TECHNISCHE UNIVERSITÄT
ILMENAU

# Ein Beitrag zur Entwicklung von Methoden zur Stereoanalyse und Bildsynthese im Anwendungskontext der Videokommunikation

## Dissertation

zur Erlangung des akademischen Grades
Doktoringenieur (Dr.-Ing.)

vorgelegt der Fakultät für Elektrotechnik und Informationstechnik
der Technischen Universität Ilmenau

von Dipl.-Ing. Christian Weigel
geboren am 06. Februar 1977 in Potsdam

Gutachter: Prof. Dr.-Ing. Dr. rer. nat. h.c. mult. Karlheinz Brandenburg (TU Ilmenau)
Prof. Dr.-Ing. Marcus Magnor (TU Braunschweig)
Dr.-Ing. Karsten Müller (Fraunhofer HHI, Berlin)

Tag der Einreichung: 05.11.2013
Tag der wissenschaftlichen Aussprache: 27.08.2014

urn:nbn:de:gbv:ilm1-2014000269

# Kurzfassung

Die vorliegende Arbeit leistet einen Beitrag zum Forschungsbereich der Stereoanalyse und Bildsynthese im speziellen Kontext der privaten Videokommunikation. Bei der privaten Videokommunikation geht durch die unterschiedliche Positionierung der Kamera und des Videofensters typischerweise der Blickkontakt zwischen den Kommunikationsteilnehmern verloren. Ziel dieser Arbeit ist die Wiederherstellung des Blickkontaktes mittels der Synthese einer virtuellen Kameraansicht, die in Blickrichtung der Kommunizierenden ausgerichtet ist.

Die Arbeit umreißt zunächst den positiven Einfluss des Blickkontaktes in der Videokommunikation. Anschließend wird eine tiefgehende Betrachtung der notwendigen technischen Grundlagen im Bereich Stereoanalyse und Bildsynthese durchgeführt. Aufbauend auf diesen Grundlagen wird der der Stand der Technik im Bereich des bildbasierten Renderings im Allgemeinen sowie der Blickkorrektur mittels 3D-Analyse und -synthese im Speziellen umfassend behandelt.

Zunächst wird ein Modell von Qualitätsparametern entwickelt, welches die Entscheidungen hinsichtlich Kameraanordnung und Aufnahmesystem determiniert. Notwendige Messungen hinsichtlich Synchronizität und Datenspeicherung werden präsentiert. Im Bereich der Algorithmen der Stereoanalyse werden etablierte lokale und globale Algorithmen analysiert und adaptiert. Verschiedene Kostenmaße, konsistenzbasiertes Füllen, zeitliche und örtliche Glättung sowie eine abschließende Segmentierung werden hinsichtlich des konkreten Anwendungsfalls der Blickkorrektur in der privaten Videokommunikation entwickelt. Darauf aufbauend werden die beiden Syntheseverfahren des trifokalen Transfers sowie des 3D-Warpings weiter entwickelt. Ein wichtiger Beitrag der Arbeit ist ein konturbasiertes Füllverfahren sowie Maßnahmen im Bereich der Punktglättung.

Zwei umfangreiche Experimente mit zahlreichen Probanden bestätigen die Korrektheit der Annahme, dass Blickkontakt durch das vorgestellte Verfahren hergestellt werden kann. Sie demonstrieren sowohl die sehr gute Wahrnehmung des Augenkontaktes als auch die signifikante Verbesserung der Akzeptanz und subjektiven Qualitätswahrnehmung durch die entwickelten Algorithmen im Vergleich zum Ausgangspunkt der Arbeit. Eine qualitativer Vergleich mit dem Stand der Technik und eine Diskussion der Ergebnisse, gepaart mit einem Ausblick in die Zukunft des behandelten Forschungsgebietes, schließen die Arbeit ab.

# Abstract

This thesis contributes to the research area of stereo vision and view synthesis in the field of private video communication. During private video communication eye contact between the participants is typically lost due to the different placement of the camera and the video window. The goal of this thesis is to re-establish the eye contact by synthesizing of the view of a virtual camera such that the virtual camera faces towards the participant.

The thesis firstly sketches the positive effect of eye contact in video communication. An in-depth review of mathematical foundations in the fields of stereo vision and view synthesis follows. On this foundation the thesis comprehensively covers the state of the art of image based rendering and particularly of eye-gaze correction via 3D-analysis and synthesis. In the first step of the method development the thesis establishes a model of quality factors which determines decisions about camera placement and recording system. Measurements with respect to synchronization and data storage are presented. Local and global algorithms for stereo vision are analyzed and adapted. The thesis contributes to the field of stereo vision algorithms by means of development and combination of different cost functions, consistency based inpainting, spatial and temporal smoothing and segmentation with respect to the use case of private video communication. Using the extracted disparity map, two approaches for view synthesis – trifocal transfer and 3D warping – are employed and extended. One important contribution of the thesis is a contour-based inpainting algorithm as well as point base image smoothing techniques.

