ist der Positionsfehler für System 1 (MOCCD-Algorithmus) entsprechend der mittleren euklidischen Distanz zwischen gemessener und Ground-Truth-Position für die Referenzpunkte zwischen 50 und 100 mm für die Sequenzen 1–5 und zwischen 120 und 250 mm für Sequenz 6 bis 9. Die Standardabweichung liegt bei 50–100 mm. Bei manchen Sequenzen bricht das Tracking ab und das Objekt geht verloren (siehe Prozentzahlen über den Fehlerbalken).

Die mittlere Positionsgenauigkeit für System 2 (ShapeFlow-Algorithmus) auf Grauwertbildern (siehe Abb. 8.8) ist für alle Sequenzen etwas niedriger als für System 1, wobei die Anzahl der Bilder in der das Objekt noch gefunden wird stets höher oder zumindest vergleichbar ist. Die komponentenweisen, mittleren Geschwindigkeitsfehler für System 2 zeigen keinen systematischen Offset. Die Standardabweichungen sind leicht geringer für Sequenz 1–5 als für Sequenz 6–9, korrespondierend mit ca. 10 bzw. 15 mm pro Zeitschritt. Wenn Farbbilder verwendet werden, verringert sich der mittlere Positionsfehler auf weniger als 50 mm für die Sequenzen 1–5, wobei das Objekt in allen Bildern gefunden wird, jedoch bleibt der Fehler größenteils unverändert für die Sequenzen 6–9. Die Standardabweichung der komponentenweisen Geschwindigkeitsfehler sind ebenfalls fast unverändert gegenüber den Grauwertaufnahmen.

Für System 3 (ICP-Algorithmus + Bewegungsanalyse basierend auf Verschiebungsvektorfeldern) zeigt der mittlere Postionsfehler Werte zwischen 50 und 200 mm (siehe Abb. 8.10). Im Vergleich zu System 2 ist der Anteil der Bilder, in denen das Objekt verfolgt werden kann, geringer für Sequenz 1, 2, 5 und 6, jedoch höher für die „schwierigeren“ Sequenzen 7–9. Die Standardabweichung der komponentenweisen, lateralen Geschwindigkeitsfehler sind geringer als für System 2 und liegen bei 7–10 mm pro Zeitschritt für alle Sequenzen. Jedoch ermittelt System 3 nicht die Geschwindigkeitskomponente entlang der optischen Achse des Kamerasystems.

Die Verfolgung des Objekts über alle Bider hinweg erfolgt mittels System 4 (Fusion von ICP und MOCCD, siehe Abb. 8.11). Der mittlere Positionsfehler beträgt 40–70 mm für die Sequenzen 1–5 und 80–110 mm für die Sequenzen 6–9. Die Standardabweichung der komponentenweisen, lateralen Geschwindigkeitsfehler beträgt 6–10 mm pro Zeitschritt. System 4 ermittelt wiederum nicht die Geschwindigkeitskomponente entlang der

optischen Achse.

Die Genauigkeiten von System 5 (Fusion von ICP und ShapeFlow) ist sehr ähnlich der von System 4 (siehe Abb. 8.12). Wiederum wird das Objekt in allen Sequenzen über alle Bilder hinweg verfolgt. Als zusätzliche Information liefert System 5 die Geschwindigkeitskomponente entlang der optischen Achse des Kamerasystems, welche mit einer Genauigkeit von 8–15 mm pro Zeitschritt ermittelt wird.

Es ist hervorzuheben, dass trotz der kleinen Basisbreite des trinokularen Kamerasystems, ist der Messfehler der Geschwindigkeit entlang der optischen Achse nahezu identisch mit denen der lateralen Geschwindigkeitskomponenten ist. Abb. 8.13 zeigt die ermittelten Geschwindigkeitskomponenten des Referenzpunktes am Handgelenk (blaue Kurven) und die korrespondierenden Ground-Truth Werte (Bezeichnung “GT”, rote Kurven) für einen Teil von Sequenz 4. Dies zeigt die Genauigkeit der instantanen Bewegungsinformation, welche durch System 5 ermittelt wird.

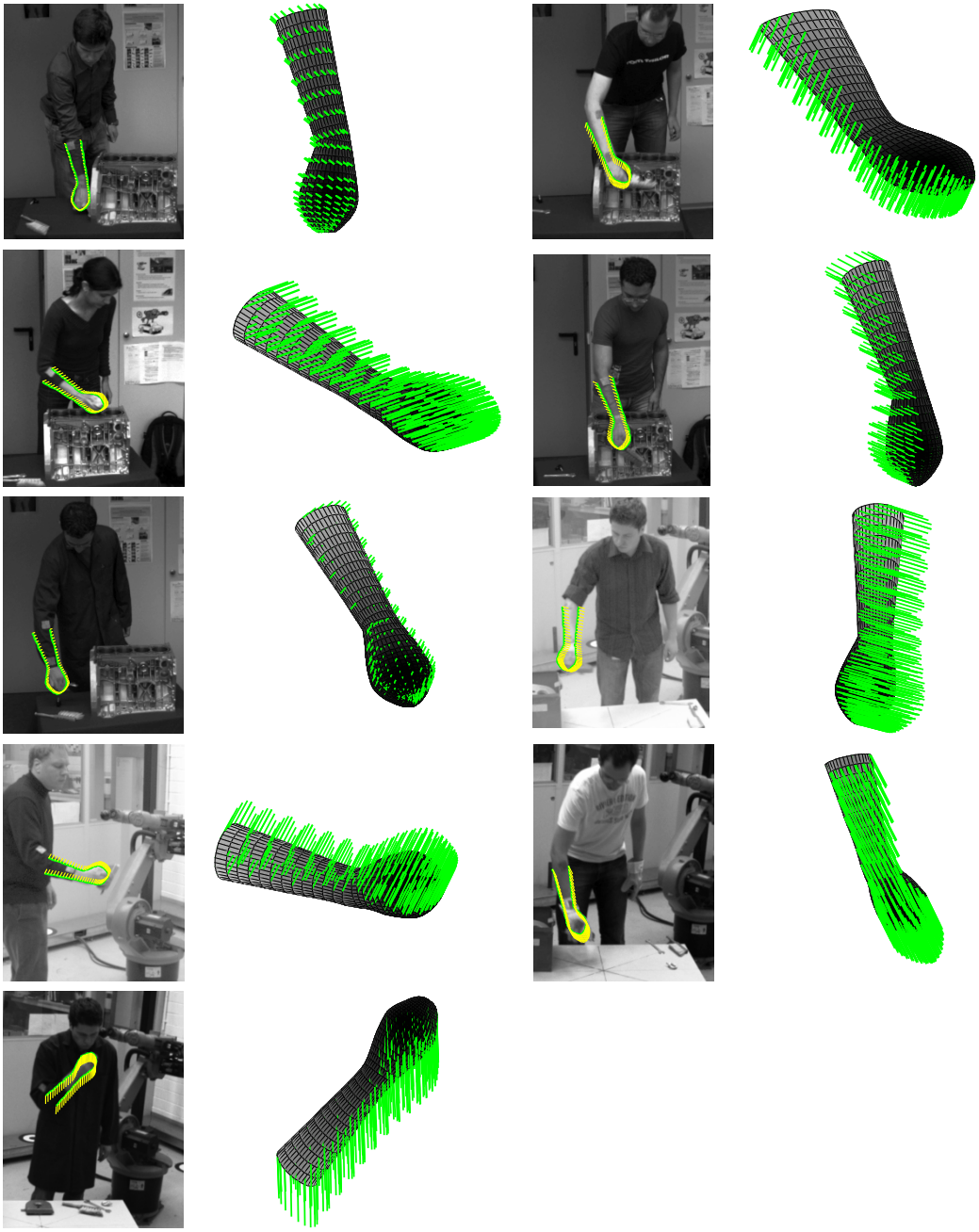


Abbildung 8.4.: Darstellung der Testsequenzen und der Ergebnisse der Fusion. Jeweils links ist ein Beispielbild der Sequenz mit der ins Bild projizierten Poseschätzung gezeigt. Rechts davon ist jeweils das 3D-Hand-Unterarm-Modell in der entsprechenden Pose gezeigt. Außerdem sind Szenenflussvektoren auf der Modelloberfläche entsprechend der ermittelten Objektbewegung dargestellt.

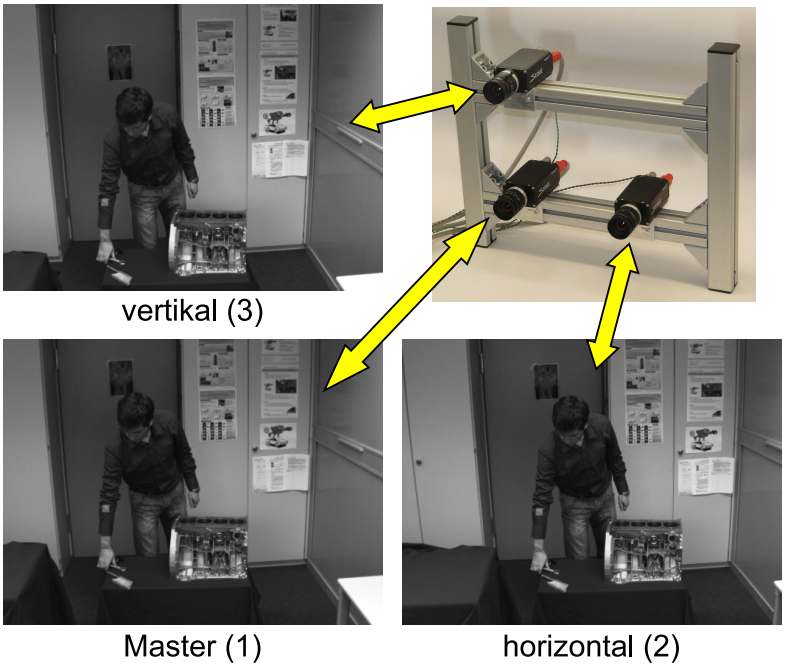


Abbildung 8.5.: Anordnung des trinokularen Kamerasystems.

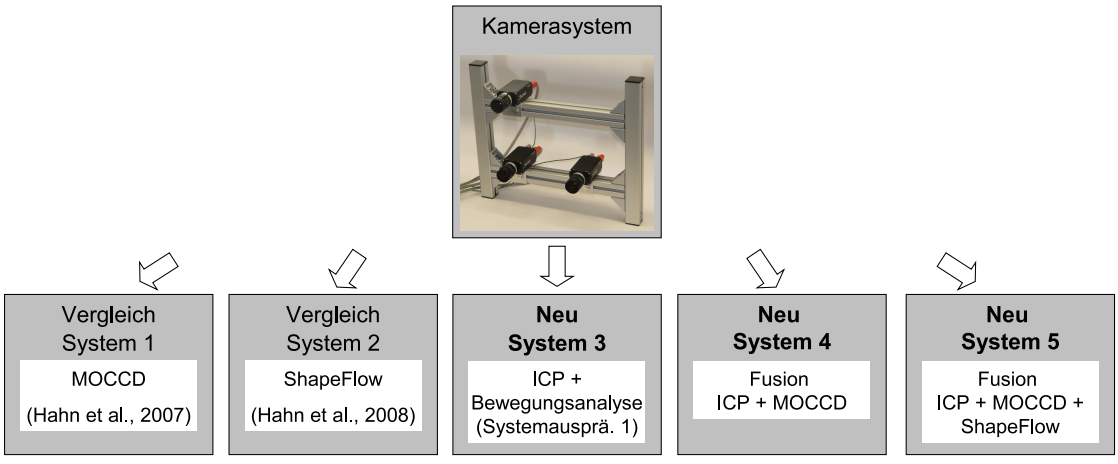


Abbildung 8.6.: Evaluierungssysteme unterteilt in neue entwickelte Systeme und Systeme die zum Vergleich herangezogen wurden.

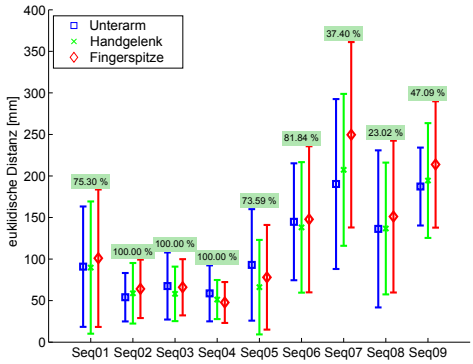


Abbildung 8.7.: 3D Pose-Estimation-Ergebnis für System 1. Die Prozentzahlen oberhalb der Fehlerbalken gibt den Anteil der Sequenz an, bis wohin eine Objektverfolgung möglich war.

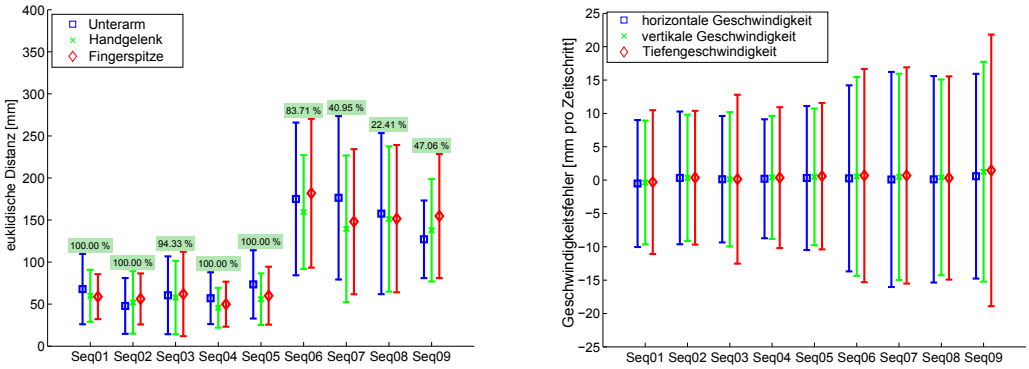


Abbildung 8.8.: Raum-zeitliche 3D Pose-Estimation-Ergebnisse für System 2.

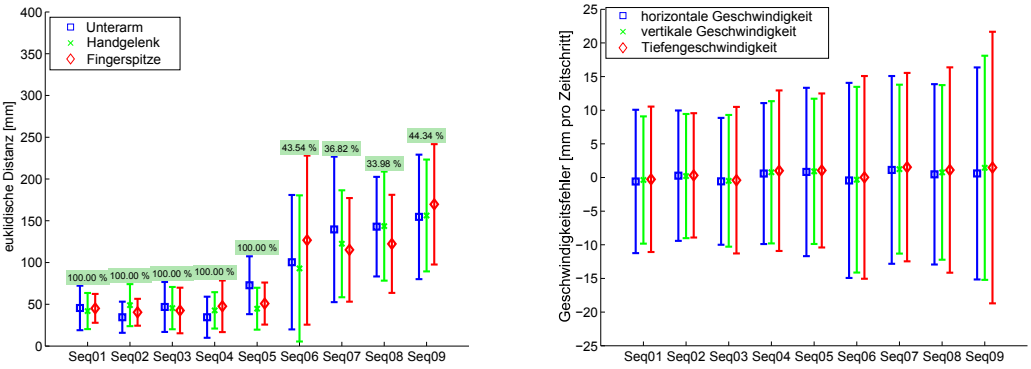


Abbildung 8.9.: Raum-zeitliche 3D Pose-Estimation-Ergebnisse für System 2 unter Verwendung von Farbinformationen.

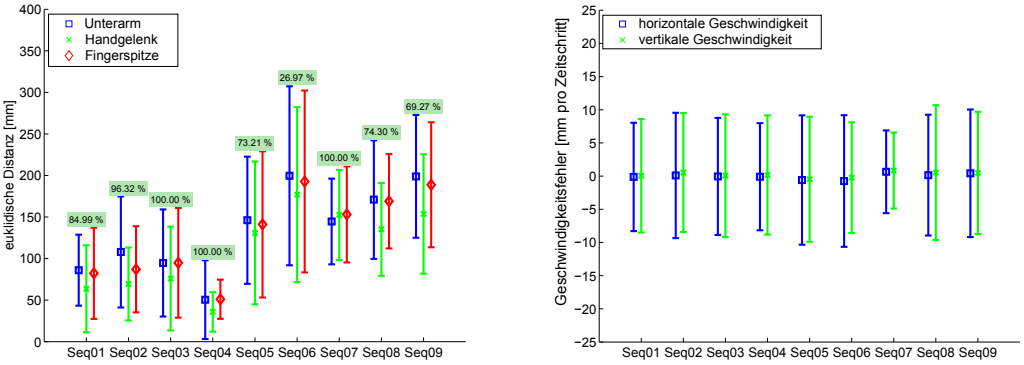


Abbildung 8.10.: Raum-zeitliche 3D Pose-Estimation-Ergebnisse für System 3.

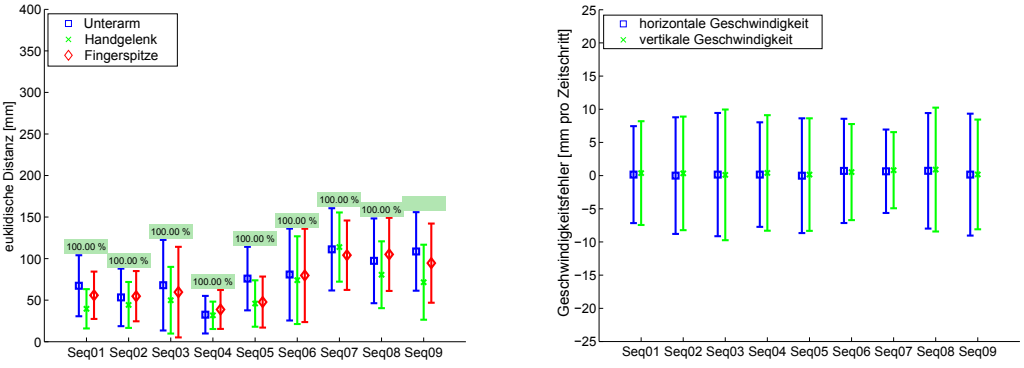


Abbildung 8.11.: Raum-zeitliche 3D Pose-Estimation-Ergebnisse für System 4.