Two comprehensive subjective studies prove the assumption that eye contact can be re-established by the proposed system. They demonstrate the well perceived eye-contact as well as the significantly improved acceptance of quality due to the developed methods compared to the initial situation. The thesis finally discusses the results, followed by a qualitative comparison to the state of the art.

# Danksagung

Ich möchte mich an dieser Stelle bei allen bedanken, die direkt oder indirekt zur Entstehung dieser Arbeit beigetragen haben. Meinem Doktorvater Prof. Dr.-Ing. Dr. rer. nat. h.c. mult. Karlheinz Brandenburg für die Möglichkeit der Durchführung der Arbeit im Fachgebiet Elektronische Medientechnik. All meinen ehemaligen Kollegen am IMT für Motivation und Unterstützung, vor allem meiner Projektpartnerin Sara Kepplinger sowie Andreas Koch und Bernd Hildenbrandt. Ich danke meinen Diplomanden Leif Lennart Kreibich, Peter Schübel, Martin Wallebohr, Thomas Korn, Julia Schmidt, Rene Döhring und Sebastian Schwarz sowie Maxim Volkov für ihre wertvollen Denk- und Programmierarbeiten. Meiner Mutter danke ich für diverse Seiten Korrekturlesen.

Mein größter Dank gilt meiner Frau Rositsa für die grenzenlose Unterstützung und Geduld sowie meinen beiden Kindern Dominik und Adelina für das Verständnis, dass sie aufbrachten, wenn sie mal wieder auf Papa verzichten mussten.

# Erklärung

Ich versichere, dass ich die vorliegende Arbeit ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Die aus anderen Quellen direkt oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet.

Bei der Auswahl und Auswertung folgenden Materials haben mir die nachstehend aufgeführten Personen in der jeweils beschriebenen Weise unentgeltlich geholfen:

1. Frau Dipl.-Ing. (FH) Sara Kepplinger (Erfassung und Aufbereitung der subjektive erhobenen Daten der in Kapitel 6 vorgestellten Experimente innerhalb des DFG-Forschungsprojektes „Skalierbare Algorithmen für 3D Videoobjekte unter Berücksichtigung subjektiver Qualitätsfaktoren“.)

Weitere Personen waren an der inhaltlich-materiellen Erstellung der vorliegenden Arbeit nicht beteiligt. Insbesondere habe ich hierfür nicht die entgeltliche Hilfe von Vermittlungs- bzw. Beratungsdiensten (Promotionsberater oder anderer Personen) in Anspruch genommen.

Niemand hat von mir unmittelbar oder mittelbar geldwerte Leistungen für Arbeiten erhalten, die im Zusammenhang mit dem Inhalte der vorgelegten Dissertation stehen.

Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder ähnlicher Form einer Prüfungsbehörde vorgelegt.

Ich bin darauf hingewiesen worden, dass die Unrichtigkeit der vorstehenden Erklärung als Täuschungsversuch bewertet wird und gemäß § 7 Abs. 10 der Promotionsordnung den Abbruch des Promotionsverfahrens zur Folge hat.

(Ort, Datum) (Unterschrift)

# Allgemeine Hinweise zur Arbeit

An dieser Stelle folgen allgemeine Hinweise zur Form und Notation der Arbeit. Die Arbeit enthält viele Abbildungen zur Veranschaulichung von Sachverhalten. Die Größe der Abbildungen ist für die Betrachtung auf einer DIN A4-Seite ($210\ \mathtt{mm} \times 297\ \mathtt{mm}$) ausgelegt. Die elektronische Version enthält Pixelbilder üblicherweise unkomprimiert, jedoch können Versionen existieren, bei denen diese komprimiert wurden. Die Bilder der Testvideos haben die originale Größe von $640\ \mathtt{px} \times 480\ \mathtt{px}$. Der schriftlichen Arbeit liegt eine DVD mit den Videos bei.

Mehrfach verwendeten mathematischen Symbole sind im Anhang auf Seite 167 aufgeführt. Die Notationen orientieren sich an den im behandelten Forschungsbereich üblichen Notationen. Es wurde darauf geachtet größtmögliche Konsistenz in der Verwendung zwischen den Kapiteln zu erhalten. Auf eine separate Kennzeichnung in der Notation homogener Vektoren wurde verzichtet, da sich aus dem Kontext ergibt, ob es sich um die homogen Repräsentation eines Vektors handelt. Matrizen werden stets groß und gerade geschrieben (Bsp. $\mathtt{H}$). Vektoren im $\mathbb{R}^2$ bzw. $\mathbb{P}^2$ klein und fett (Bsp: $\mathbf{v}$). Vektoren im $\mathbb{R}^3$ bzw. $\mathbb{P}^3$ groß und fett (Bsp: $\mathbf{V}$). Skalare sind stets kursiv (Bsp. $d$, $X$).

Dezimaltrennzeichen reeller Zahlen wurden bewusst in der für deutsche Texte untypischen englischen Notation mit Punkt geschrieben(Bsp: $3.14$). Der Grund dafür liegt in der Verwendung von Matlab und anderen Werkzeugen währen der Arbeit, deren Ausgabe stets diese Notation verwendet und so eine fehlerfreie Übernahme von (Meß-)werten vereinfachte.

# Inhaltsverzeichnis

# 1. Einleitung

Kommunikation zwischen Menschen über weite Entfernungen mittels technischer Hilfsmittel ist heutzutage allgegenwärtig. Neben der auditiven Kommunikation durch Telefone gewinnt die audiovisuelle Kommunikation - die Übertragung von Ton- *und* Bilddaten - zunehmend an Bedeutung. Besonders durch mit Kameras ausgestatteten Computern und entsprechende Software, zunehmend auch durch leistungsfähige Mobiltelefone, wird diese Entwicklung vorangetrieben.

Diese Arbeit beschäftigt sich mit der Lösung eines bekannten Problems in Videokommunikationsanwendungen: Das Problem des *mangelnden Blickkontaktes*. Es entsteht dadurch, dass Blickkontakt zwischen den Kommunikationspartnern nur entsteht, wenn diese jeweils in die sie aufnehmende Kamera blicken. Das Gegenüber hat dadurch den Eindruck angesehen zu werden. Um den Kommunikationspartner zu betrachten ist der Blick jedoch meist auf dessen Darstellung auf dem Bildschirm gerichtet, wobei der Augenkontakt nicht hergestellt werden kann. Abbildung 1.1a verdeutlicht den Sachverhalt.

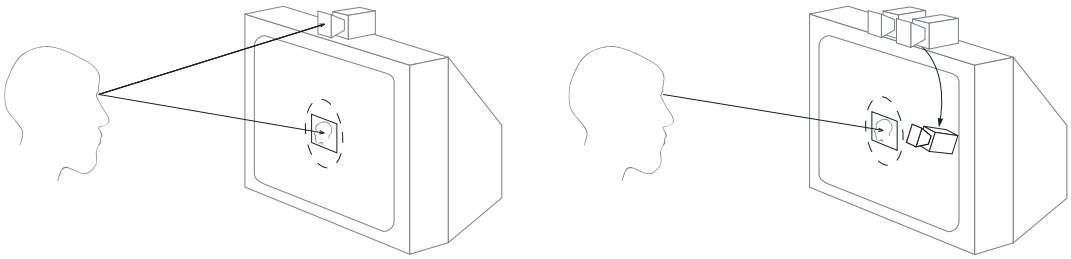
Die Erzeugung einer virtuellen Ansicht mittels Bildanalyse- und Bildsynthesealgorithmen jeweils aus der Position des dargestellten Kommunikationspartners kann dieses Problem beheben. Das Vorgehen kann auch als Blickkorrektur verstanden werden, wobei die neue Blickrichtung durch die veränderte Ansicht auf den Kommunikationspartner simuliert wird (Abb. 1.1b).

Bevor die Aufgabenstellung der Arbeit sowie technische Hintergründe näher erläutert werden, soll an dieser Stelle die Motivation durch eine kurzen geschichtlichen Abriss der Videokommunikation und deren Bedeutung begründet werden.

## 1.1. Videokommunikation

Zunächst sei der Begriff *Videokommunikation* in Anlehnung an die Definition von Wilcox definiert [Wilcox und Gibson, 2005]: Videokommunikation ist der gegenseitige Echtzeit-Austausch von digitalem Video und Audio zwischen zwei oder mehreren Teilnehmern an zwei oder mehreren Orten.