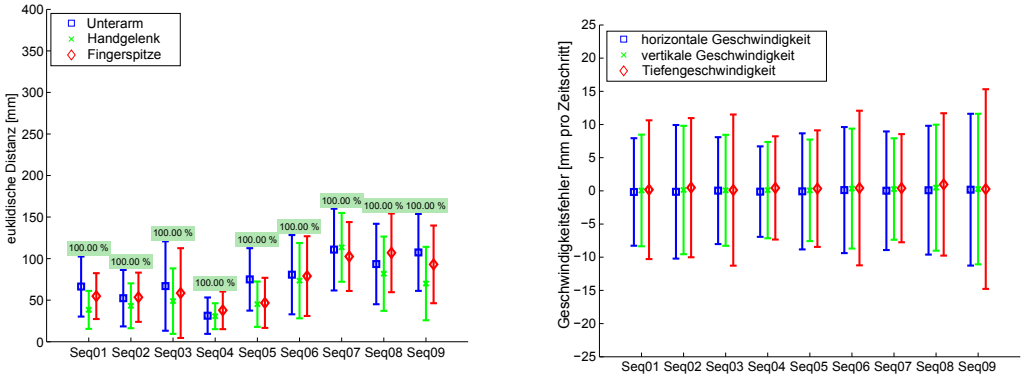


Abbildung 8.12.: Raum-zeitliche 3D Pose-Estimation-Ergebnisse für System 5.

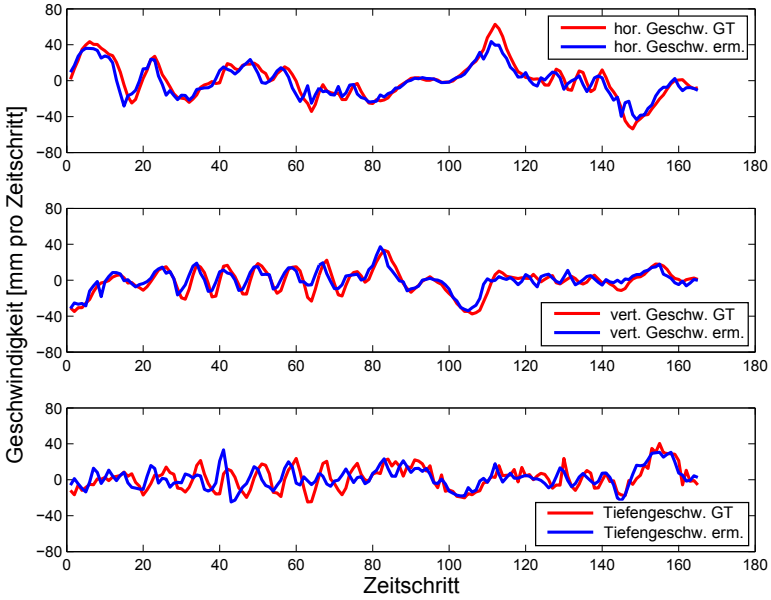


Abbildung 8.13.: Die instantan ermittelten Geschwindigkeitskomponenten $\dot{x}$ (oben), $\dot{y}$ (mitte) und $\dot{z}$ (unten) des Referenzpunkts am Handgelenk unter Verwendung von System 5. Die Kennzeichnung “GT” steht für Ground-Truth.

9. Untersuchungen im Straßenverkehrsszenario

9.1. Experimente zur Systemausprägung 3

Zur Untersuchung der Genauigkeiten der Poseschätzung im Straßenverkehrsszenario werden sieben verschiedene Sequenzen benutzt, die typische Kreuzungsszenarien darstellen: ein Fahrzeug fährt geradeaus über die Kreuzung und ein Fahrzeug, welches links oder rechts abbiegt und jeweils noch mit unterschiedlichen Geschwindigkeiten. Zur Bildaufnahme wurden drei Farbkameras mit einer Auflösung von 1034×776 Pixel verwendet. Die Kameras wurden nebeneinander mit verschiedenen Abständen angeordnet, was zu den drei Basisbreiten von 102 mm, 228 mm und 380 mm führt. Die Bildwiederholrate liegt für alle Sequenzen bei 14 Bildern pro Sekunde. Die Farbinformationen werden ausschließlich für die Ground-Truth-Berechnung (siehe Kap. 7) benutzt, wohingegen die Bilder für die Stereoalgorithmen zu Grauwertbildern gewandelt werden. Evaluiert wird die Genauigkeit der Poseschätzung in Hinblick auf verschiedene Konfigurationen des Systems: drei verschiedene Basisbreiten, drei verschiedene Stereoalgorithmen, zwei verschiedene Optimierungsansätze und der Unterschied zwischen der klassischen, euklidischen Fehlermetrik und der angepassten, polaren Fehlermetrik werden untersucht.

Die Evaluierung basiert auf zwei geometrischen Indikatoren: der Gierwinkelunterschied $\Delta\Theta$ zwischen dem wahren Gierwinkel und dem Ergebnis der Poseschätzung und der mittleren Distanz zwischen den Ground-Truth-Punkten sowie der korrespondierenden Fläche. Die Abweichungen werden mittels Fehlerbalken dargestellt, welche auf dem jeweiligen Median der ganzen Sequenz und dem 25% und dem 75% Quantil bestehen. Für die meisten Sequenzen bietet die Pose-Estimation ein sinnvolles Ergebnis für ca. 90% der Zeitschritte. Lediglich in Sequenz 6, wo das Fahrzeug fast parallel der z-Achse fährt, führt dies zu einer geringen Ausdehnung der Punktewolke in x Richtung, was zu ungenauen Poseergebnissen führt. Es werden in den Abbildungen jeweils zunächst die Ergebnisse für alle Zeitschritte dargestellt. Darunter ist jeweils die Ergebnisse gefiltert dargestellt, d.h. nur für die Zeitschritte, in denen ein sinnvolles Poseergebnis geschätzt werden konnte.

9.1.1. Basisbreiten

Im ersten Teil der Evaluierung wird das Systemverhalten bei verschiedenen Basisbreiten des Kamerasystems analysiert. Abb. 9.1 zeigt den Gierwinkelfehler, Abb. 9.2 den Distanzfehler für die drei verschiedenen Basisbreiten für alle sieben Sequenzen, wobei drei der in Abschnitt 4.2 beschriebenen Stereoansätze zum Einsatz kommen: Census-Stereo + Census-Fluss, Korrelationsstereo + Census-Fluss und das Spacetime Stereo. Der Ver-

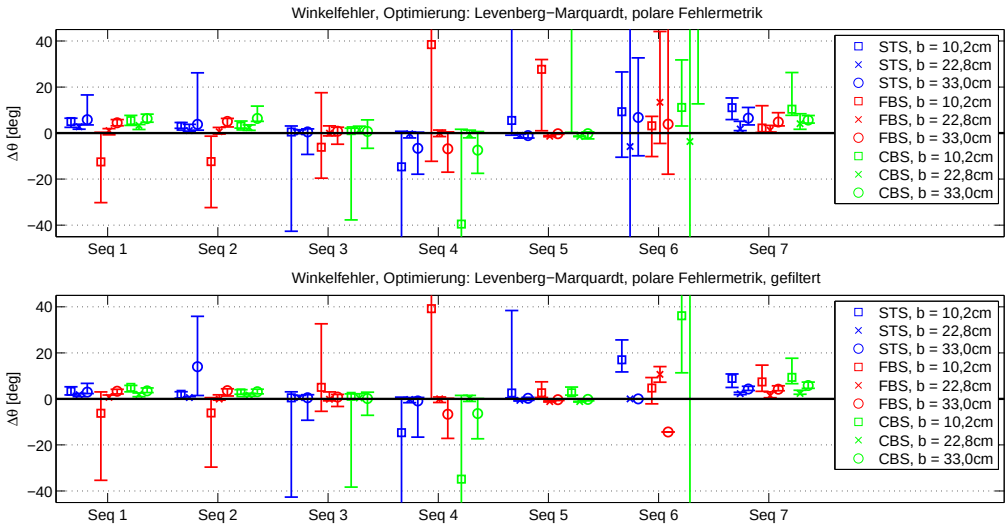


Abbildung 9.1.: Abhängigkeit des Gierwinkelfehlers von der Basisbreite, Levenberg-Marquardt Optimierung, polare Fehlermetrik. Blau beschreibt das Spacetime Stereo (STS), rot das Census-Stereo (FBS) und grün das Korrelationsstereo (CBS). Die kleine Basisbreite ist durch Quadrate gekennzeichnet, die mittlere Basisbreite durch Kreuze und die große Basisbreite ist durch Kreise gekennzeichnet. Die mittlere Basisbreite liefert in Bezug auf den Gierwinkelfehler die besten Ergebnisse. Oben: Ergebnisse für alle Zeitschritte einer Sequenz. Unten: Ergebnisse der Zeitschritte, in denen ein sinnvolles Poseergebnis ermittelt werden konnte.

gleich zwischen den verschiedenen Basisbreiten zeigt, dass die kleinste Basisbreite nicht ausreichend für diese Anwendung ist. Insbesondere für Fahrzeuge, die vor der Kamera links oder rechts abbiegen (Sequenz 3 und 4) ergeben sich hohe Ungenauigkeiten bei der Pose-Estimation. Die größte Basisbreite führt ebenfalls zu großen Fehlern, da durch den großen Abstand zwischen den Kameras und dem daraus resultierenden Parallaxeneffekt der Stereoalgorithmus weniger korrekte Punkte produziert.

9.1.2. Stereoalgorithmus

Um die verschiedenen Steroalgorithmen miteinander vergleichen zu können, werden alle Sequenzen mit der mittleren Basisbreite und dem Levenberg-Marquardt-Algorithmus berechnet. Abb. 9.3 und Abb. 9.4 zeigen die Ergebnisse der verschiedenen Stereo-Algorithmen, wobei beiden Fehlermetriken (euklidisch und polar) evaluiert wurden.

Die Ergebnisse des Gierwinkelfehlers zeigen einen vernachlässigbaren Unterschied zwischen dem pixelgenauen Census-Stereo und den anderen beiden Stereoverfahren mit Subpixelgenauigkeit. Die Ergebnisse sind ähnlich für alle drei Verfahren.

Bezogen auf die Position jedoch kann ein Unterschied zwischen pixel-genauem und

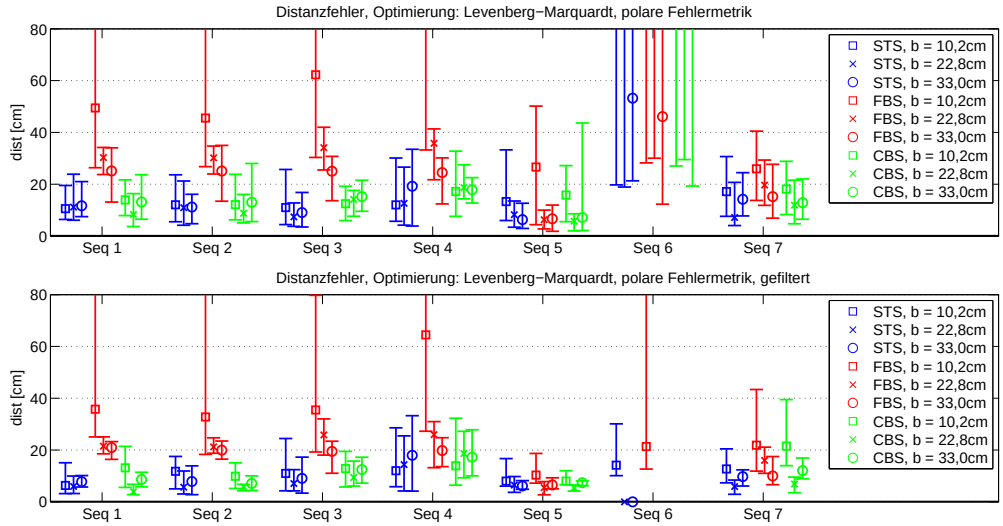


Abbildung 9.2.: Abhängigkeit des Abstandsfehlers von der Basisbreite, Levenberg-Marquardt Optimierung, polare Fehlermetrik. Anordnung, Farben und Kennzeichnungen wie in Abb. 9.1.

subpixel-genauem Stereoverfahren festgestellt werden. Die Abstände des Census-Stereo Ergebnisses zeigen einen bis zu dreimal höheren Fehler für Sequenz 1, 2 und 3. Zwischen Spacetime Stereo und Korrelations-Stereo kann kein signifikanter Unterschied festgestellt werden.

Um einen Vergleich mit einem dichten Stereo-Verfahren zu erhalten, bei dem für nahezu alle Pixel im Bild ein Tiefenwert verfügbar ist, wurden 3D-Punkte für drei der Sequenzen (Sequenz 1, 3 und 9) mit dem Semi-Global-Matching-Verfahren (Hirschmüller, 2005) berechnet. Anschließend wurde die gleiche Verarbeitungskette mit polarer Fehlermetrik durchlaufen. Es kann festgestellt werden, dass bei dem dichten Verfahren, welches subpixelgenaue Korrespondenzen bildet, ähnliche Ungenauigkeiten wie bei den anderen beiden subpixelgenauen Verfahren auftreten. So liegt der Winkelfehler für die drei Sequenzen bei $1.5° \pm 1.3°$. Für den Distanzfehler werden Werte von 10.4 cm $\pm$4 cm ermittelt. Somit sind beide Fehler vergleichbar bzw. leicht höher als beim Korrelations- oder Spacetime-Stereo. Nachteilig ist jedoch, dass durch das dichte Verfahren eine weitaus höhere Anzahl von 3D-Punkten generiert wird. Diese Punkte müssen segmentiert werden und auch die Poseschätzung benutzt eine weitaus höhere Zahl von Objektpunkten. Dies bedeutet einen höheren Berechnungsaufwand und somit eine Verlangsamung der gesamten Verarbeitungskette, wobei sich jedoch die Genauigkeiten nicht verbessern.

9.1.3. Fehlermetrik

Abb. 9.3 und Abb. 9.4 zeigen die Pose-Estimation-Fehler für die beiden Fehlermetriken. Für den Positionsfehler zeigen beide Metriken ein ähnliches Verhalten, wohingegen die

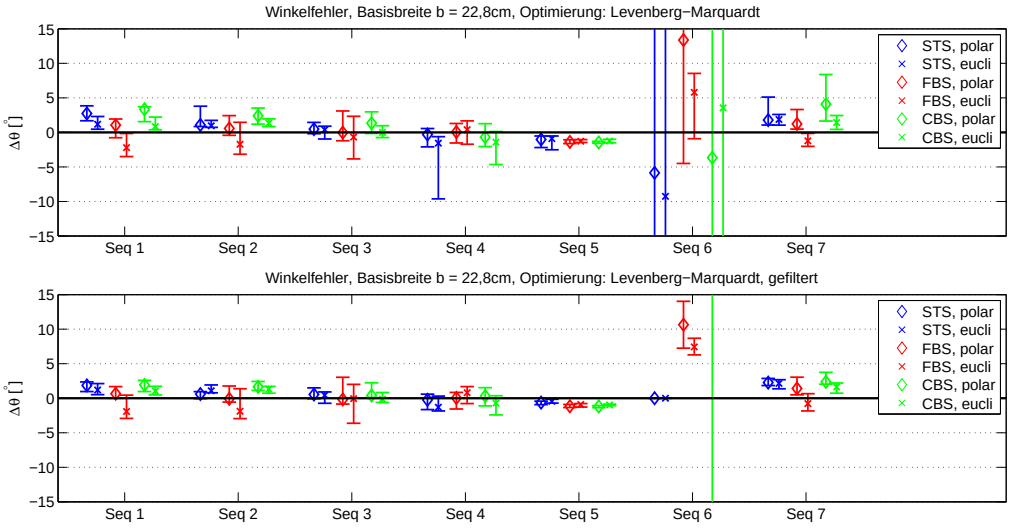


Abbildung 9.3.: Abhängigkeit des Gierwinkelfehlers in Bezug auf den Stereo-Algorithmus, Levenberg-Marquardt Optimierung, mitttlere Basisbreite. Die Farben (blau, rot und grün) kennzeichnen die Stereo-Algorithmen, wobei Rauten für die polare und Sterne für die euklidische Fehlermetrik stehen. Oben: Ergebnisse für alle Zeitschritte einer Sequenz. Unten: Ergebnisse der Zeitschritte, in denen ein sinnvolles Poseergebnis ermittelt werden konnte.

polare Fehlermetrik einen geringen Gierwinkelfehler für die meisten Sequenzen erzeugt. Die Berechnungszeit ist dabei für beide Metriken ähnlich.

9.1.4. Optimierungsalgorithmus

Das Ergebnis der Downhill-Simplex Optimierung wird in Abb. 9.5 für den Gierwinkelfehler und in Abb. 9.6 für den Positionsfehler gezeigt.

Die Gierwinkelfehler sind alle höher für den Downhill-Simplex-Algorithmus. Die Anzahl der Funktionsaufrufe liegen für den Levenberg-Marquardt-Algorithmus in der gleichen Größenordnung (ungefähr 100 Funktionsaufrufe) wie für den Downhill-Simplex-Algorithmus, jedoch mit einer besseren Pose-Estimation-Genauigkeit. Daher ist in diesem Fall der Levenberg-Marquardt-Algorithmus vorzuziehen, auch weil sich bei der Positionsgenauigkeit ein ähnliches Verhalten zeigt.

9.2. Experimente zur Systemausprägung 4

Die Datenbasis für die Evaluierung von Systemausprägung 4 ist gleich der für Systemausprägung 3.