Diese Definition lässt sich weiter differenzieren in die Formen *Videokonferenz* und *Videochat*. Erstere impliziert üblicherweise die Teilnahme von mehr als zwei Personen und meist auch



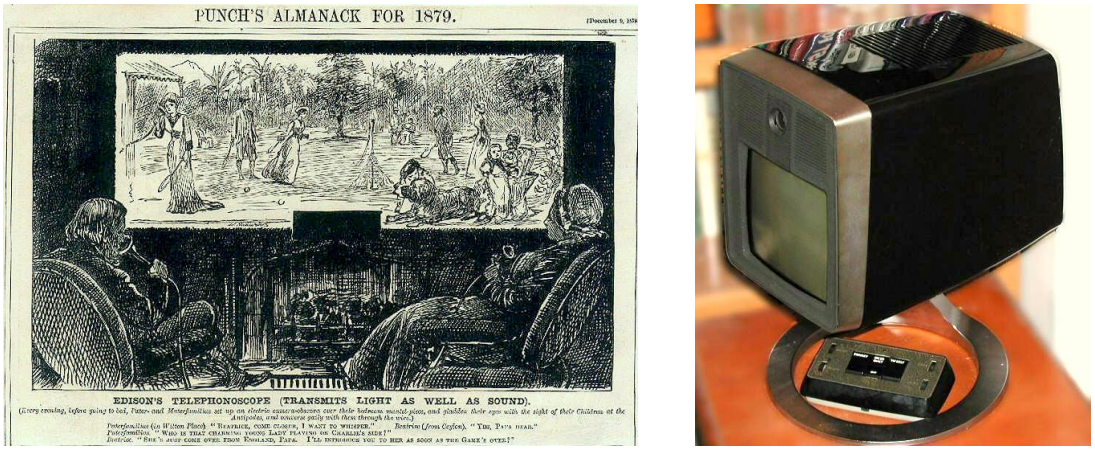
(a) (b)

**Abbildung 1.1.:** Schematische Darstellung des Problems des „Nicht angesehen werdens“ bei der Videokommunikation. (a) Der Teilnehmer richtet seinen Blick auf die Darstellung seines Kommunikationspartners auf dem Bildschirm (unterer Sichtstrahl) und nicht in die Kamera (oberer Sichtstrahl). (b) Durch Verfahren der Stereoanalyse und Bildsynthese kann eine virtuelle Kamera an der Position der Darstellung des Kommunikationspartners erzeugt werden. Durch die Ausrichtung auf den Teilnehmer entsteht Blickkontakt, obwohl dieser nicht in die real vorhandene Kameras sieht. Nach [Korn, 2009]

die Verbindung mehrerer Orte. Beim Videochat ist oft nur die Kommunikation zwischen zwei Personen gemeint. Die Begrifflichkeiten haben auch eine technische Komponente. So sind Videokonferenzesysteme technisch aufwändige Konstrukte im Firmenbereich, mit denen versucht wird, eine reale Konferenzsituation nachzustellen. Bei Videochats kommen eher einfache Webcams, ggf. ein Headset und ein PC im Heimbereich zum Einsatz. Die begrifflichen Grenzen verschwimmen jedoch zusehends. Im Kontext dieser Arbeit ist ausschließlich der visuelle Teil bei Videochats Gegenstand der Betrachtung. Eine aufwändige und somit für den Massenmarkt zu teure Lösung zu entwickeln, ist nicht das Ziel. Die auditive Komponente wird in den folgenden Kapiteln nicht weiter betrachtet, obschon ihrer Bedeutung im Gesamtkontext nicht vernachlässigt werden darf.

### 1.1.1. Geschichte und Bedeutung

In der Geschichte der Videokommunikation gab es bereits kurz nach der Erfindung des Telefons Ideen auch Bilder und Bildsequenzen elektrisch zu übertragen. Erste künstlerische Visionen entstanden schon 1878 durch George du Maurier (vgl. Abb.1.2a). Alexander Graham Bell formulierte 1891 erstaunlich präzise Gedanken zu einem System zur Wandlung und Wiedergabe von Licht [Bell, 1891]. Erste existierende Geräte waren eng mit der Entwicklung des Fernsehens verknüpft und wurden beispielsweise 1927 von AT&T demonstriert [Unbekannter Autor, 2011]. 1937 bot die Deutsche Reichspost öffentliche Video-Telefonzellen als „*Gegenseh-Fernsprechanlagen*“ [Weiher und Wagner, 1991, S. 152]. Für den Heimbereich konzipiert war das AT&T *Picturephone*, welches 1967 vorgestellt wurde [Bacon, 1968; Molnar, 1969] (vgl. Abb. 1.2b). Ein ähnliches System für die gewerbliche Nutzung gab es 1970 in Frankreich mit dem „*Visiophone*“ [Unbekannter Autor,




(a) (b)

**Abbildung 1.2.:** Vision und reales Beispiel für Videokommunikationssysteme. (a) Vision eines Systems auf einer Postkarte von George du Maurier (b) AT&T Pixturephone. Quelle: Wikimedia Commons

1971]. Die Produktion aller System wurde nach kurzer Zeit wieder eingestellt. Kommerzielle Systeme, die digitale Übertragung nutzten, wurden zu Beginn der 1980er Jahre durch die Firma *Compression Labs* eingeführt (VTS 1.5) [Wilcox und Gibson, 2005]. 1984 entwickelte *PictureTel* als Ausgründung des Massachusetts Institute of Technology ein erstes System als Konkurrenz. Es benötigte geringere Bitraten (224 Kbps) zur Übertragung und war erstmals auch als Softwarelösung verfügbar. 1984 folgte *VTEL* mit einem System, welches auf DOS-basierten PCs lief. Beide Systeme erforderten Zusatzhardware. Trotz sinkender Kosten und Anforderungen an Hardware und Übertragungsstrecke blieben die Systeme meist Speziallösungen im Firmenbereich. Primäre befürwortende Argumente war die Möglichkeit von Konferenzen ohne Reisen und die daraus resultierende Kosten- und Zeitersparnis [Wilcox und Gibson, 2005]. Die Etablierung von Standards für Video- und Audiokodierung wie H.261, H.263 [ITU-T, 1993, 1996a] und Protokolle wie H.323 und H.324 [ITU-T, 1996a,b], immer billigere und leistungsfähigere Hardware sowie günstigere Übertragungsbandbreite führten zur zunehmenden Verbreitung von Videokommunikationssystemen im Unternehmensbereich. Apple führte 1993 mit *Share View* ein System für den Mac, Intel 1994 mit *ProShare* ein System für den PC ein. Im Heimbereich wurde Videokommunikation zunächst durch Zusatzkarten ermöglicht, die die Analog-Digital Wandlung, Internetanbindung und Teile der Audiokompression übernahmen, z. B. von der Teles AG (vgl. [Gulich, 1998, S. 238ff]. Eingesetzte Software zu dieser Zeit war u. a. *NetMeeting* von Microsoft oder *CuSeeMe*, die nach den ITU-Standards arbeiteten, mitunter aber eigene Codecs einsetzten. Schnellere Prozessoren und die Integration von Multimediakomponenten in das PC-System machten in den folgenden Jahren zunächst Audiogespräche über Computer per *Voice over IP* (VoIP) möglich und populär. Der schnell am weitesten verbreitete Dienst für VoIP *Skype*

wurde 2003 gegründet. Durch die Einführung von Videochat in Skype im Januar 2006 wurde diese Funktion der bereits sehr großen Anzahl von Skype-Nutzern mit einem Mal verfügbar gemacht. Skype nutzt proprietäre Audio- und Videocodecs. Alternativen zu Skype waren und sind heutzutage beispielsweise Apple *FaceTime*, *Google Talk* und kürzlich das browserbasierte *Google Hangout*. Freie und quelloffene Lösungen, welche u. a. die ITU-Standards für Protokoll und Codecs benutzen, existieren mit *Ekiga*, *Jitsi* oder *homer* ebenso wie proprietäre Lösungen z. B. *goober* und *iVisit*, die kostenlose Software mit einem Dienstangebot bündeln. Die Verfügbarkeit billiger Kameras und Breitbandanschlüsse ermöglicht somit jedem Computerbesitzer Videochats. Zunehmend findet auch eine Verbreitung auf Smartphones und TV-Geräten statt. All dies zeigt, dass diverse technische Lösungen existieren, um Videochats einer breiten Masse verfügbar zu machen. Deren Verwendung hängt aber auch von Marketingstrategie und anderen Faktoren ab, die nicht Gegenstand dieser Arbeit sind. Dennoch könnten die in dieser Arbeit entwickelten Algorithmen die Akzeptanz von Videochats in der Zukunft weiter steigern.