Für die Bilder, die in Standardstereogeometrie rektifiziert wurden, ergibt sich eine

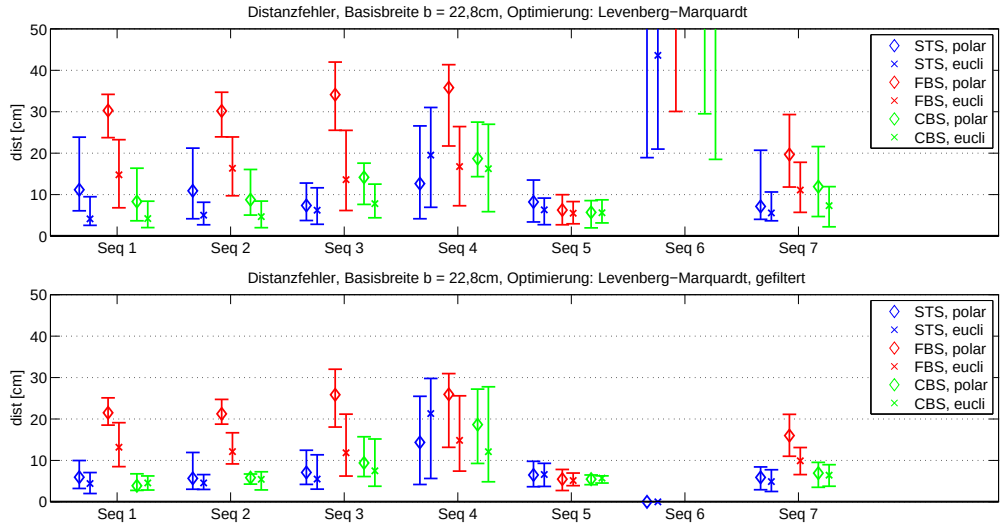


Abbildung 9.4.: Abhängigkeit des Abstandsfehlers in Bezug auf den verwendeten Stereo-Algorithmus, Levenberg-Marquardt Optimierung, mittlere Basisbreite. Anordnung, Farben und Kennzeichnungen wie in Abb. 9.3.

Kamerakonstante von $f = 1350$ Pixel, sodass bei einer Basisbreite von $b = 0.228$ m sich $bf = 307.8$ m Pixel ergibt. Laut Krüger und Wöhler (2009) können die Positionen zweier Schachbrettecken (d.h. der Disparitätswert) mit einer Standardabweichung von 0.032 Ppixel gemessen werden. Bei einer typischen Objektdistanz zur Kamera von $z_0 = 20$ m beträgt die Disparität $bf/z_0 = 15.39$ Pixel, was gemäß Fehlerfortpflanzungsgesetz zu einem Tiefendifferenzfehler von 0.106 m führt. Die Distanz in der Szene zwischen den Markern, welche auf den Seiten des Fahrzeugs angebracht wurden, liegt bei ca. 3 m, sodass der maximale Fehler für den Gierwinkel, wenn das Fahrzeug lateral zur optischen Achse fährt, bei ca. 2° liegt. Die absoluten Tiefen- und Gierwinkelfehler sind proportional zu z_0^2.

Abb. 9.7 zeigt eine Beispieltrajektorie und den ermittelten Gierwinkel für Sequenz S5. Die Referenztrajektorie wurde durch die mittlere Position der vier Farbmarker ermittelt. Die ermittelte Trajektorie wurde basierend auf dem Mittelpunkt der korrespondierenden Modellfläche generiert.

Die Werte, welche die Posegenauigkeit beschreiben, sind, wie bereits bei den Experimenten zu Systemausprägung 3 eingeführt, die mittlere euklidische Distanz e_{dist} der Referenzpunkte zu der korrespondierenden Modellebene und die Differenz $\Delta\theta$ zwischen dem ermittelten Gierwinkel und seinem Referenzwert aus den Ground-Truth-Daten. Die Referenzdaten für die zeitlichen Poseableitungen werden durch numerische Differentiation der Posereferenzdaten unter Berücksichtigung der Zeitschritte erzeugt. Da die resultierende Gierrate zu rauschbehaftet ist, um verlässliche Referenzwerte zu liefern, wird die Gierrate tiefpassgefiltert, indem ein Polynom vierten Grades an die differenzierten Gierwinkelwerte angepasst wird. In Abb. 9.8 und Abb. 9.9 werden die Unterschiede zwi-

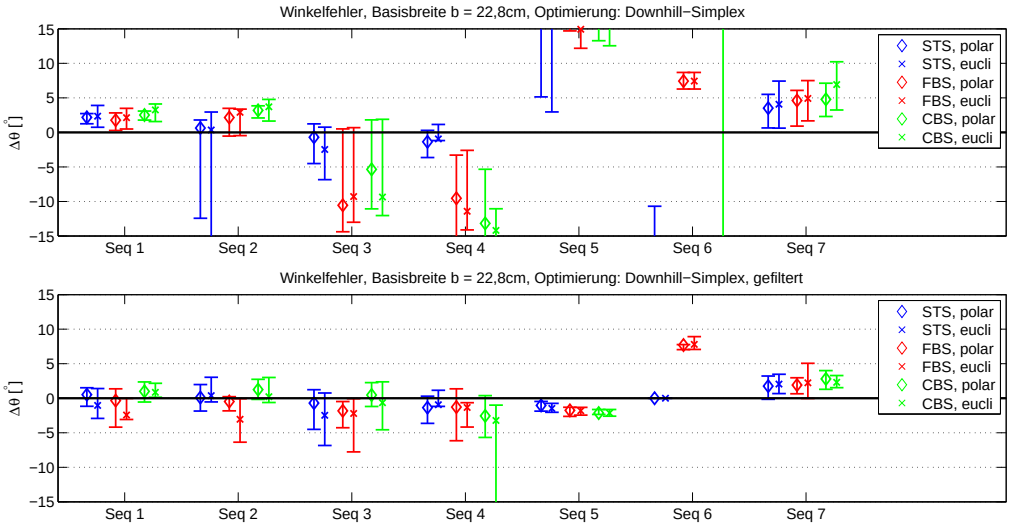


Abbildung 9.5.: Gierwinkelfehler für die Downhill-Simplex Optimierung, mittlere Basisbreite. Anordnung, Farben und Kennzeichnungen wie in Abb. 9.3.

schen ermittelten Poseparametern und den zugehörigen Referenzwerten als Fehlerbalken dargestellt, die den Median und das 25% und das 75% Quantil beschreiben. Der modellbasierte Szenenfluss wird stets über das Ergebnis des Quaternionen-ICPs initialisiert.

Der mittlere Gierwinkelfehler $\Delta\theta$ des nichtlinearen ICP-Ansatzes mit polarer Fehlermetrik ist deutlich geringer als der der Quaternionenbasierten Lösung, d.h. $2.4° \pm 2.1°$ gegen $4.3° \pm 6.3°$ (siehe Abb. 9.8, oben). Der mittlere Gierwinkelfehler des modellbasierten Szenenflusses liegt bei $1.8° \pm 3.4°$. Das relativ große Fehlerintervall wird größtenteils durch die hohen Unsicherheiten in Sequenz S6 verursacht, wohingegen die Fehler in den anderen Sequenzen vergleichbar mit denen des nichtlinearen ICP-Ansatzes sind. Der mittlere Positionsfehler e_{dist} des quaternionenbasierten ICP-Algorithmus ist um ein Drittel geringer als beim nichtlinearen ICP (0.21 ± 0.07 m vs. 0.31 ± 0.08 m). Der modellbasierte Szenenfluss liefert einen vergleichbaren Mittelwert für e_{dist} von 0.19 ± 0.11 m (siehe Medianwerte in Abb. 9.8, unten).

Die Genauigkeit $\Delta\dot{\theta}$ der durch den modellbasierten Szenenfluss-Ansatz bestimmten Gierrate, gemittelt über alle Testsequenzen korrespondiert mit $1.2° \pm 1.3°$ pro Zeitschritt (siehe Abb. 9.9). Der mittlere horizontale Geschwindigkeitsfehler $\Delta\dot{x}$ beträgt 0.06 ± 0.07 m pro Zeitschritt, wohingegen der mittlere Fehler $\Delta\dot{z}$ der Geschwindigkeitskomponente entlang der optischen Achse höher ist: 0.11 ± 0.09 m pro Zeitschritt. Für den modellbasierten Szenenfluss zeigt die Evaluierung keinen signifikanten Unterschied zwischen der Verwendung der mittelwertfreien SSD und dem normierten Kreuzkorrelationskoeffizienten als Vergleichsmaß. Vergleichsweise hohe Fehler tauchen vor allem in Sequenz S4, S5 und S6 auf, wo in den meisten Bildern das Fahrzeug weit entfernt von der Kamera ist($z \approx 30$ m) und lediglich die Hinterseite sichtbar ist. Vergleicht man die geschätzten Bewegungsparametern mit denen, die sich differentiell aus den Poseergebnis-

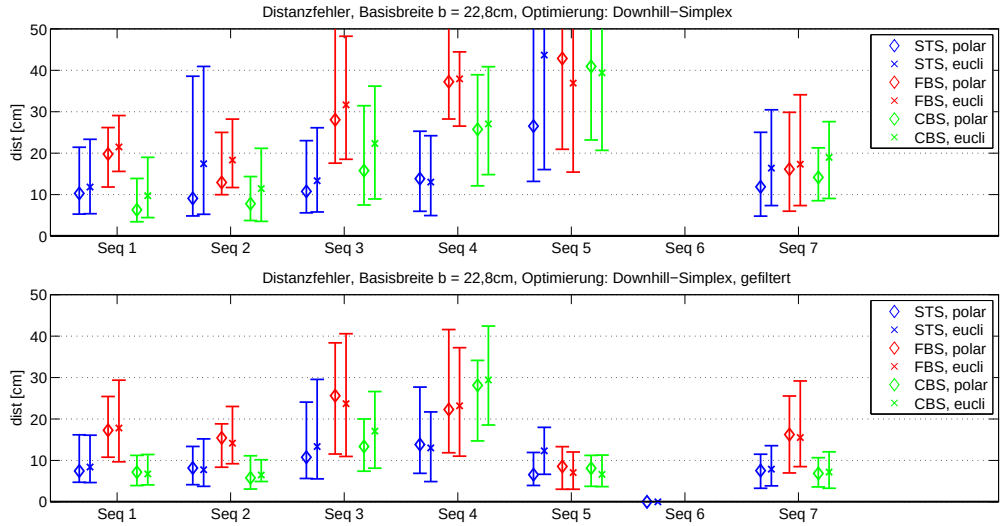


Abbildung 9.6.: Abstandsfehler für die Downhill-Simplex Optimierung, mittlere Basisbreite. Anordnung, Farben und Kennzeichnungen wie in Abb. 9.3.

sen ermitteln lassen, kann festgestellt werden, dass der Fehler der differentiell geschätzten Bewegungsparameter verglichen mit der Ground-Truth etwa dreimal größer ist, als der Fehler der direkt mithilfe des modellbasierten Szenenfluss-Verfahrens geschätzten Bewegungsparameter. Daher ist eine direkte Schätzung der Objektbewegung differentiell ermittelten Bewegungsdaten vorzuziehen.

Als weiteres Beispiel wurde eine kurze Bildsequenz von 1.5 s Länge betrachtet, worin ein PKW einen Abbiegevorgang durchführt (siehe Abb. 9.10). Hier ist die Bildgröße 640×480 Pixel, die Bildwiederholfrequenz liegt bei 12 Bildern pro Sekunde, und die Basisbreite des Stereokamerasystems liegt bei 300 mm. Im beobachteten Szenario konnten keine Farbmarker am Fahrzeug angebracht werden, wodurch keine Referenzdaten zur Verfügung stehen. Die x- und z-Position und der Gierwinkel θ, welche durch den modellbasierten Szenenfluss ermittelt wurden, zeigen, dass das Fahrzeug lateral von rechts nach links in Richtung der Kamera fährt und sich dabei um 25° dreht, was einer mittleren Gierrate von 1.4° pro Zeitschritt entspricht (siehe Abb. 9.11). Die Messungen der direkt aus jeweils zwei Zeitschritten ermittelten Gierrate sind stark rauschbehaftet, zeigen aber einen konsistent abnehmenden Wert von 1.5° auf 1.0° pro Zeitschritt. Die gemessene, instantan ermittelte, laterale Geschwindigkeit $\dot{x}$ von 0.14–0.18 m pro Zeitschritt ist konsistent mit der lateralen Position x über die Zeit betrachtet. Die Tiefengeschwindigkeit rauscht stark, zeigt jedoch, dass der modellbasierte Szenenfluss dazu in der Lage ist, kleine Geschwindigkeitskomponenten entlang der z-Achse von nur ein paar Zentimetern pro Zeitschritt zu erkennen, was weniger als einem Prozent des Objektabstands zur Kamera entspricht. Am Ende der Sequenz sind die Tiefengeschwindigkeitswerte nahezu Null, da sich das Fahrzeug nun fast ausschließlich lateral zur Kamera bewegt.

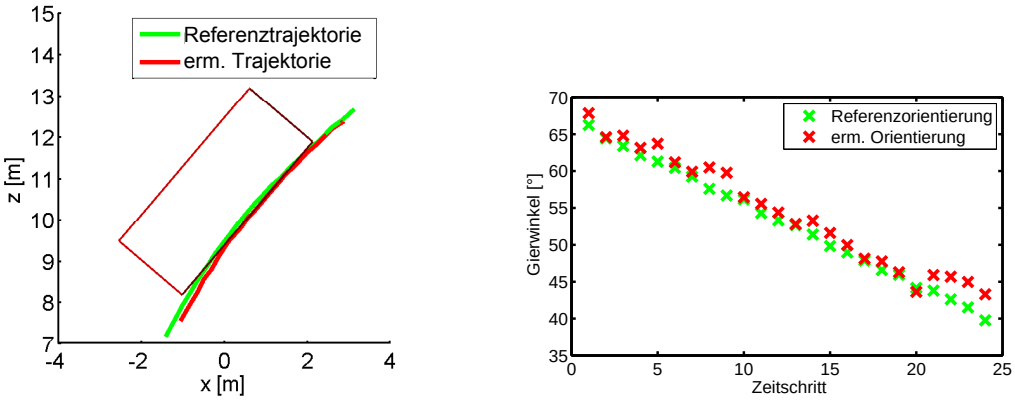


Abbildung 9.7.: Links: Trajektorie für Sequenz S5 mit dem Fahrzeugmodell für einen Zeitschritt, wobei der Mittelpunkt der sichtbaren Modellebene dargestellt ist. Rechts: Verhalten des Gierwinkels θ für die gleiche Sequenz.

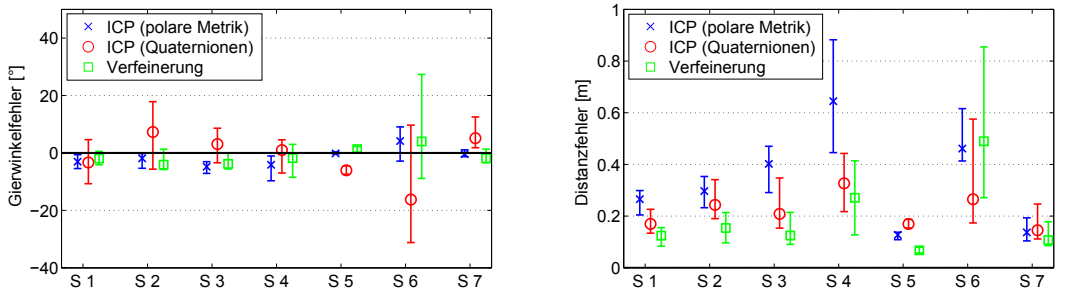


Abbildung 9.8.: Gierwinkelfehler $\Delta\theta$ (links) und Positionsfehler δ (rechts). Die Symbole kennzeichen die Medianwerte und die Fehlerbalken das 25% bzw. 75% Quantil. “Verfeinerung” kennzeichnet das Ergebnis des modellbasierten Szenenflusses.

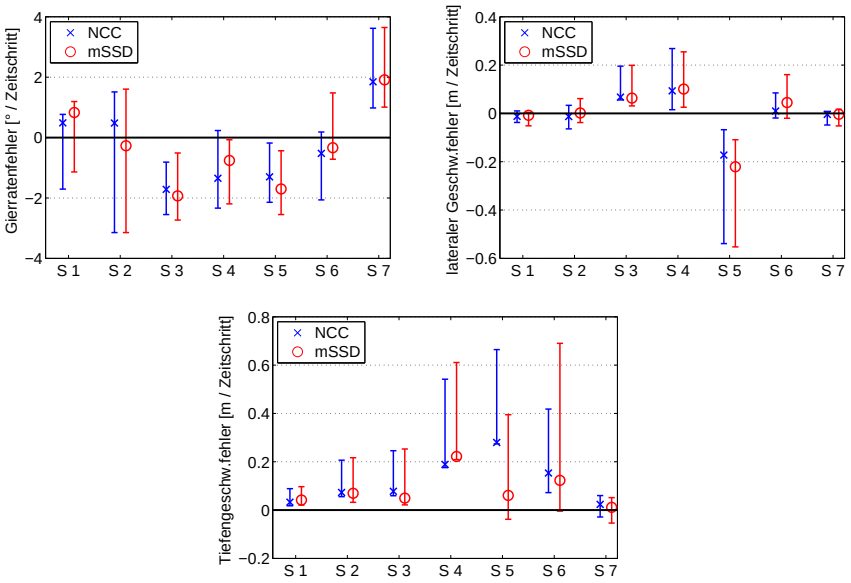


Abbildung 9.9.: Fehler der Gierrate ($\Delta\dot{\theta}$, oben links), Horizontalgeschwindigkeit ($\Delta\dot{x}$, oben rechts) und der Tiefengeschwindigkeit ($\Delta\dot{z}$, unten). Die Symbole kennzeichen die Medianwerte und die Fehlerbalken das 25% bzw. 75% Quantil.