### 1.1.2. Blickkontakt in der Videokommunikation

Die Motivation der Arbeit, durch Bildverarbeitungsalgorithmen Blickkontakt in Videokommunikationsszenarien herzustellen, wird durch eine Anzahl kommunikationswissenschaftlicher Studien gestützt. Zunächst ist festzustellen, dass direkter Augenkontakt in Kommunikationssituationen eher selten ist. Vielmehr „wandert“ der Blick über den gesamten Gesichtsbereich wobei jedoch Augen und Mundregion am meisten betrachtet werden [Ellgring, 1995, S. 27ff] (vgl. Abb. 1.3). Kommunikationswissenschaftler nutzen daher auch weniger den Begriff „Blickkontakt“ sondern stattdessen z. B. „Anblicken“. In dieser Arbeit wird er synonym verwendet. Die Dauer des Anblickens in einer Kommunikationssituation ist unterschiedlich, je nachdem ob gesprochen (40% ) oder zugehört (70%) wird [Ellgring, 1995, S. 27ff] [Argyle, 1988, S. 159]. Im Einzelfall sind jedoch starke Abweichungen dieser Werte zu beobachten. Diese sind durch weitere determinierende Faktoren wie den Charakter, die Beziehung zwischen den Kommunikationspartnern, das Gesprächsthema und die Gesprächssituation zu erklären. Die Bedeutung und positiven Auswirkungen von Blickkontakt in konventionellen Gesprächen wurde durch Argyle u. a. gezeigt [Argyle und Dean, 1965; Argyle und Ingham, 1972]. So signalisieren Konversationspartner durch Augenkontakt Gesprächsbereitschaft [Tang, 2007], regulieren soziale Nähe, erzeugen Intimität und erhöhte Aufmerksamkeit oder bestimmen den Gesprächsfluss [Argyle und Dean, 1965; Kendon, 1967; Kleinke, 1986]. Argyle weist auch den Zusammenhang zwischen der Häufigkeit von Blickkontakt und der Distanz zwischen den Gesprächspartnern nach, wobei bei Annäherung in einen bestimmten Bereich der Blickkontakt abnimmt[Argyle und Ingham, 1972]. In [Colburn u. a., 2000; Garau u. a., 2001; Vertegaal u. a., 2001] wird festgestellt, dass Blickkontakt bei der Kommunikation mit virtuellen Avataren eine Verbesserung der Kommunikation bewirkt. Andere Untersuchungen bestätigen dies auch für die Videokommunikation unter realen Personen. So

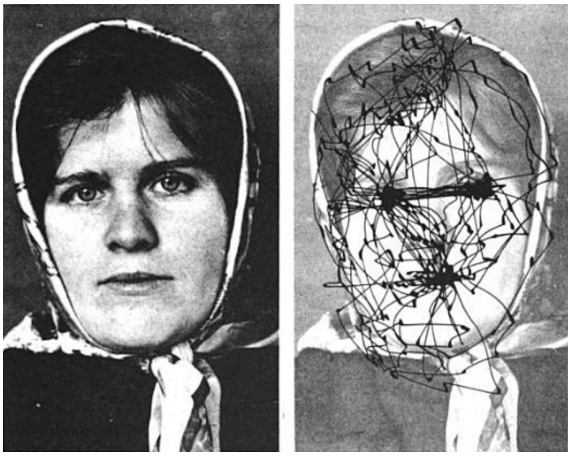


**Abbildung 1.3.:** Weg des Blickes beim Ansehen eines Gesichtes. Es ist klar zu erkennen, dass Augen- und Mundregionen am meisten angesehen werden. Aus [Yarbus, 1967; Argyle, 1988]

wird in [Vertegaal u. a., 2000] ein positiver Effekt auf den Gesprächsablauf festgestellt und in [Mukawa u. a., 2005], dass sich Blickkontakt positiv auf die Etablierung des Gesprächs auswirkt. In [van Eijk u. a., 2010] wurde kürzlich untersucht, wie weit der Blick in die Kamera in 2D- und stereoskopischen Kommunikationsszenarien abweichen darf, um sich noch angesehen zu fühlen. Das Ergebnis war u. a., dass bereits Winkel von mehr als $1.2\,^\circ$ horizontal und $1.7\,^\circ$ vertikal im $2\,\mathtt{m}$ Abstand ausreichen, um den Blickkontakt zu verlieren. Dies gilt sowohl für die monoskopische als auch für stereoskopische Darstellung. Auch technische Konstruktionen um Blickkontakt herzustellen, seien es der einfache Teleprompter im Bereich des Fernsehens [Oppenheimer, 1960, 1959] oder aufwändige Hardware-Kommunikationssysteme mit halbdurchlässigen Spiegeln wie in [Ishii und Kobayashi, 1992; Okada u. a., 1994] zeugen von dem Wunsch, angeblickt zu werden und anblicken zu können.

Die psychologischen Hintergründe und Effekte sollen in dieser Arbeit nicht weiter behandelt werden. Vielmehr besteht die Motivation darin, die durch technische Unzulänglichkeiten erzeugte unnatürliche Situation des Nicht-Anblickens in der Videokommunikation zu beseitigen. Geschieht dies möglichst unmerkbar für die Teilnehmer, so ist ihnen letztlich selbst überlassen, ob der Kommunikationspartner angesehen wird oder nicht.

## 1.2. Systematik der Arbeit

Im verbleibenden Teil dieses Kapitels wird zunächst ein allgemeines Verständnis der Problemstellung vermittelt. Anschließend stellt die Arbeit notwendige mathematische Grundlagen vor und erklärt im Detail gängige Verfahren im Bereich Stereoanalyse und Bildsynthese. Diese sind elementare Betrachtungsgegenstände des vorgeschlagenen Verfahrens. Ausgehend von einer detaillierten Betrachtung und Diskussion des aktuellen Standes der Technik im konkreten

Anwendungsfall der Videokommunikation wird die allgemeine Forschungsfrage konkretisiert. Anschließend werden neue Methoden und Ansätze zu deren Lösung entwickelt. Eine Evaluation der Ergebnisse schließt die Arbeit ab.

## 1.3. Hintergrund und Entwicklung der Forschungsfrage

Algorithmen der Stereoanalyse sind seit Beginn der digitalen Bilderverarbeitung ein wichtiger Forschungsgegenstand. Ausgehend von mindestens zwei zweidimensionalen Abbildern einer natürlichen Szene durch Foto- oder Videokameras ist das Ziel der Algorithmen die Extraktion der Tiefeninformationen der aufgenommen Szene aus deren Abbildungen in den Kameras. Die zunehmende Rechenleistung von Computern sowie die neu entwickelte mathematische Formulierung geometrischer Zusammenhänge haben in den letzten Jahren erhebliche Fortschritte in der automatischen Tiefenanalyse bewirkt und eine Vielzahl von Verfahren hervorgebracht. Waren erste Einsatzgebiete zunächst auf eine Bestimmung spärlicher Tiefeninformationen ausgerichtet, so ist heutzutage die Extraktion einer kompletten dreidimensionalen Repräsentation der Szene das Ziel.

Ausgehend von dieser Repräsentation ist es möglich, eine neue, so genannte virtuelle Ansicht auf die Szene zu synthetisieren und somit eine „freie“ Betrachtung dieser Szene zu ermöglichen. Auch diese Synthesealgorithmen sind ein aktueller Forschungsgegenstand. Unabhängig von der Repräsentationsform der extrahierten 3D-Szene und von den Algorithmen zur Bildsynthesen wird der Vorgang der Stereo-Bildanalyse und Synthese unter dem Terminus *Image Based Rendering* (Bildbasiertes Rendering) zusammengefasst. Je nach Art und Umfang der extrahierten 3D-Repräsentation sind verschiedene Anwendungsgebiete dafür denkbar, z. B. sind die Einbettung der synthetisierten Ansicht in synthetische 3D-Welten, die Unterstützung zur Erzeugung von Ansichten für autostereoskopische Displays, die Nutzung für die Stereo- und *Multi-View*-Kodierung (Mehransichten) oder die Verwendung im Kontext von Videokommunikation.

Die Abhandlungen in dieser Arbeit konzentrieren sich auf den letztgenannten Anwendungsfall. Die Verwendung von zwei Kameras in der Videokommunikation dient momentan der stereoskopischen Aufnahme und Wiedergabe der beteiligten Kommunikationspartner. Hierbei findet keinerlei Analyse, sondern eine direkte Übertragung der Bildinhalte beider Kameras statt. Die Darstellung auf einem geeigneten stereoskopischen Wiedergabegerät erzeugt beim Betrachter einen Tiefeneindruck durch den schon seit langer Zeit bekannten und genutzten stereoskopischen Effekt (vgl. Abb. 1.4). Die Aufnahme der Kommunikationspartner mit einer Stereo-Kamera lässt sich aber auch für die Extraktion einer 3D-Repräsentation nutzen. Diese Repräsentation lässt wiederum eine Erzeugung der virtuellen Ansicht auf den jeweiligen Kommunikationspartner zu. So ergibt sich die Möglichkeit, das Problem des mangelnden Blickkontaktes lösen. Daraus lässt sich folgende zentrale Forschungsfrage ableiten:

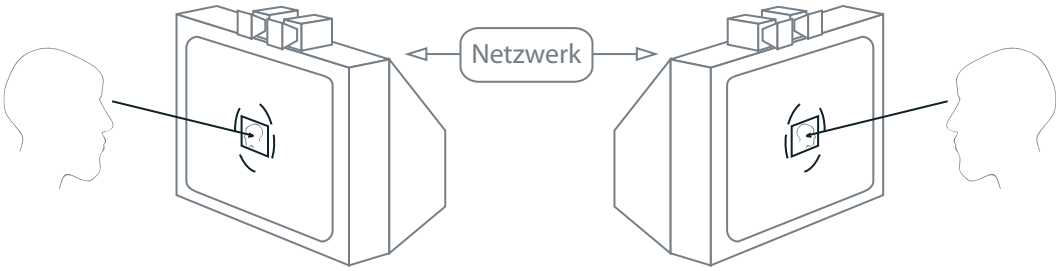



**Abbildung 1.4.:** Veranschaulichung der Stereobildübertragung in Videokommunikationsanwendungen.

**Forschungsfrage:** Welche Algorithmen für Bildanalyse und Bildsynthese sind geeignet, um eine Blickkorrektur in der Videokommunikation durch die Erzeugung virtueller Ansichten in akzeptabler Qualität zu erzeugen?