Abbildung 9.10.: Beispielsequenz, in der ein PKW rechts um eine Kurve fährt.

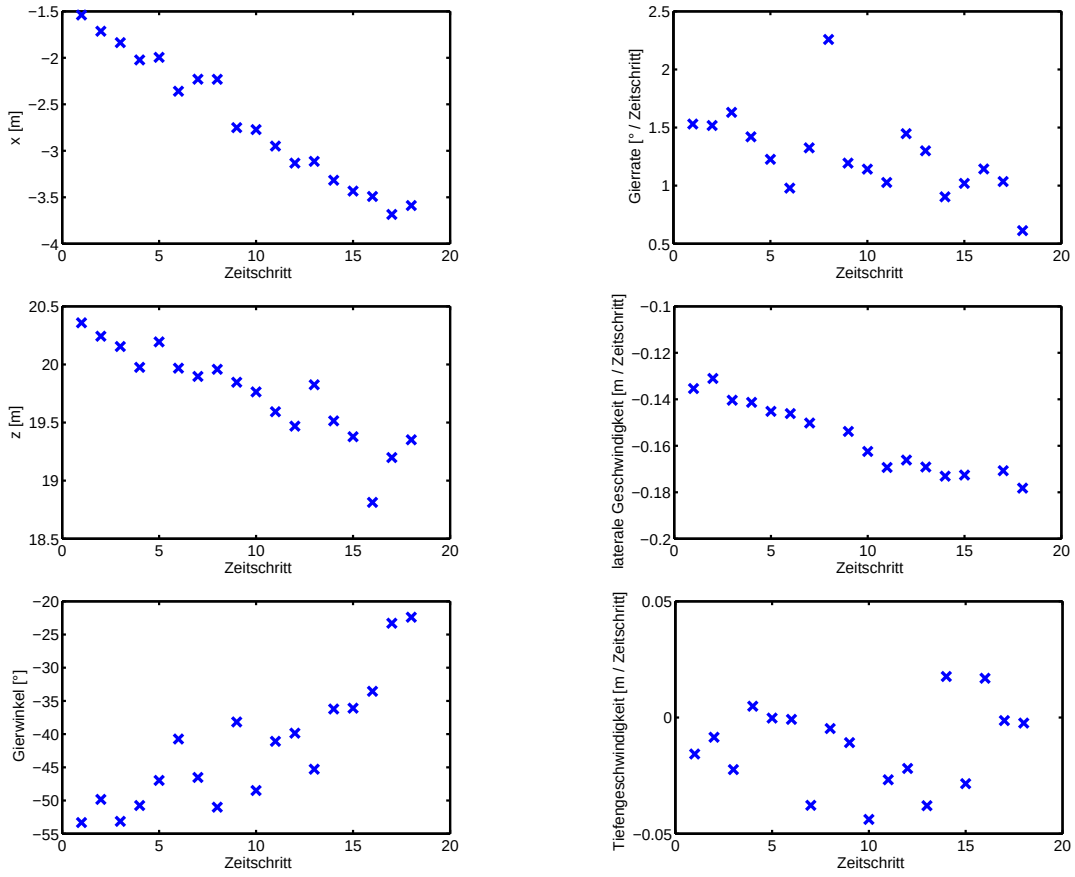


Abbildung 9.11.: Mithilfe des modellbasierten Szenenflusses bestimmte Poseparameter und deren zeitliche Ableitung für die Bildsequenz aus Abb. 9.10. Die Schätzung des Gierwinkels ist anspruchvoll, weshalb es Ausreißer bei der Schätzung gibt. Bei der Tiefengeschwindigkeit ist die Skalierung der Darstellung zu beachten.

10. Untersuchungen zur Integration von Modellwissen

10.1. Experimente zur Korrektur von Fehlkorrespondenzen

In diesem Abschnitt wird die vorgestellt Methode zur Korrektur von Fehlkorrespondenzen evaluiert. Zur Bildaufnahme wird ein PointGrey Digiclops Kamerasysten mit einer Bildauflösung von 1024×768 Pixel, einer Kamerakonstanten von 6 mm ($f = 1350$ pixels) und einer Basisbreite von $b = 100$ mm verwendet. Bei der Stereobildverarbeiutng kommt das Spacetime-Stereo zum Einsatz, wobei zur Korrespondenzanalyse drei unterschiedliche Constraints (Uniqueness-, Uniqueness- und Ordering-, oder Minimum-Weighted-Matching-Constraint, siehe Kap. 2.1.2) verwendet und evaluiert werden. Es werden vier verschiedene Testsequenzen benutzt, wobei in jeder ein verhältnismäßig kleines Objekt vor einer repetitiven Struktur sichtbar ist. Das rechte Kamerabild jeder Sequenz ist in Abb. 10.1a abgebildet. Szene 1 zeigt eine Person, die vor einem großen Zaun steht, Szene 2 eine Tastatur mit einer Hand davor, Szene 3 ein Arm mit einer repetitiven Struktur auf dem Ärmel und einen Stab davor und Szene 4 einen Gebäudeeingang mit einer Person davor. Szene 1, 2 und 4 benutzten das Ebenenmodell, wohingegen bei Szene 3 das Hand-Arm-Modell zum Einsatz kommt.

Eine manuell definierte Ground-Truth dient dazu, die resultierenden 3D-Punktewolken quantitativ zu evaluieren (siehe Abb. 10.1b). Für jede initiale 3D-Punktewolke ist das Adaptionsergebnis des jeweiligen Modells in Abb. 10.2 dargestellt. In allen vier Beispielen wird die Modelladaption korrekt durchgeführt und die aufgrund der sich wiederholenden Struktur entstehenden Phantomobjekte werden deutlich sichtbar.

Um die Ergebnisse der 3D-Szenenrekonstruktion quantitativ darzustellen, wird eine Art ROC-Kurve (engl.: „receiver operating characteristics“) benutzt, die das Verhältnis darstellen zwischen F_{rep} der 3D-Punkte, die korrekt der repetitiven Struktur zugeordnet wurden, und F_{obj} der 3D-Punkte, die korrekt dem Objekt davor zugeordnet wurden darstellt, wobei der Wert für den Parameter λ_e verändert wird (siehe Abb. 10.4). Jeder einzelne 3D-Punkt wird evaluiert, ob er zum Hintergrund (fällt aus der weiteren Betrachtung heraus), zur repetitiven Struktur oder zum Objekt gehört und einen korrekten Disparitätswert aufweist, wobei eine geringe Abweichung erlaubt wird, was auf Basis der manuell definierten Ground-Truth geschieht.

Zum Vergleich wurde das modellbasierte Stereoverfahren aus Kap. 4.4.8 auf Szene 1 angewendet (siehe Abb. 10.3). Dieses Verfahren zeigt eine sehr gute Rekonstruktion des Zauns, jedoch geht dabei natürlich die Information, dass sich vor dem Zaun noch eine Person befindet verloren, da diese nicht Teil der Modellannahme ist. Außerdem muss eine Poseinitialisierung bekannt sein, was durch das FFT-basierte Verfahren aus

Szene	Constraint	$\lambda_e = 0$	$\lambda_e = 0.01$
Zaun mit Person	Uniqueness	23317	24755
	Uniqueness + Ordering	15509	22752
	Min. Weighted Matching	25354	25463
Tastatur mit Hand	Uniqueness	9392	9924
	Uniqueness + Ordering	7613	9029
	Min. Weighted Matching	9980	10145
Arm mit Stab	Uniqueness	9164	9587
	Uniqueness + Ordering	8329	8601
	Min. Weighted Matching	9604	10064
Gebäude mit Person	Uniqueness	5023	5651
	Uniqueness + Ordering	3270	5136
	Min. Weighted Matching	5648	5654

Tabelle 10.1.: Anzahl der extrahierten 3D-Punkte für die untersuchten Beispielsequenzen ohne Modellwissen ($\lambda_e = 0$) und unter Einbeziehung von Modellwissen ($\lambda_e = 0.01$). Durch Einbeziehung von Modellwissen steigt die absolute Anzahl von 3D-Punkten. Insbesondere bei Anwendung des Ordering-Constraints wird die Anzahl der extrahierten 3D-Szenenpunkte erhöht, da die korregierten Punkte im Bereich der repetitiven Struktur die Korrespondenzanalyse insgesamt verbessern.

Kap. 6.1 entfällt. Es zeigt sich somit, dass die beiden komplementären Verfahren (lokales und modellbasiertes Stereo) nicht in der Lage sind, den Zaun und die Person davor zu erkennen.

Für alle vier Experimente gilt, dass eine Kombination von Uniqueness und Ordering Constraint bei der Analyse der Korrespondenzmatrix $\mathbf{E_t}$ (siehe Kap. 6.1) den höchsten Anteil von korrekten Punkten auf der repetitiven Struktur und korrekte Punkte auf dem Objekt liefert. Setzt man $\lambda_e = 0$, bedeutet das eine vollständige Unterdrückung des Modellwissens. Durch die Anhebung von λ_e auf einen Wert von 0.001 wird der Wert für F_{rep} ebenfalls erhöht, wobei der Anteil von F_{obj} weitestgehend konstant bleibt. Wenn λ_e weiter erhöht wird, erhöht sich F_{rep} nur noch langsam, während sich F_{obj} stark verringert. Als Konsequenz zeigt sich, dass intuitiv die besten Ergebnisse mit einem Wert für λ_e zwischen 0.001 und 0.01 für alle vier Experimente erzielt wird. Im Gegensatz zur klassischen Ausreißerbehandlung, wird durch die Verwendung der vorgestellten Methode nicht die absolute Anzahl von 3D-Punkten verringert, sondern es wird die Anzahl sogar bei allen Experimenten erhöht, wie Tabelle 10.1 belegt.

In Abb. 10.5 wird die resultierende 3D-Punktewolke für die vier Experimente im Disparitätsraum dargestellt, wobei $\lambda_e = 0$ gesetzt wurde bzw. $\lambda_e = 0.01$. In Szene 1 (Zaun mit Person, siehe Abb. 10.5a) bleiben lediglich einige wenige Punkte der „Phantomzäune“ übrig, wenn man λ_e von 0 auf 0.01 erhöht, wobei das Objekt davor richtig extrahiert wird. Ein ähnliches Verhalten zeigt sich in Szene 2 (Tastatur mit Hand, siehe Abb. 10.5b),

in der eine noch geringere Anzahl von falschen Punkten auf der repetitiven Struktur zurückbleibt. Diese bilden keine Cluster und sind eher zufällig im Disparitätsraum verteilt. In Szene 3 (Arm und Stab, siehe Abb. 10.5c) ist der Effekt nicht so ausgeprägt wie in Szene 1 oder 2. Jedoch zeigt sich auch hier deutlich, dass durch einen höheren Wert für λ_e die Punkte mit falscher Disparität auf der sich wiederholenden Struktur weniger werden, gerade wenn man sich den Bereich bei $u \approx 750$ Pixel und Disparitäten von 140 und 175 Pixel ansieht. Die Anzahl der falsch zugeordneten Objektpunkte erhöht sich zwar (diese Punkte haben eine Disparität erhalten, die nahe der des Arms ist), jedoch bleibt das Objekt durch zwei große Punktecluster in der Punktewolke erhalten. Für Szene 4 (Gebäude mit Person davor, siehe Abb. 10.5d) wirkt sich eine Erhöhung von λ_e auf Werte zwischen 0.001 und 0.0025 stark auf den Anteil von korrekten Punkten auf dem Gebäude aus, sodass Phantomobjekte verschwinden und die Anzahl von korrekten Punkten auf dem Objekt nur vernachlässigbar abnimmt.

In allen vier Szenen wurde die Anzahl der 3D Punkte, die Phantomobjekte bilden signifikant reduziert und vor allem Cluster mit falsch zugeordneten Punkten verschwinden. Daher eignet sich das vorgestellte Verfahren, um in den in dieser Arbeit berücksichtigten Szenarien zwischen einer repetitiven Struktur, einem Phantomobjekt und einem wahren Objekt zwischen Struktur und Kamera zu unterscheiden.

10.1.1. Fehlerbetrachtung

Es ist notwendig zu untersuchen, in welchem Umfang Ungenauigkeiten bei den bestimmten Werten a_s, b_s, and c_s der Modellebene im 3D-Raum (siehe Gl. 6.6) zu ungenauen Modelldisparitäten führen, da die Modellebene direkten Einfluss auf die Korrektur der Fehlkorrespondenzen und somit auf die spätere 3D-Punktewolke hat.

Die Korrespondenzanalyse basiert auf der Matrix $\mathbf{E_t}$ (siehe Gl. 6.8). Die Berechnung des Disparitätsunterschiedes zwischen 3D-Punkt und Modell und die experimentellen Untersuchungen benötigen außerdem eine Zuweisung der 3D-Punkte zum Modell oder zu einem Objekt vor der repetitiven Struktur basierend auf einer Disparitätsschwelle. Daher ist es notwendig zu untersuchen, in welchem Umfang Ungenauigkeiten bei den bestimmten Werten a_s, b_s, and c_s der Modellebene im 3D-Raum zu ungenauen Modelldisparitäten führen. Für die Fehler Δa_d, Δb_d und Δc_d der Modellebenenparameter im Disparitätsraum, führt das Fehlerfortpflanzungsgesetz zu folgenden Zusammenhängen:

$$\Delta a_d = \left|\frac{\partial a_d}{\partial c_s}\right| \Delta c_s + \left|\frac{\partial a_d}{\partial a_s}\right| \Delta a_s = \frac{l a_s}{c_s^2}\Delta c_s + \frac{l}{c_s}\Delta a_s \tag{10.1}$$

$$\Delta b_d = \left|\frac{\partial b_d}{\partial c_s}\right| \Delta c_s + \left|\frac{\partial b_d}{\partial b_s}\right| \Delta b_s = \frac{l b_s}{c_s^2}\Delta c_s + \frac{l}{c_s}\Delta b_s \tag{10.2}$$

$$\Delta c_d = \left|\frac{\partial c_d}{\partial c_s}\right| \Delta c_s = \frac{l f}{c_s^2}\Delta c_s \tag{10.3}$$

Für eine approximative, quantitative Fehleranalyse wird eine Kamerakonstante von $f = 1350$ Pixel und eine Basisbreite von $l = 0.1$ m angenommen (siehe Kap. 10.1). Die mittlere Distanz zur repetitiven Struktur entspricht in etwa dem Wert von c_s, was etwa

10 m für Außenszenen und 1 m für Innenszenen entspricht (siehe Kap. 10.1). Unter der sehr pessimistischen Annahme, dass der Wert von c_s mit einer Genauigkeit von 5% bestimmt werden kann, resultiert dies in einem Fehler für c_d unter Berücksichtigung von Gleichung (10.3) von 0.7 Pixel für Außenszenen und 6.8 Pixel für Innenszenen.

Die Ausdehnung des Objekts im Bild wird mit g bezeichnet und entspricht typischerweise 100–300 Pixeln, sodass $g = 200$ Pixel angenommen werden können. Der maximale Disparitätsfehler $\Delta d_{\max}$ infolge der Unsicherheit Δa_d des Modellparameters a_d im Disparitätsraum korrespondiert dann mit $\Delta d_{\max} = g \cdot \Delta a_d$. Zur Vereinfachung wird angenommen, dass für die tatsächliche Modellebene $a_s = b_s = 0$ ist, d. h. die Modellebene ist parallel der Bildebene. Wird zudem, wiederum vom schlechtesten Fall ausgehend, angenommen, dass $\Delta a_s = 0.6$ entspricht, was einem Winkelfehler von mehr als 30° für eine frontoparallele Ebene gleichkommt, ergeben sich $\Delta a_d = 0.006$ und $\Delta d_{\max} = 1.2$ Pixel für Außenszenen und $\Delta a_d = 0.060$ und $\Delta d_{\max} = 12$ Pixel für Innenszenen.

Basierend auf dieser Fehleranalyse wird die minimale Wellenlänge der repetitiven Struktur für Außen- und Innenszenen definiert, für die die vorgestellte Methode verwendet wird.

10.2. Experimente zur Klassifikation von artikulierten Objekten

Zum Training des in Kap. 6.2 beschriebenen Klassifikators wird ein Trainingsdatensatz verwendet, der normalisierte Ansichten der Hand-Unterarm-Region von verschiedenen Personen mit unterschiedlicher Bekleidung enthält, wie in Abb. 10.6 beispielhaft gezeigt. Neben diesen Beispielen für die erste Klasse (*HU*), werden auch Negativbeispiele verwendet, die keine Abbildung eines Hand-Unterarms enthalten. Diese werden der zweiten Klasse (*kein HU*) zugeordnet. Insgesamt besteht die verwendete Trainingsdatenmenge (Lernset) aus 1592 einzelnen Beispielen.

Zur Bewertung des Klassifikationsansatzes für artikulierte Objekte wird in erster Linie die sogenannte ROC-Kurve benutzt. Diese zeigt auf einen Blick, wie sich der gelernte Klassifikator auf einem ihm unbekannten und somit unabhängigen Testdatensatz (Testset) verhält. Das verwendete Testset besteht aus insgesamt 470 Beispielen beider Klassen. Zur Ermittlung der ROC-Kurve wird der Quotient aus falsch-positiv und richtig-positiven Klassifikationsergebnissen über eine sich ändernde Schwelle für die Klassenzugehörigkeitswahrscheinlichkeit aufgetragen. Als falsch-positiv bezeichnet man dabei einen Bildausschnitt, der als Hand-Unterarm erkannt wird, jedoch keinen Hand-Unterarm darstellt. Als richtig-positiv wird ein Beispiel bezeichnet, wenn es eine Hand-Unterarm-Region zeigt und auch der entsprechenden Klasse zugeordnet wird.

In dem verwendeten System haben die Bildausschnitte eine Größe von 60×15 Pixel, wodurch insgesamt 900 Grauwerte zur Verfügung stehen. Die Merkmalsreduktion fasst diese in einem 30-dimensionalen Merksmalsvektor zusammen, der anschließend dem Polynomklassifkator übergeben wird. Die Hauptkomponentenanalyse zur Merkmalsreduktion wird verwendet, da bei direkter Benutzung der 900 Grauwerte die Polynomstrukturliste viel zu groß wird, speziell wenn ein Klassifiaktor zweiten Grades zum Einsatz kommt. Dadurch wäre dann auch die Menge der benötigten Trainingsdaten enorm.

Durch die Reduktion der Dimensionalität des Merkmalsvektors auf 30 wird eine handhabbare Menge von Trainingsbeispielen benötigt und auch die Adation des Klassifikators an diese Trainingsmenge ist mit aktueller Rechnertechnology durchführbar. Durch die Hauptkomponenten wird dafür gesorgt, dass die signifikanten Daten im Lernset trotz Dimensionsreduktion an den Klassifikator weitergeleitet werden.

Die Kombination aus Polynomklassifkator und vorgeschalteter Hauptkomponentenanalyse wird maßgeblich über zwei Parameter gesteuert: zum einen über die Dimensionalität des Merkmalsvektors, der von der Hauptkomponentenanalyse an den Klassifikator übergeben wird, und zum anderen über den Grad des Polynoms auf dessen Basis der Polynomklassifikator arbeitet. Die Dimensionalität wird mit 30 Dimensionen definiert.

Theoretisch liefert ein Polynomklassifikator zweiten Grades immer bessere Klassifikationsergebnisse, als ein Klassifikator ersten Grades. Eine Evaluierung des Grades des Polynomklassifikators ist in Abb. 10.7 dargestellt. Hier wird für eine die Dimensionalität des Merksmalsvektors von 30 der Grad 1 und 2 gegenübergestellt. Der Klassifikator auf Basis Polynoms 2. Grades zeigt bessere Klassifikationsergebnisse und wird daher im System verwendet. Die Polynomstrukturliste besteht beim Klasifikator zweiten Grades und einer Dimensionalität von 30 aus 465 Elementen. Der Klassifikator zweiten Grades wird in einem vollquadratischen Ansatz verwendet, d.h. alle Elemente der Polynomstrukturliste werden beim Training adaptiert und zur Klassifikation herangezogen.

Abschließend wird für eine Testsequenz beispielhaft die Klassifikationsergebnisse gezeigt. Abb. 10.8 zeigt die Ergebnisse für eine der Sequenzen aus dem Datensatz der auch zur Evaluierung von Systemausprägung 2 (siehe Kap. 8.2) verwendet wurde. Das Ergebnis der Klassifikation zeigt, dass die Wahrscheinlichkeit, dass es sich bei dem jeweiligen Bildausschnitt um eine Hand-Unterarm-Abbildung handelt, für den ersten Teil der Sequenz meist bei ca. 70% liegt. Ab Zeitschritt 310 wurden die Poseergebnisse manipuliert, sodass die Pose bis zum Schluss der Sequenz gleich bleibt, obwohl sich der Arm weiter in der Szene bewegt. Dadurch wird ein typisches Fehlverhalten bei Trackingverfahren aus dem Stand der Forschung simuliert: der Verlust des Objekts z.B. bei zu schnellen Bewegungen. Neben dem eigentlichen Klassifikationsergebnis ist auch noch dessen zeitliche Filterung zu sehen. Durch diese Filterung wird erreicht, dass kurzzeitige Fehlklassifikationen nicht direkt zum Stillstand des Systems führen. Es wird in diesem Falle der Mittelwert aus den jeweils letzten 5 Klassifikationsergebnissen gebildet. Dadurch ergibt sich jedoch auch eine leicht verzögerte Erkennung von Fehlern der Pose-Estimation.

Wertung

Die Evaluierung zeigt, dass mit dem vorgestellten Verfahren eine Klassifikation von artikulierten Objekten möglich ist. Um die Klassifikationsleistung noch weiter zu steigern ist die Verwendung einer größeren Lerndatenbank denkbar, sowie die Verwendung eines anderen Klassifikationsansatzes. Es ist außerdem zu erwähnen, dass die aktuelle Lerndatenbank aus Bildern verschiedenster Personen und Bekleidungen bestehen. Auch hier würden sich weitere Verbesserungen erreichen lassen, indem verschiedene Klassifikatoren für verschiedene Arten von Bekleidung verwendet werden, da hier ein signifikanter Unterschied in der Abbildung des Arms entsteht.

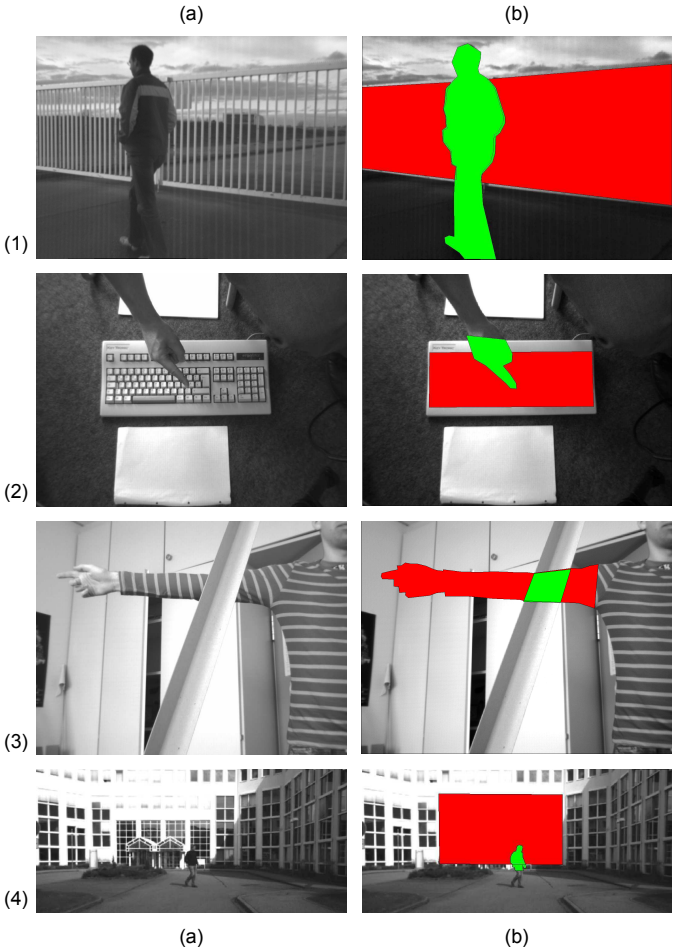


Abbildung 10.1.: Testsequenzen 1, 2, 3 und 4 (von oben nach unten). Links: Rechtes Kamerabild jeder Sequenz. Rechts: Manuell gelabelte Szenenobjekte. Bildregionen mit repetitiver Struktur sind in rot dargestellt. Das Objekt vor der repetitiven Struktur ist in grün dargestellt.

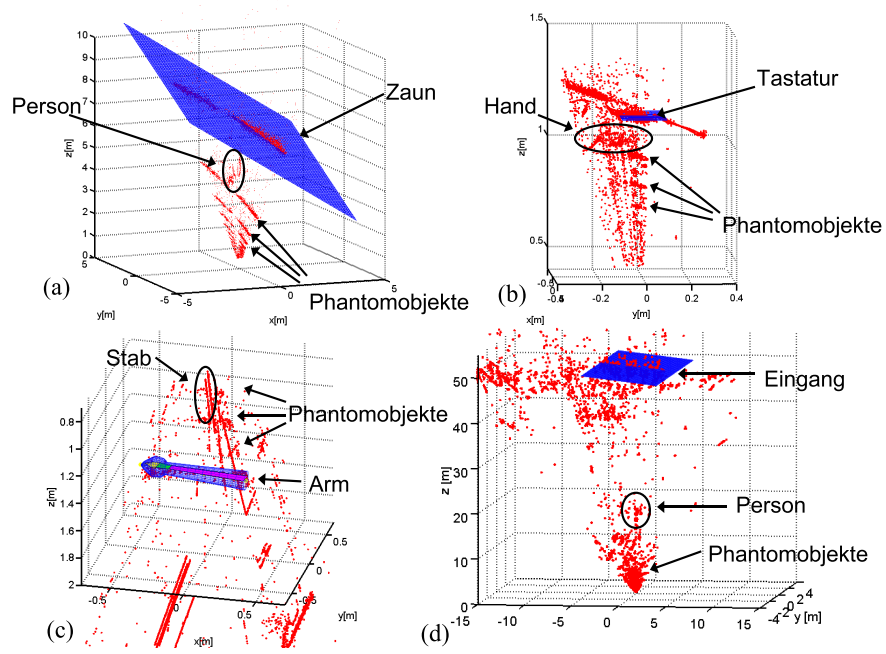


Abbildung 10.2.: Initiale 3D-Punktewolke (korrespondierend mit $\lambda_e = 0$, siehe Gl. 6.8) für alle vier Szenen (hier a – d) im 3D-Raum zusammen mit den adaptierten Modellen, wobei die Kombination aus Uniqueness und Ordering Constraint zur Generierung der Punktewolke benutzt wurde.

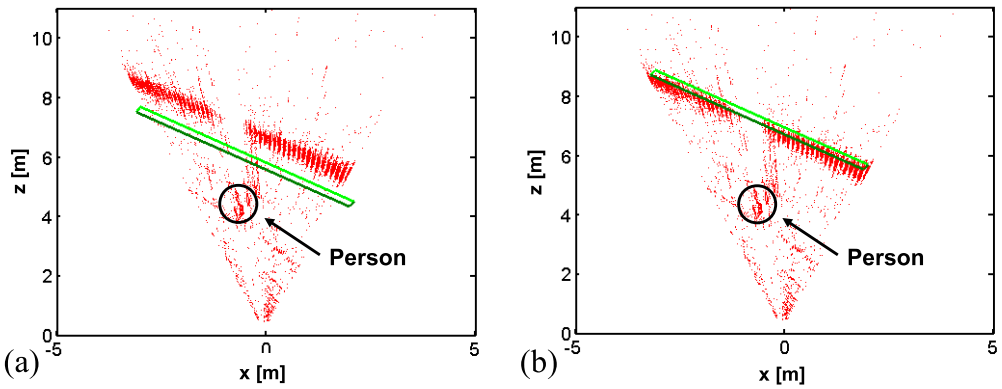


Abbildung 10.3.: Um das Verfahren mit einem anderen modellbasierten Verfahren zu vergleichen wurde Szene 1 mit dem modellbasierten Stereoalgorithmus (siehe Kap. 4.4.8) ausgewertet. Links: Initialisierung des modellbasierten Stereoalgorithmus. Rechts: Endergebnis des modellbasierten Stereoalgorithmus. Die Punktewolke wurde mithilfe des Spacetime-Stereos berechnet. Die Person vor dem Zaun, welche durch einen Kreis gekennzeichnet ist, verschwindet durch die Modellannahme, da lediglich der Zaun modelliert wurde.

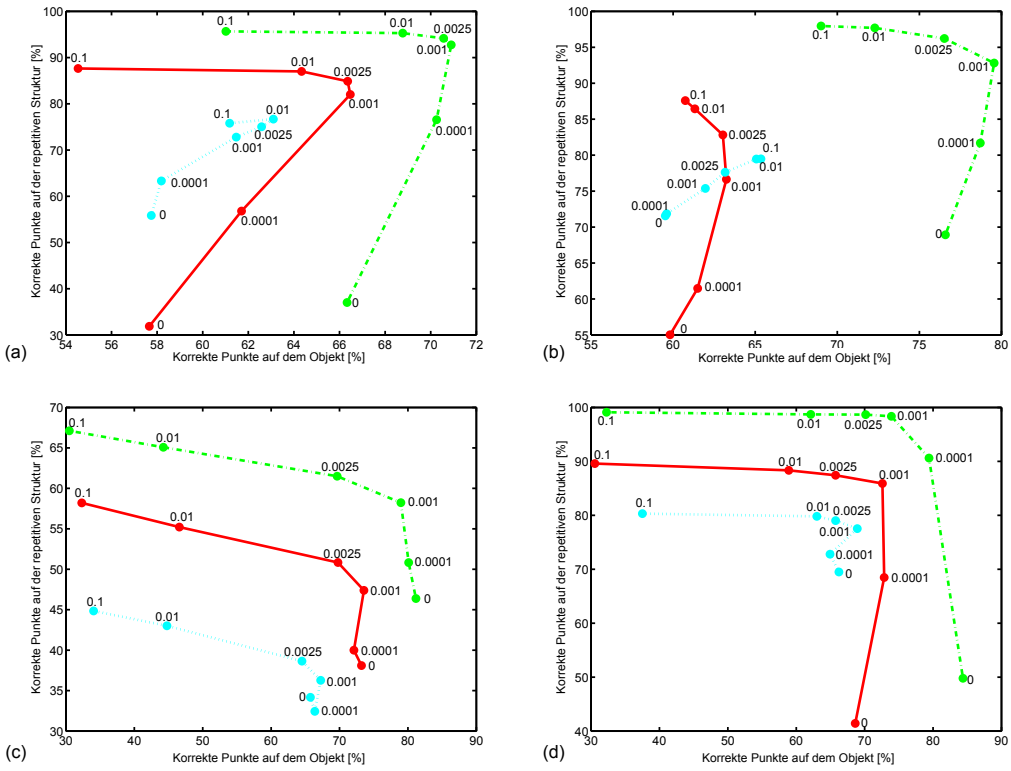


Abbildung 10.4.: 3D-Rekonstruktionsergebniss für die vier untersuchten Szenen. Die rote, durchgezogene Kurve wurde durch die Verwendung des Uniqueness-Constraints, die grüne, gestrichelte Kurve durch die Kombination aus Uniqueness- und Ordering Constraint und die hellblaue durch das Minimum-Weighted-Matching-Constraint generiert. Die Zahlen an den Kurven beschreiben den entsprechenden Wert des Gewichtungsfaktors λ_e. (a) Szene 1 (Zaun mit Person). (b) Szene 2 (Tastatur mit Hand). (c) Szene 3 (Arm mit Stab). (d) Szene 4 (Gebäudefront). Die Kurven zeigen den Anteil korrekter Punkte auf der repetitiven Struktur und den Anteil korrekter Punkte auf dem Objekt vor der repetitiven Struktur. Es zeigt sich, dass durch einen gößeren Einfluß des Modellwissens in der Korrespondenzanalyse es mehr korrekte Punkte auf der repetitiven Struktur gibt, wobei der Anteil der korrekten Punkte etwa gleich bleiben. Wird die Korrespondenzanalyse zu sehr vom Modellwissen beeinflußt ($\lambda_e \geq 0.0025$), entstehen Punkte mit falscher Disparität im Bereich des Objekts vor der repetitiven Struktur.

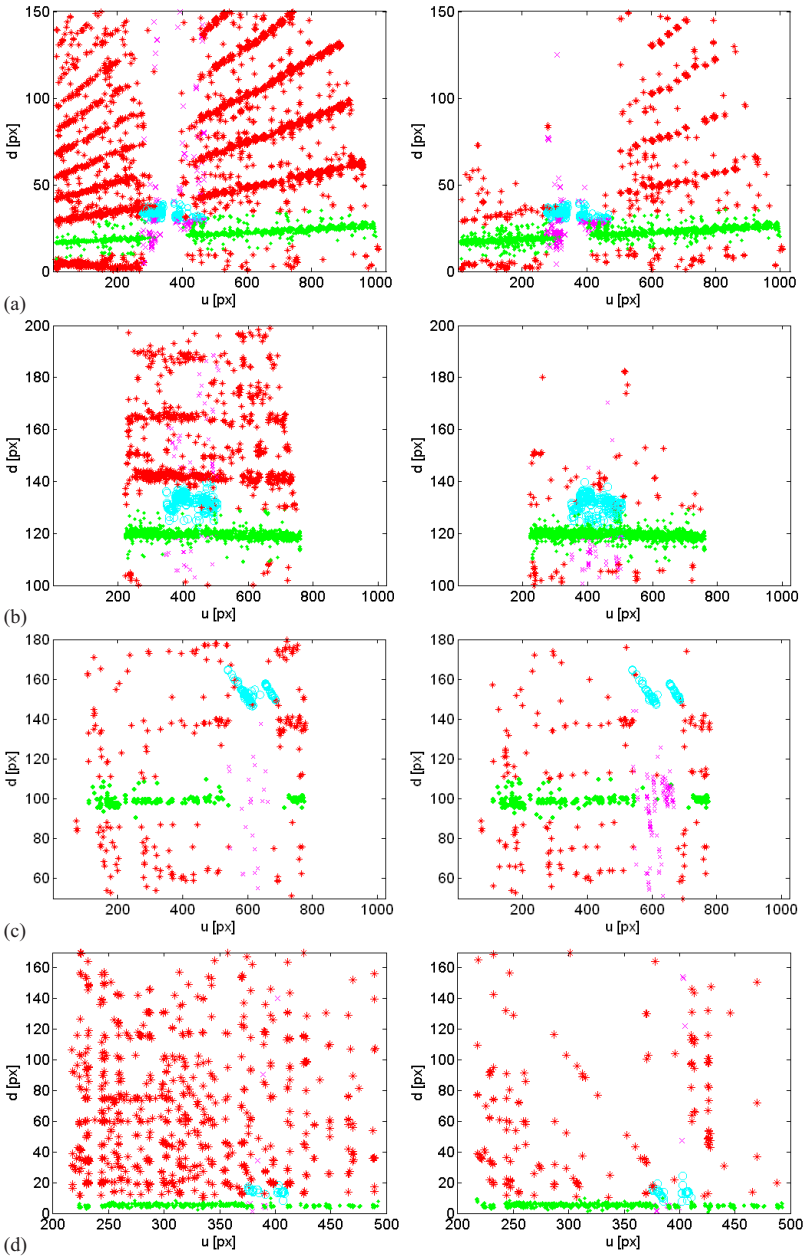


Abbildung 10.5.: 3D-Punktewolken mit $\lambda_e = 0$ (linke Spalte), d. h. ohne Anwendung von Modellinformationen (siehe Gl. 6.8) und $\lambda_e = 0.01$ (rechte Spalte). (a) Szene 1 (Zaun mit Person). (b) Szene 2 (Tastatur mit Hand). (c) Szene 3 (Arm mit Stab). (d) Szene 4 (Gebäude mit Person). Grüne Punkte kennzeichnen korrekte, rote Sterne inkorrekte Punkte im Bereich der repetitiven Struktur, wohingegen hellblaue Kreise korrekte und violette Kreuze inkorrekte Punkte auf dem Objekt darstellen.

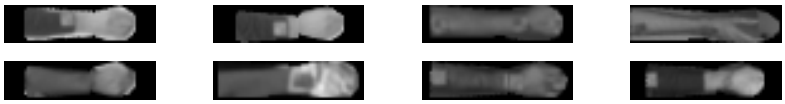


Abbildung 10.6.: Beispiele aus dem Datensatz zum Trainieren des Klassifikators. In dem Datensatz sind Bildausschnitte der Hand-Unterarm-Region von Personen mit unterschiedlicher Kleidung vorhanden: ohne Bekleidung, mit Pullover bzw. Hemd oder mit Handschuh.

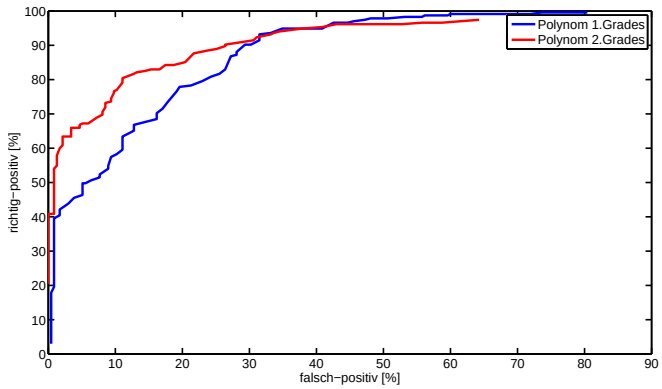


Abbildung 10.7.: Evaluierung des Polynomgrades des Klassifkators auf dem Testset. Der Klassifikator 2.Grades zeigt bessere Ergebnisse, jedoch ist dadurch die Größe der Polynomstrukturliste wesentlich gestiegen.

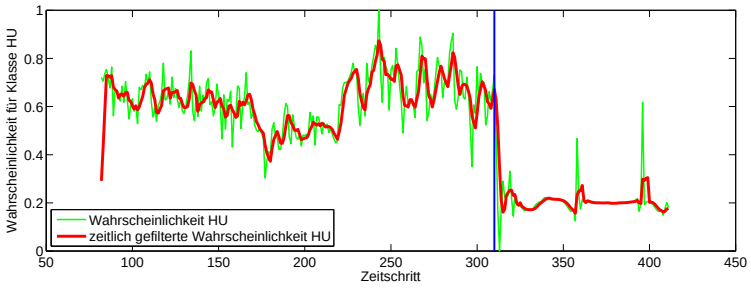


Abbildung 10.8.: Zeitliches Verhalten der Ausgabe des vollquadratischen Polynomklassifikators für Klasse *HU* (Hand-Unterarm) mit und ohne zeitliche Filterung. Die senkrechte Linie bezeichnet den Abbruch des Tracks.

Teil IV.

Zusammenfassung und Ausblick

11. Zusammenfassung und Schlußfolgerungen

Im Rahmen dieser Arbeit wurde ein System zur Analyse der Position, Orientierung und Bewegung von rigiden und artikulierten Objekten aus Stereobildsequenzen entwickelt. Das System wurde in zwei unterschiedlichen Szenarien evaluiert und durch verschiedene Systemausprägungen angepasst: das Produktionsszenario in der Fabrik und das Straßenverkehrsszenario im Kreuzungsbereich. Außerdem wurden zwei Ansätze vorgestellt, die beschreiben, wie die Integration von Modellwissen als Rückkopplungen in den Ablauf des Gesamtsystems zur Steigerung der Robustheit beitragen können.

Auf Basis der Stereobildsequenzen wurden zunächst raum-zeitliche Szenenrekonstruktionsdaten ermittelt. Dazu wurden zum einen Verfahren aus dem Stand der Forschung zur Berechnung von 3D-Punkten und des optischen Flusses kombiniert und somit ein unvollständiger Szenenfluss dargestellt. Außerdem wurde das Spacetime-Stereo-Verfahren im Rahmen dieser Arbeit aufgegriffen und an verschiedenen Stellen weiterentwickelt, was zur Erhöhung der Rekonstruktionsgenauigkeit und zur Steigerung der Robustheit des Verfahrens beitrug. Die Verwendung des jeweiligen Verfahrens zur Berechnung des Szenenflusses ist in erster Linie durch das Szenario bestimmt.

Auf Basis dieser Szenenflussinformationen wurde dann eine allgemeingültige Methode zur Vorsegmentierung vorgestellt. Hauptbestandteil ist hier das von Konrad (2006) entwickelte Clusterverfahren. Dadurch erhält man eine objektspezifische 3D-Punktewolke, die das beobachtete Objekt repräsentiert und gleichzeitig durch die Verwendung von Szenenflussinformationen auch Bewegungsinformationen der einzelnen 3D-Punkte enthält, was später für die Bewegungsanalyse wichtig ist.

In der Literatur findet sich der sogenannte Iterative Closest Point-Algorithmus meist als das einschlägige Verfahren zur Bestimmung der Transformation zwischen zwei 3D-Punktewolken. In dieser Arbeit wird dieser Algorithmus dazu benutzt, ein objektspezifisches 3D-Modell an die aus der Szene extrahierte Punktewolke anzupassen. Dabei wird jedoch nicht der ursprüngliche von Besl und McKay (1992) erwähnte ICP-Algorithmus verwendet, sondern verschiedene neuentwickelte Modifikationen davon, die je nach Anwendung eine bessere Robustheit, eine höhere Genauigkeit oder auch eine schneller Verarbeitung ermöglichen. Durch eine ebenfalls neuentwickelte Fusion mit einem konturbasierten Ansatz kann eine weitere Steigerung der Systemleistung in Bezug auf die Robustheit erreicht werden. Ein grundsätzlich anderer Ansatz ist das modellbasierte Stereo, welches direkt auf den Eingangsbildern der Kamera arbeitet und durch die Ausnutzung aller vorhandenen Informationen eine sehr hohe Genauigkeit bei der Poseschätzung von Fahrzeugen zeigt.

Die abschließende Bestimmung der Objektbewegung baut wiederum auf verschiedenen

Ansätzen auf. Dabei stellt die neuartige Erweiterung des von Schunck (1989) bekannten Constraint-Line-Clusterings eine schnelle Art der Schätzung der vollständigen Bewegungsparameter eines Objekts durch die Analyse der optischen Fluss-Daten, welche mit den 3D-Punkten der Objektpunktewolke verknüpft sind, dar. Diese Daten wurden außerdem zur Initialisierung des von Hahn et al. (2008b) vorgestellten ShapeFlow-Algorithmus benutzt. Des Weiteren wurde auch die Idee des modellbasierten Stereos raum-zeitlich erweitert, wodurch der neuartige Ansatz des modellbasierten Szenenflusses entstand. Hier werden wiederum alle verfügbaren Informationen (in diesem Fall aus zwei Zeitschritten jeweils zwei Stereobilder, also insgesamt vier Bilder) genutzt, um möglichst genau die Bewegung des Objekts bestimmen zu könnnen.

Im Rahmen dieser Arbeit wurden außerdem zwei neuartige Methoden zur Rückkopplung von Berechnungsergebnissen in die Verarbeitungshierarchie entwickelt. Zum einen ein Verfahren, mit dessen Hilfe Fehlkorrespondenzen in der Stereobildverarbeitung vermieden werden können. Zum anderen ein Verfahren mit dem es möglich ist, artikulierte Objekte zu klassifizieren. Beiden Methoden bringen einen erheblichen Vorteil, basieren auf dem vorher vorgestellten Systemaufbau und sind somit ohne großen Aufwand mit in das Gesamtsystem integrierbar.

Die Evaluierung in dieser Arbeit basiert ausschließlich auf realen Szenen, welche mit verschiedenen Algorithmen berechnet wurden, um die einzelnen Methoden gegeneinander testen zu können. Dabei wurde auch stets eine von einem unabhängigen System ermittelte Ground-Truth verwendet, um objektiv die Qualität der Ergebnisse bewerten zu können. Auch wurden Verfahren aus dem Stand der Forschung herangezogen, um sicherzustellen, dass durch die neuentwickelten Verfahren ein Fortschritt erreicht wurde.

Die Evaluierungen, welche im Rahmen dieser Arbeit durchgeführt wurden, führen zu mehreren Schlussfolgerungen, die im Folgenden nochmals kurz zusammengefasst sind:

1. Die Wahl eines geeigenten Stereoalgorithmus zur 3D-Szenenrekonstruktion ist stets von der Anwendung, dem Umfeld, der Objektgröße, der verfügbaren Rechenzeit und der notwendigen Genauigkeit abhängig.

2. Eine raum-zeitliche Szenenrekonstruktion vereinfacht die Segementierung der Szene in einzelne Objekte gegenüber einer rein räumlichen Rekonstruktion.

3. Eine Initialisierung des ICP-Algorithmus durch eine Hauptkomponentenanalyse der 3D-Punktewolke ist meist sinnvoll und vereinfacht die nichtlineare Poseschätzung erheblich.

4. Der ICP-Algorithmus kann als Basis zur modellbasierten Poseschätzung verwendet werden. Um eine hohe Genauigkeit in Verbindung mit einer guten Robustheit zu erhalten, sind jedoch verschiedene Anpassungen notwendig.

5. Eine problemspezifische Fehlermetrik im ICP-Algorithmus führt speziell im Fernbereich zu einer Erhöhung der Genauigkeit der Poseschätzung.

6. Die ICP-Variante auf Basis der geschlossenen Lösung mittels Quaternionen führt zu einer ca. 40 mal schnelleren Poseschätzung als bei Verwendung einer iterativen, nichtlinearen Optimierung, wobei die Genauigkeit jedoch leicht abnimmt.

7. Das modellbasierte Stereo führt zu sehr hohen Genauigkeiten bei der Poseschätzung.

8. Auf Basis von Spacetime-Stereo-Daten ist eine vollständige Schätzung der Objektbewegung für rotationssymmetrische Objekte mittels erweitertem Constraint-Line-Clustering-Ansatz möglich.

9. Eine Fusion von punkte- und konturbasierten Pose-Estimation-Methoden verbessert sowohl Genauigkeit als auch Robustheit des Systems, benötigt jedoch Wissen über die abgebildete Objektkontur.

10. Das neuentwickelte, modellbasierte Szenenfluss-Verfahren erreicht durch die Verwendung aller zur Verfügung stehender Daten eine gute Genauigkeit bei der Bewegungsschätzung, auch in Bezug auf die sehr anspruchvoll zu schätzende Tiefengeschwindigkeit.

11. Rückkopplungen in der Verarbeitungshierarchie sind sinnvoll und erhöhen bei einem geringen Mehraufwand die Robustheit des Gesamtsystems.

12. Fehlkorrespondenzen in der Sterobildverarbeitung können durch das Einbringen von Modellwissen stark minimiert werden.

13. Die Klassifikation artikulierter Objekte wird unter Verwendung der Objektpose ermöglicht.

12. Ausblick

Die Daten zur Pose und zur Bewegung des beobachteten Objekts sind Voraussetzung für verschiedene Anwendungen, u.a. im Produktionsumfeld und im Straßenverkehrsumfeld. Dabei sind die beiden Anwendungsszenarien zu trennen. In der Produktion ist sicherlich im ersten Schritt eine Verbesserung der Erkennung von Kollisionsgefahren umsetzbar. Außerdem können durch die Bewegungserkennung Schutzräume enger abgesteckt und dynamisch ausgelegt werden, was zu einer Erhöhung der Flexibilität und zu Vorteilen bei der Entwicklung von Produktionsstandorten führt. Des Weiteren ist so eine Interaktion zwischen der Maschine und dem Menschen möglich. Durch die sichere Erkennung der Körperteile des Menschen, in Verbindung mit den aus der Robotersteuerung bekannten Bewegungsmuster der Maschine, können Kollisionen verhindert werden, die Bahn des Roboters umgeplant werden oder notfalls die Fahrt gestoppt werden, was Vorraussetzung für eine sichere Interaktion ist. Dadurch wird es erstmals möglich, die Stärken von Mensch und Maschine in einem Produktionsschritt zu vereinen. Außerdem ist durch das detaillierte Wissen über die Trajektorie der menschlichen Hand-Unterarm-Region eine Handlungserkennung, wie beispielsweise von Hahn et al. (2008a) beschrieben, umsetzbar. Durch diese Analyse ist zum einen eine Vollständigkeitsprüfung der notwendigen Arbeitsschritte möglich. Zum anderen kann eine Langzeitprädiktion erstellt werden, um noch früher mögliche Kollisionen erkennen zu können. Zusammengefasst ist die Analyse der Pose und der Bewegung menschlicher Körperteile der Grundstein für zukünftige Produktionsstätten, bei denen Sicherheit, Flexibilität und eine Kombination von automatisierten und manuellen Handlungsabläufen im Vordergrund stehen.

Im Fahrzeugumfeld und im speziellen im Bereich der Fahrerassistenzsysteme werden in den kommenden Jahren viele Innovationen in den Markt eingeführt. Dabei steht das Thema Sicherheit nach wie vor an oberster Stelle, wobei auch hier die Information über die Pose und die Bewegung der anderen Verkehrsteilnehmer Grundlage für die verschiedenen Systeme ist. Darauf aufbauend ist eine Situationsanalyse möglich, die die Gefahr von Zusammenstößen nicht nur im Längsverkehr, sondern auch im querenden Verkehr erkennt. Dadurch wird eine Fahrerassistenz im Kreuzungsbereich möglich, was aufgrund der hohen Fahrzeuggeschwindigkeiten in Verbindung mit sich teilweise abrupt ändernden Bewegungsrichtungen anspruchsvoll ist. Daher ist es auch wichtig, nicht nur die Position und die Geschwindigkeit der anderen Verkehrsteilnehmer zu kennen, sondern auch deren Gierwinkel und Gierraten. Hierdurch lassen sich Abbiegemanöver frühzeitig erkennen und mit in der Situationsanalyse verarbeiten. Es lässt sich außerdem eine Langzeitprädiktion der anderen Fahrzeuge mithilfe einer Trajektorienklassifikation durchführen, wie es beispielsweise von Hermes et al. (2009) bereits beschrieben wurde. Durch gezieltes Warnen des Fahrers bzw. im Notfall durch direkten, autonomen Eingriff in das eigene Fahrverhalten kann der Fahrer gerade in komplizierten Kreuzungsumfeldern unterstützt

werden, wodurch sich Unfälle vermeiden lassen. Zusammengefasst ist die Information über die Pose und die Bewegung der anderen Verkehrsteilnehmer Grundlage für die Assistenz des Fahrers in komplexen Verkehrssituationen.

Symbolverzeichnis

α_1, α_2	Posewinkel des Hand-Unterarm-Modells für den Unterarm(1) bzw. die Hand(2) entsprechend der Drehung um die y-Achse
β_1, β_2	Posewinkel des Hand-Unterarm-Modells für den Unterarm(1) bzw. die Hand(2) entsprechend der Drehung um die z-Achse
$\mathcal{F}$	Objektivebene der Kamera
$\mathcal{O}$	optisches Zentrum der Kamera
$\mathcal{R}$	Bildebene der Kamera
$\Delta\dot{z}$	Geschwindigkeitsdisperion entlang der z-Achse
Δg	laterale Geschwindigkeitsdisperion über das Objekt
Δt	Dauer eines Zeitschritts
δ	Vertikaler Verlauf der modellierten Grauwert im Spacetime-Stereo
$\dot{u}, \dot{v}$	optischer Fluss-Komponenten
$\dot{x}, \dot{y}, \dot{z}$	Geschwindigkeitskomponenten eines 3D-Punkts in metrischen Einheiten
λ	Regularisierungsparameter
λ_e	Gewichtungsparameter bei der Korrektur von Fehlkorrespondenzen
$\delta_{\mathbf{dist}} = [\delta_e, \delta_v]^T$	Distanzvektor im graphenbasierten Clusterverfahren
$\dot{\mathbf{\Phi}}$	zeitliche Ableitung der Pose, Objektbewegung
$\dot{\mathbf{q}}$	Quaternion
$\mathbf{\Phi}$	Objektposevektor
$\mathbf{\Phi}_{\text{fusion}}$	Ergebnis der Pose-Estimation-Fusion
$\mathbf{C}$	Korrespondenzmatrix im Spacetime-Stereo
$\mathbf{D}$	Matrix der Disparitätswerte für die Korrespondenzanalyse im Spacetime-Stereo
$\mathbf{E_{SSD}}$	Matrix der SSD-Vergleichswerte im Spacetime-Stereo
$\mathbf{m}$	Bildpunkt
$\mathbf{m}_0$	Hauptpunkt des Kamerasystems
$\mathbf{P_0} = [P_{0_x}, P_{0_y}, P_{0_z}]^T$	Aufpunkt des Hand-Unterarm-Modells
$\mathbf{P}$	Projektionsmatrix
$\mathbf{p}$	Parametervektor der Gauwertmodellierung im Spacetime-Stereo
$\mathbf{R_y}, \mathbf{R_z}$	Rotationsfunktion um die y- bzw. z-Achse
$\mathbf{R}$	Rotationsmatrix

$\mathbf{S_Y}$	Sobel-Operator für vertikale Kanten
$\mathbf{t}$	Translationsvektor
$\mathbf{w}$	3D-Punkt
μ	Zeitlicher Verlauf der modellierten Grauwert im Spacetime-Stereo
ω_o	Winkelgeschwindigkeit der Objektrotation orthogonal der Bildebene
ω_p	Winkelgeschwindigkeit der Objektrotation parallel der Bildebene
ϕ_n	Normalfluss
$\sigma_p, \sigma_o, \sigma_c$	Wert für das Punktabstands-, Orientierungsähnlichkeits- bzw. Ansichtsähnlichkeitskriterium in der Pose-Estimation-Fusion
θ_e, θ_v	Schwellenparameter für das graphenbasierte Clusterverfahren
$\tilde{\mathrm{o}}_r^{(i)}$	Projektiver Vektor der Bildausschnitte in der modellbasierten Stereobildverarbeitung
${}^{l}_{r}H^{(i)}$	Homographie der posevermittelten Verknüpfung zwischen linkem und rechtem Stereobild in der modellbasierten Stereobildverarbeitung
a_d, b_d, c_d	Parameter der Modellebene im Disparitätsraum bei der Korrektur von Fehlkorrespondenzen
a_s, b_s, c_s	Parameter der Modellebene im 3D-Raum bei der Korrektur von Fehlkorrespondenzen
b	Basisbreite des Stereokamerasystems
C	Kamerakoordinatensystem
d	Disparität
$D_{\max}$	Abstandsschwelle für die Modellanpassung mittels ICP-Algorithmus
E_φ	Winkelfehlerterm bei der problemspezifische ICP-Variante
E_r	Distanzfehlerterm bei der problemspezifische ICP-Variante
e_{ICP}	Fehlerfunktion des ICP-Algorithmus
E_{MBS}	Fehlerfunktion in der modellbasierten Stereobildverarbeitung
E_{MSF}	Fehlerfunktion beim modellbasierten Szenenfluss
$e_{\tanh}$	Fehlerfunktion für die Grauwertmodellierung im Spacetime-Stereo
f	Kamerakonstante
$I_s(\mathbf{p}, u)$	synthetische Grauwertverteilung in Abhängigkeit des Parametervektors $\mathbf{p}$ und der Position u innerhalb einer Zeile
J	Bildkoordinatensystem
k_1, k_2, k_3, k_4, k_5	Linsenverzeichnungsparameter
k_u, k_v	Skalierungsfaktoren/Anzahl der Pixel pro Meter auf Bildsen-

	sor
l_h, l_a	Länge der Hand bzw. des Unterarms
M_c	Punktewolkencluster, objektunspezifisch
m_H, m_W, m_L	Höhe(H), Breite(W) und Länge(L) des Fahrzeugmodells
M_m	Szenenfluss-Punktewolke, die alle sich bewegenden Punkte beinhaltet
M_p	Punktewolke der Szene
M_p	Szenenfluss-Punktewolke der beobachteten Szene
M_{Obj}	Objektspezifisches Punktecluster
P	Pixelkoordinatensystem
$r_1, \dots r_7$	Radien des Hand-Unterarm-Modell
$s_r^{(i)}, s_l^{(i)}$	Bildausschnitte im rechten bzw. linken Bild in der modellbasierten Stereobildverarbeitung
t_x, t_z	Translationskomponente entlang der x- bzw. z-Achse
u, v	Bildkoordinaten
u_e	betragsgrößter Gradient in der Modellfunktion des Spacetime-Stereos
w_{ICP}	Gewicht für das ICP-Ergebnis in der Pose-Estimation-Fusion
w_{MOCCD}	Gewicht für das MOCCD-Ergebnis in der Pose-Estimation-Fusion
W	Weltkoordinatensystem
$w(\mathbf{p}, u, v, t)$	Gewichtsfunktion für die Grauwertmodellierung mit dynamischer Maskierung im Spacetime-Stereo
w_f	Gewichtsfunktion der Ausreißerbehandlung im ICP-Algorithmus
x, y, z	3D-Koordinaten
$\Delta u_{\text{ROI}} = [\Delta u_1, \Delta u_2]^T$	Versatzinformationen für neue ROI-Definition im Spacetime-Stereo
I(u,v)	Bildmatrix der Grauwerte mit den Pixelkoordinaten u und v

Literaturverzeichnis

Aggarwal, J. K. und Cai, Q. (1999). Human motion analysis: A review. *Computer Vision and Image Understanding: CVIU*, 73(3):428–440.

Amberg, B., Blake, A., Fitzgibbon, A., Romdhani, S., und Vetter, T. (2007). Reconstructing high quality face-surfaces using model based stereo. In *Int. Conf. on Computer Vision*, pages 1–8.

Anandan, P. (1989). A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 2(3):283–310.

Arun, K. S., Huang, T. S., und Blostein, S. D. (1987). Least-squares fitting of two 3-d point sets. *IEEE Trans. Pattern Anal. Mach. Intell.*, 9(5):698–700.

Baker, H. und Binford, T. (1981). Depth from edge and intensity based stereo. In *International Joint Conference on Artificial Intelligence*, pages 631–636.

Baker, S., Roth, S., Scharstein, D., Black, M. J., Lewis, J., und Szeliski, R. (2007). A database and evaluation methodology for optical flow. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1–8.

Barnard, S. T. und Fischler, M. A. (1982). Computational stereo. *ACM Comput. Surv.*, 14(4):553–572.

Barron, J. L., Fleet, D. J., und Beauchemin, S. S. (1994). Performance of optical flow techniques. *Int. J. Computer Vision*, 12(1):43–77.

Barth, A. und Franke, U. (2008). Where will the oncoming vehicle be the next second? In *IEEE Intelligent Vehicles Symposium*.

Besl, P. J. und McKay, N. D. (1992). A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256.

Bhatia, S., Sigal, L., Isard, M., und Black, M. J. (2004). 3d human limb detection using space carving and multi-view eigen models. In *In: IEEE workshop on articulated and nonrigid motion, CVPR*.

Biber, P., Andreasson, H., Duckett, T., und Schilling, A. (2004). 3d modeling of indoor environments by a mobile robot with a laser scanner and panoramic camera. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2004)*.

Birchfield, S. und Tomasi, C. (1998). A pixel dissimilarity measure that is insensitive to image sampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20:401–406.

Blais, G. und Levine, M. D. (1995). Registering multiview range data to create 3d computer objects. *IEEE Trans. Pattern Anal. Mach. Intell.*, 17(8):820–824.

Bock, H. H. (1974). *Automatische Klassifikation*. Vandenhoeck & Ruprecht, Göttingen, Germany.

Brenner, C. (2005). Building reconstruction from images and laser scanning. *International Journal of Applied Earth Observation and Geoinformation*, 6(3-4):187 – 198. Data Quality in Earth Observation Techniques.

Brodsky, T., Fermüller, C., und Aloimonos, Y. (1998). Directions of motion fields are hardly ever ambiguous. *Int. J. Comput. Vision*, 26(1):5–24.

Bronshtein, I. N. und Semendyayev, K. A. (1997). *Handbook of mathematics (3rd ed.)*. Springer-Verlag, London, UK.

Brown, M. Z., Burschka, D., und Hager, G. D. (2003). Advances in computational stereo. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(8):993–1008.

Brox, T., Bruhn, A., Papenberg, N., und Weickert, J. (2004). High accuracy optical flow estimation based on a theory for warping. In *In Proc. 8th European Conference on Computer Vision*, pages 25–36. Springer.

Bruhn, A. (2006). *Variational Optic Flow Computation: Accurate Modelling and Efficient Numerics*. PhD thesis, Saarland University.

Bruhn, A., Weickert, J., und Schnörr, C. (2005). Lucas/kanade meets horn/schunck: Combining local and global optic flow methods. *Intl. J. of Computer Vision*, 61(3):211–231.

Bullock, D. und Zelek, J. (2005). Towards real-time 3-d monocular visual tracking of human limbs in unconstrained environments. *Real-Time Imaging*, 11(4):323–353.

Canny, J. (1986). A computational approach to edge detection. *IEEE Trans. on Pattern Analysis and. Machine Intelligence*, 8(6):679–698.

Cheung, G. K., Kanade, T., Bouguet, J.-Y., und Holler, M. (2000). A real time system for robust 3d voxel reconstruction of human motions. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 2:2714.

Collado, J. M., Hilario, C., de la Escalera, A., und Armingol, J. M. (2004). Model based vehicle detection for intelligent vehicles. In *IEEE Intelligent Vehicles Symposium*, pages 572–577.

Comaniciu, D. und Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(5):603–619.

Craig, J. J. (1989). *Introduction to Robotics*. Addison-Wesley Publishing Company, 2. edition.

Dang, T., Hoffmann, C., und Stiller, C. (2002). Fusing optical flow and stereo disparity for object tracking. In *Proceedings of the 5th IEEE International Conference on Intelligent Transportation Systems*.

Delamarre, Q. und Faugeras, O. (1998). Finding pose of hand in video images: a stereo-based approach.

Dellaert, F. und Thorpe, C. E. (1997). Robust car tracking using kalman filtering and bayesian templates. In *Conference on Intelligent Transportation Systems*.

Demirdjian, D. (2003). Enforcing constraints for human body tracking. *Computer Vision and Pattern Recognition Workshop*, 9:102.

Demirdjian, D. und Darrell, T. (2002). 3-d articulated pose tracking for untethered diectic reference. *Multimodal Interfaces, IEEE International Conference on*, 0:267.

Dhond, U. R. und Aggarwal, J. (1989). Structure from stereo - a review. *IEEE Transactions on Systems, Man, and Cybernectics*, 19:1489–1510.

Di Stefano, L., Marchionni, M., und Mattoccia, S. (2004). A pc-based real-time stereo vision system. *MGV*, 13(3):197–220.

Duda, R. O. und Hart, P. E. (1973). *Pattern classification and scene analysis*. John Wiley & Sons Inc.

Ebert, D. (2003). *Bildbasierte Erzeugung kollisionsfreier Transferbewegungen für Industrieroboter*. PhD thesis, Universität Kaiserslautern.

Eggert, D. W., Lorusso, A., und Fisher, R. B. (1997). Estimating 3-d rigid body transformations: a comparison of four major algorithms. *Machine Vision and Applications*, 9(5-6):272–290.

Elkmann, N., Fritzsche, M., Schulenburg, E., und Teutsch, C. (2008). Lisa: ein assistenzroboter für den einsatz in laborumgebungen zur sicheren mensch- roboter-interaktion. In *Robotik 2008, München*.

Faugeras, O. (1993). *Three-Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, Cambridge, MA, USA.

Faugeras, O. und Keriven, R. (1998). Variational principles, surface evolution, pde's, level set methods and the stereo problem. In *IEEE Trans. Image Processing*, volume 7, pages 336–344.

Fermüller, C. und Aloimonos, Y. (1995). Global rigidity constraints in image displacement fields. In *ICCV '95: Proceedings of the Fifth International Conference on Computer Vision*, page 245, Washington, DC, USA. IEEE Computer Society.

Fermüller, C. und Aloimonos, Y. (1997). On the geometry of visual correspondence. *Int. J. Comput. Vision*, 21(3):223–247.

Fermüller, C., Shulman, D., und Aloimonos, Y. (2001). The statistics of optical flow. *CVIU*, 82(1):1–32.

Fielding, G. und Kam, M. (1997). Applying the hungarian method to stereo matching. In *IEEE Conference on Decision and Control*, pages 549–558.

Fischler, M. A. und Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395.

Forsyth, D. A. und Ponce, J. (2002). *Computer Vision: A Modern Approach*. Prentice Hall.

Franke, U. und Joos, A. (2000). Real-time stereo vision for urban traffic scene understanding. In *Procs. IEEE Intelligent Vehicles Symposium 2000*, pages 273–278, Dearborn, USA.

Franke, U. und Kutzbach, I. (1996). Fast stereo based object detection for stop&go traffic. In *Proceedings of the 1996 IEEE Intelligent Vehicles Symposium*, pages 339–344.

Franke, U., Rabe, C., Badino, H., und Gehrig, S. (2005). 6d-vision: Fusion of stereo and motion for robust environment perception. In *DAGM '05*, Vienna.

Fua, P. (1991). Combining stereo and monocular information to compute dense depth maps that preserve depth discontinuities. In *International Joint Conference on Artificial Intelligence, Sydney, Australia*, pages 1292–1298.

Fusiello, A., Trucco, E., und Verri, A. (2000). A compact algorithm for rectification of stereo pairs. *Machine Vision and Applications*, 12(1):16–22.

Gavrila, D. (1999). The visual analysis of human movement: A survey. *CVIU*, 73(1):82–98.

Gavrila, D. M. und Davis, L. S. (1996). 3-d model-based tracking of humans in action: a multi-view approach. In *CVPR '96: Proceedings of the 1996 Conference on Computer Vision and Pattern Recognition (CVPR '96)*, page 73, Washington, DC, USA. IEEE Computer Society.

Goodrich, M. und Schultz, A. (2007). *Human-robot interaction*. Number 1,3 in Foundations and trends in human-computer interaction. Now, Boston, Mass. [u.a.].

Gövert, T. (2006). Konzeption und implementierung eines systems zur raumzeitlichen konturbasierten 3d-stereoanalyse im produktionsszenario. Master's thesis, Universität Bielefeld.

Haag, M. und Nagel, H.-H. (1999). Combination of edge element and optical flow estimates for 3d-model-based vehicle tracking in traffic image sequences. *International Journal of Computer Vision*, 35:295–319.

Hahn, M., Krüger, L., und Wöhler, C. (2008a). 3d action recognition and long-term prediction of human motion. In Gasteratos, A., Vincze, M., und Tsotsos, J., editors, *Proc. Int. Conf. on Computer Vision Systems, Santorini, Greece.*, volume 5008/2008 of *Lecture Notes in Computer Science*, pages 23–32. Springer-Verlag Berlin Heidelberg.

Hahn, M., Krüger, L., und Wöhler, C. (2008b). Spatio-temporal 3d pose estimation and tracking of human body parts using the shapeflow algorithm. In *Proc. Int. Conf. on Pattern Recognition, Tampa, USA*.

Hahn, M., Krüger, L., Wöhler, C., und Gross, H.-M. (2007). Tracking of human body parts using the multiocular contracting curve density algorithm. In *3DIM '07: Proceedings of the Sixth International Conference on 3-D Digital Imaging and Modeling*, pages 257–264, Washington, DC, USA. IEEE Computer Society.

Hamilton, W. R. (1866). *Elements of Quaternions*. Longmans, Green, & co.

Hanek, R. (2004). *Fitting Parametric Curve Models to Images Using Local Self-adapting Separation Criteria*. PhD thesis, Technische Universität München, München.

Haralick, R., Joo, H., Lee, C., Zhuang, X., Vaidya, V., und Kim, M. (1989). Pose estimation from corresponding point data. *IEEE Transactions on Systems, Man and Cybernetics*, 19(6):1426–1446.

Hartley, R. I. und Zisserman, A. (2004). *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition.

Heap, T. und Hogg, D. (1996). Towards 3d hand tracking using a deformable model. In *Proceedings of the 2nd International Conference on Automatic Face and Gesture Recognition (FG '96)*, pages 140–145.

Hermes, C., Barth, A., Wöhler, C., und Kummert, F. (2009). Object motion analysis and prediction in stereo image sequences. In *8. Oldenburger 3D-Tage*, Oldenburg.

Hirschmüller, H. (2005). Accurate and efficient stereo processing by semi-global matching and mutual information. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2*, pages 807–814, Washington, DC, USA. IEEE Computer Society.

Hirschmüller, H., Innocent, P. R., und Garibaldi, J. (2002). Real-time correlation-based stereo vision with reduced border errors. *Int. J. Comput. Vision*, 47(1-3):229–246.

Hirschmüller, H. und Scharstein, D. (2007). Evaluation of cost functions for stereo matching. In *Conference on Computer Vision and Pattern Recognition*. IEEE.

Hofmann, M. und Gavrila, D. M. (2009). Multi-view 3d human pose estimation combining single-frame recovery, temporal integration and model adaptation. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.

Horn, B. (1986). *Robot Vision*. MIT Press.

Horn, B. (1987a). Motion fields are hardly ever ambiguous. *IJCV*, 1(3):239–258.

Horn, B. K. und Schunck, B. G. (1980). Determining optical flow. Technical report, MIT, Cambridge, MA, USA.

Horn, B. K. P. (1987b). Closed-form solution of absolute orientation using unit quaternions. *J. Opt. Soc. Am. A*, 4(4):629.

Huber, P. J. (1981). *Robust Statistics*. John Wiley and Sons.

Huguet, F. und Devernay, F. (2007). A variational method for scene flow estimation from stereo sequences. In *IEEE Eleventh International Conference on Computer Vision, ICCV 07, Rio de Janeiro, Brazil*.

Ju, S. X., Black, M. J., und Yacoob, Y. (1996). Cardboard people: A parameterized model of articulated image motion. In *FG '96: Proceedings of the 2nd International Conference on Automatic Face and Gesture Recognition (FG '96)*, page 38, Washington, DC, USA. IEEE Computer Society.

Kämpchen, N. (2007). *Feature-Level Fusion of Laser Scanner and Video Data for Advanced Driver Assistance Systems*. PhD thesis, University Ulm.

Kanade, T. und Okutomi, M. (1994). A stereo matching algorithm with an adaptive window: Theory and experiment. *IEEE Trans. Pattern Anal. Mach. Intell.*, 16(9):920–932.

Kass, M., Witkin, A., und Terzopoulos, D. (1988). Snakes: Active contour models. *International Journal of Computer Vision*, V1(4):321–331.

Kato, T., Ninomiya, Y., und Masaki, I. (2002). An obstacle detection method by fusion of radar and motion stereo. *IEEE Transactions on Intelligent Transportation Systems*, 3(3):182–188.

Kleinehagenbrock, M., Lang, S., Fritsch, J., Lömker, F., Fink, G. A., und Sagerer, G. (2002). Person tracking with a mobile robot based on multi-modal anchoring. In *Proc. IEEE Int. Workshop on Robot and Human Interactive Communication*, pages 423–429, Berlin, Germany. IEEE, IEEE.

Knoop, S., Vacek, S., und Dillmann, R. (2006). Sensor fusion for 3d human body tracking with an articulated 3d body model. In *Proceedings 2006 IEEE International Conference on Robotics and Automation*, pages 1686 – 1691.

Koller, D., Danilidis, K., und Nagel, H.-H. (1993). Model-based object tracking in monocular image sequences of road traffic scenes. *Int. J. Comput. Vision*, 10(3):257–281.

Kollnig, H. und Nagel, H.-H. (1997). 3d pose estimation by directly matching polyhedral models to gray value gradients. *International Journal of Computer Vision*, 23:283–302.

Konrad, M. (2006). Segmentierung von mehrdimensionalen daten eines hochauflösenden kfz-radars. Master's thesis, Technische Universität Ilmenau.

Koschan, A. (1993). What is new in computational stereo since 1989: A survey on current stereo papers. Technical report, Technische Universität Berlin.

Kraus, K., Jansa, J., und Kager, H. (1997). *Photogrammetry, Vol. 2, Advanced Methods and Applications*. Dï¿½mmler.

Kreßel, U. und Schürmann, J. (1997). *Handbook of Character Recognition and Document Image Analysis*, chapter Pattern classification techniques based on function approximation, pages 49–78. World Scientific.

Krüger, L. (2007). *Model Based Object Classification and Localisation in Multiocular Images*. PhD thesis, University of Bielefeld.

Krüger, L. und Wöhler, C. (2009). Accurate chequerboard corner localisation for camera calibration and scene reconstruction. *Submitted to Pattern Recognition Letters*.

Lee, J. und Kunii, T. (1993). Constraint-based hand animation. *In Models and Techniques in Computer Animation*, pages 110ï¿½–127. Springer Verlag.

Leibe, B., Cornelis, N., Cornelis, K., und Gool, L. V. (2006). Integrating recognition and reconstruction for cognitive traffic scene analysis from a moving vehicle. In *DAGM Annual Pattern Recognition Symposium*, pages 192–201. Springer.

Leibe, B. und Schiele, B. (2004). Scale-invariant object categorization using a scale-adaptive mean-shift search. In *DAGM Annual Pattern Recognition Symposium*, pages 145–153.

Li, R. und Sclaroff, S. (2008). Multi-scale 3d scene flow from binocular stereo sequences. *Computer Vision and Image Understanding*, 110(1):75–90.

Liang Wang, Weiming Hu, T. T. (2003). Recent developments of human motion analysis. *Pattern Recognition*, 36(3):585–601.

Lin, M. H. (1999). Tracking articulated objects in real-time range image sequences. In *ICCV*, pages 648–653.

Liu, H., Hong, T.-H., Herman, M., Camus, T., und Chellappa, R. (1998). Accuracy vs efficiency trade-offs in optical flow algorithms. *Computer Vision Image Understanding*, 72(3):271–286.

Lowe, D. G. (1991). Fitting parameterized three-dimensional models to images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(5):441–450.

Lucas, B. und Kanade, T. (1981). An iterative image registration technique with an application to stereo vision. In *IJCAI81*, pages 674–679.

MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. In Cam, L. M. L. und Neyman, J., editors, *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press.

Marr, D. und Poggio, T. (1979). A computational theory of human stereo vision. *RoyalP*, B-204:301–328.

Marr, D. und Ullman, S. (1981). Directional selectivity and its use in early visual processing. *Proc. R. Soc. Lond. B*, 211:151–180.

Matthies, L., Maimone, M., Johnson, A., Cheng, Y., Willson, R., Villalpando, C., Goldberg, S., Huertas, A., Stein, A., und Angelova, A. (2007). Computer vision on mars. *International Journal of Computer Vision*.

Moeslund, T. B., Hilton, A., und Krüger, V. (2006). A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2):90–126.

Moeslund, T. B., Madsen, C. B., und Granum, E. (2005). Modelling the 3d pose of a human arm and the shoulder complex utilising only two parameters. *Integrated Computer-Aided Engineering*, 12.

Mündermann, L., Corazza, S., und Andriacchi, T. (2007). Accurately measuring human movement using articulated icp with soft-joint constraints and a repository of articulated models. In *CVPR07*, pages 1–6.

Mündermann, L., Corazza, S., und Andriacchi, T. P. (2006). The evolution of methods for the capture of human movement leading to markerless motion capture for biomechanical applications. *Journal of NeuroEngineering and Rehabilitation*, 3.

Murray, D. und Little, J. J. (2004). Segmenting correlation stereo range images using surface elements. In *Proceedings of the 3D Data Processing, Visualization, and Transmission*, pages 656–663.

Nagel, H. (1989). On a constraint equation for the estimation of displacement rates in image sequences. *PAMI*, 11(1):13–30.

Narayanan, K., Kumaran, R., und Gowdy, J. (2005). Stereo-based elliptical head tracking. In *Eusipco2005*.

Nedevschi, S., Danescu, R., Frentiu, D., Marita, T., Oniga, F., Pocol, C., Schmidt, R., und Graf, T. (2004). High accuracy stereo vision system for far distance obstacle detection. In *IVS*, pages 292–297.

Nedevschi, S., Danescu, R., Marita, T., Oniga, F., Pocol, C., Sobol, S., Tomiuc, C., Vancea, C., Meinecke, M. M., Graf, T., To, T. B., und Obojski, M. A. (2007). A sensor for urban driving assistance systems based on dense stereovision. In *Proceedings of the 2007 IEEE Intelligent Vehicles Symposium*, pages 276–283.

Noriega, P. und Bernier, O. (2007). Multicues 2d articulated pose tracking using particle filtering and belief propagation on factor graphs. In *IEEE International Conference on Image Processing (ICIP)*, pages V: 57–60.

Ott, R. (1977). *Über zweistufige, quadratische Klassifikation*. PhD thesis, Universität Erlangen.

Pedrotti, F. L., Pedrotti, L. S., Bausch, W., und Schmidt, H. (2005). *Optik für Ingenieure*. Springer, 3. edition.

Phong, T. Q., Horaud, R., Yassine, A., und Tao, P. D. (1996). Object pose from 2-D to 3-D point and line correspondences. *International Journal of Computer Vision*, 15(3):225–243.

Plänkers, R. und Fua, P. (2003). Articulated soft objects for multiview shape and motion capture. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(9):1182–1187.

Pons, J.-P., Keriven, R., Faugeras, O., und Hermosillo, G. (2003). Variational stereovision and 3d scene flow estimation with statistical similarity measures. In *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*, page 597, Washington, DC, USA. IEEE Computer Society.

Ramanan, M.-D., Forsyth, S. M.-D. A., und Zisserman, S. M.-A. (2007). Tracking people by learning their appearance.

Rey, W. J. J. (1983). *Introduction to Robust and Quasi-Robust Statistical Methods*. Springer-Verlag, Berlin.

Rosenhahn, Bodo, Kersting, Uwe, Powell, Katie, Klette, Reinhard, Klette, Gisela, Seidel, und Hans-Peter (2007a). A system for articulated tracking incorporating a clothing model. *Machine Vision and Applications*, 18(1):25–40.

Rosenhahn, B., Kersting, U., Smith, A., Gurney, J., Brox, T., und Klette, R. (2005). A system for marker-less human motion estimation. In Kropatsch, W., Sablatnig, R., und Hanbury, A., editors, *27th DAGM Symposium*, volume 3663 of *Lecture Notes in Computer Science*, pages 230–237, Vienna, Austria. Springer.

Rosenhahn, B., Klette, R., und Metaxas, D. (2007b). *Human Motion: Understanding, Modelling, Capture and Animation*. Springer.

Rosenhahn, B., Perwass, C., und Sommer, G. (2003). Pose estimation of free-form surface models. In *Pattern Recognition, Proc. 25th DAGM Symposium, LNCS 2781*, pages 574–581.

Rosenhahn, B., Schmaltz, C., Brox, T., Weickert, J., Cremers, D., und Seidel, H.-P. (2008). Markerless motion capture of man-machine interaction. In *CVPR*. IEEE Computer Society.

Rottensteiner, F., Summer, G., Trinder, J., Clode, S., und Kubik, K. (2005). Evaluation of a method for fusing lidar data and multispectral images for building detection. In *CMRT05*, pages xx–yy.

Roy, S. und Cox, I. J. (1998). A maximum-flow formulation of the n-camera stereo correspondence problem. In *ICCV '98: Proceedings of the Sixth International Conference on Computer Vision*, page 492, Washington, DC, USA. IEEE Computer Society.

Rusinkiewicz, S. und Levoy, M. (2001). Efficient variants of the icp algorithm. In *Proceedings of the Third Intl. Conf. on 3D Digital Imaging and Modeling*, pages 145–152.

Sappa, A., Aifanti, N., Grammalidis, N., und Malassiotis, S. (2005). *Advances in Vision-Based Human Body Modeling*, chapter 1, pages 1–26. IRM Press.

Scharstein, D. und Szeliski, R. (2002). A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47(1-3):7–42.

Schmidt, J., Fritsch, J., und Kwolek, B. (2006). Kernel particle filter for real-time 3d body tracking in monocular color images. In *Proc. of the 7th International Conference on Automatic Face and Gesture Recognition (FGR06)*, pages 567–572, Washington, DC, USA. IEEE Computer Society.

Schmidt, J., Wöhler, C., Krüger, L., Gövert, T., und Hermes, C. (2007). 3D scene segmentation and object tracking in multiocular image sequences. In *The 5th International Conference on Computer Vision Systems Conference Paper*.

Schreer, O. (2007). *Stereoanalyse und Bildsynthese*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Schunck, B. G. (1986). The image flow constraint equation. *Comput. Vision Graph. Image Process.*, 35(1):20–46.

Schunck, B. G. (1989). Image flow segmentation and estimation by constraint line clustering. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11(10):1010–1027.

Schürmann, J. (1996). *Pattern classification: a unified view of statistical and neural approaches*. John Wiley & Sons, Inc., NY, USA.

Sepehri, A., Yacoob, Y., und Davis, L. S. (2004). Estimating 3d hand position and orientation using stereo. In *ICVGIP*, pages 58–63.

Shi, J. und Malik, J. (1997). Normalized cuts and image segmentation. In *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, volume 0, page 731, Los Alamitos, CA, USA. IEEE Computer Society.

Shi, J. und Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905.

Siddiqui, M. und Medioni, G. (2006). Robust real-time upper body limb detection and tracking. In *Proceedings of the 4th ACM international workshop on Video surveillance and sensor networks*, pages 53–60.

Sigal, L., Sigal, L., Black, M. J., und Black, M. J. (2006). Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion. Technical report, Brown University.

Simon, D. A. (1996). *Fast and accurate shape-based registration.* PhD thesis, Carnegie Mellon University, Pittsburgh, PA, USA. Chair-Kanade,, Takeo.

Slama, C. (1980). *Manual of Photogrammetry.* Book.

Slesareva, N., Bruhn, A., und Weickert, J. (2005). Optic flow goes stereo: A variational method for estimating discontinuity-preserving dense disparity maps. In *In W. Kropatsch, R. Sablatnig, A. Hanbury (Eds.): Pattern Recognition. Lecture Notes in Computer Science, Vol. 3663, 33-40, Springer, Berlin.*

Sminchisescu, C. (2006). 3d human motion analysis in monocular video techniques and challenges. In *AVSS '06: Proceedings of the IEEE International Conference on Video and Signal Based Surveillance*, Washington, DC, USA. IEEE Computer Society.

Spiengler, J. und Thiemermann, S. (2002). Direkte mensch-roboter kooperation in der flexiblen montagezelle. In *Robotik 2002 VDI-Berichte Nr. 1679*, pages 191–195.

Stein, A., Huertas, A., und Matthies, L. (2006). Attenuating stereo pixel-locking via affine window adaptation. In *IEEE International Conference on Robotics and Automation*, pages 914 – 921.

Stein, F. (2004). Efficient computation of optical flow using the census transform. In *DAGM04*, pages 79–86.

Steux, B., Laurgeau, C., Salesse, L., und Wautier, D. (2002). Fade: a vehicle detection and tracking system featuring monocular color vision and radar data fusion. In *IEEE Intelligent Vehicle Symposium*, volume 2, pages 632–639.

Sun, J., Li, Y., Kang, S. B., und Shum, H.-Y. (2005). Symmetric stereo matching for occlusion handling. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2*, pages 399–406, Washington, DC, USA. IEEE Computer Society.

Sun, Z., Bebis, G., und Miller, R. (2006). On-road vehicle detection: A review. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(5):694–711.

Sundaresan, A. und Chellappa, R. (2006). Multi-camera tracking of articulated human motion using motion and shape cues. In *Asian Conference on Computer Vision*, pages II:131–140.

Thiemermann, S. (2002). *Direkte Mensch-Roboter-Kooperationin der Kleinteilemontage mit einem SCARA-Roboter*. PhD thesis, Universität Stuttgart.

Thrun, S., Burgard, W., und Fox, D. (2005). *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press.

Tomasi, C. und Kanade, T. (1991). Detection and tracking of point features. Technical Report CMU-CS-TR-91-132, Carnegie Mellon University.

Tonko, M. und Nagel, H. H. (2000). Model-based stereo-tracking of non-polyhedral objects for automatic disassembly experiments. In *Int.J. of Computer Vision*, volume 37, pages 99–ï¿½118.

Toulminet, G., Bertozzi, M., Mousset, S., Bensrhair, A., und Broggi, A. (2006). Vehicle detection by means of stereo vision-based obstacles features extraction and monocular pattern analysis. *IEEE Transactions on Image Processing*, 15(8):2364–2375.

Triggs, B., McLauchlan, P. F., Hartley, R. I., und Fitzgibbon, A. W. (2000). Bundle adjustment - a modern synthesis. In *ICCV '99: Proceedings of the International Workshop on Vision Algorithms*, pages 298–372, London, UK. Springer-Verlag.

Ullman, S. (1979). *The Interpretation of Visual Motion*. MIT Press.

van der Mark, W. und Gavrila, D. M. (2006). Real-time dense stereo for intelligent vehicles. *Intelligent Transportation Systems, IEEE Transactions on*, 7(1):38–50.

Vedula, S., Baker, S., Rander, P., Collins, R., und Kanade, T. (1999). Three-dimensional scene flow. In *International Conference on Computer Vision (ICCV)*, pages 722 – 729. IEEE Computer Society.

Venkateswar, V. und Chellappa, R. (1995). Hierarchical stereo and motion correspondence using feature groupings. *Int. J. Comput. Vision*, 15(3):245–269.

von Bank, C., Gavrila, D., und Wöhler, C. (2003). A visual quality inspection system based on a hierarchical 3d pose estimation algorithm. In *DAGM-Symposium*, pages 179–186.

Wallach, H. (1935). Über visuell wahrgenommene bewegungsrichtung. *Psychologische Forschung*, 20:325–380.

Wedel, A., Rabe, C., Vaudrey, T., Brox, T., Franke, U., und Cremers, D. (2008). Efficient dense scene flow from sparse or dense stereo data. In *ECCV '08: Proceedings of the 10th European Conference on Computer Vision*, pages 739–751, Berlin, Heidelberg. Springer-Verlag.

Wender, S. und Dietmayer, K. (2007). 3d vehicle detection using a laser scanner and a video camera. In *In Proceedings of 6th European Congress on ITS in Europe*, Aalborg, Denmark.

Wöhler, C. und Anlauf, J. K. (2001). Real-time object recognition on image sequences with the adaptable time delay neural network algorithm - applications for autonomous vehicles. *Image and Vision Computing*, 19:593 – 618.

Winkler, B. (2007). Safe space sharing human-robot cooperation using a 3d time-of-flight camera. In *International Robots and Vision Show*.

Winner, H., Hakuli, S., und Wolf, G. (2009). *Handbuch Fahrerassistenzsysteme*. Vieweg+Teubner Verlag / GWV Fachverlage GmbH, Wiesbaden.

Wöhler, C. (2009). *3D Computer Vision. Efficient Methods and Applications*. Springer Verlag Berlin Heidelberg.

Wöhler, C. und Krüger, L. (2003). A contour-based stereovision algorithm for video surveillance applications. In Ebrahimi, T. und Sikora, T., editors, *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 5150 of *Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference*, pages 102–109.

Wren, C., Azarbayejani, A., Darrell, T., und Pentland, A. (1997). Pfinder: Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:780–785.

Yan, W. und Forsyth, D. A. (2004). Learning the behavior of users in a public space through video tracking. Technical Report UCB/CSD-04-1310, EECS Department, University of California, Berkeley.

Zabih, R. und Woodfill, J. (1994). Non-parametric local transforms for computing visual correspondence. In *ECCV (2)*, pages 151–158.

Zach, C., Pock, T., und Bischof, H. (2007). A duality based approach for realtime tv-l1 optical flow. *Published at the Annual Symposium of the German Association for Pattern Recognition (DAGM 2007)*.

Zhang, Z. (1992). Iterative point matching for registration of free-form curves. Technical report, INRIA.

Ziegler, J., Nickel, K., und Stiefelhagen, R. (2006). Tracking of the articulated upper body on multi-view stereo image sequences. In *CVPR '06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 774–781, Washington, DC, USA. IEEE Computer Society.