## 1.4. Aufgabenstellung und Beitrag der Arbeit

Aus der im vorangegangen Kapitel entwickelten Forschungsfrage lassen sich die allgemeinen Anforderungen an die Forschungsarbeit ableiten:

- Das zu entwickelnde Verfahren muss eine akzeptable Qualität erreichen, die identisch zu oder besser als bei bisherigen Verfahren ist.

- Das zu entwickelnde Verfahren muss aufgrund des vorgegebenen Anwendungskontextes in Echtzeit berechenbar sein. Ist dies mit aktueller Rechentechnik nicht in ausreichender Qualität möglich, so ist auf eine Skalierbarkeit der Algorithmen zu achten, so dass sie mit Hardwareentwicklungen der näheren Zukunft die Anforderung ohne Anpassung erfüllen können.

- Die Nutzbarkeit des Verfahrens ist mittels einer prototypischen Implementierung anhand verschiedener Testszenarien nachzuweisen.

Die konkreten Aufgabenbereiche lassen sich in drei Teile, die Analyse, die Synthese und die Evaluation untergliedern. In beinahe jedem dieser Bereiche leistet die Arbeit einen Beitrag, der den Stand der Technik erweitert oder in einen neuen Kontext setzt.

### 1.4.1. Stereoanalyse

In der Arbeit wird auf Basis des aktuellen Standes der Technik ein neues Verfahren zur Erzeugung einer dreidimensionalen Repräsentation aus den Bildinformationen zweier Kameras entwickelt. Als Repräsentationsform wird die Speicherung der digitalen Bildinformationen zusammen mit pixelweisen Disparitäts- bzw. Tiefenwerten gewählt. Dabei wird nicht nur der Analysealgorithmus betrachtet, sondern auch die Möglichkeit neuer vor- und nachverarbeitender Algorithmen. Priorität bei der Entwicklung haben:

- die vollständige Repräsentation der Szene hinsichtlich der Erzeugung virtueller Ansichten für die Anwendung in der Videokommunikation
- die korrekte Repräsentation der Szene
- die Parallelisierbarkeit des Algorithmus

### 1.4.2. Virtuelle Bildsynthese

Bei der Synthese von virtuellen Ansichten entstehen oft Lücken, die der Bildqualität abträglich sind. Dafür gibt es vielfältige Ursachen. Einerseits werden oft Bereiche der Szene aufgedeckt, die durch die originalen Kameras nicht erfasst wurden (*Disocclusion*). In diesen Bereichen fehlen Informationen über die Szene. Ebenso existieren Bereiche in der dreidimensionalen Repräsentation, die keine Tiefeninformationen enthalten. Weiterhin ist die Synthese als ein *Resampling* zu betrachten, welches die originalen Bildpunkte auf das Raster des virtuellen Bildes transformiert, wobei ebenfalls Lücken entstehen können. Ursache dafür sind im einfachen Fall Rundungsfehler. Ebenso entstehen durch eine im Vergleich zur originalen Kameraposition „näheren“ Position der virtuellen Ansicht zum Objekt Löcher. Die Arbeit leistet einen Beitrag in der Entwicklung von Synthese- und Füllalgorithmen zur Qualitätssteigerung der virtuellen Ansicht.

### 1.4.3. Evaluation

Die Evaluation von Algorithmen und Implementierung ist als Nachweis des Erfolges der Arbeit notwendig. Es werden etablierte Methoden der Qualitätsbewertung angewandt. Als Beitrag der Arbeit entsteht ein umfangreiches Set an Testdaten sowie die Realisierung von Software zur automatisierten Erstellung solcher Testsets. Ein in der Arbeit entwickeltes Modell qualitätsbeeinflussender technischer Parameter wird in ersten Ansätzen mit subjektiven Experimenten in Beziehung gesetzt.

# 2. Grundlagen und Stand der Technik

## 2.1. Bildentstehung

Moderne Videokommunikation setzt die Nutzung von Aufnahmegeräten voraus, die eine dreidimensionale Szenerie - im Kontext der Arbeit den Kommunikationspartner und ggf. dessen Umgebung - in ein digitales Videosignal umwandeln, welches verarbeitet und anschließend übertragen werden kann. Üblicherweise kommen dabei digitale Kameras zum Einsatz, die das von der Szene reflektierte Licht über ein optisches System auf einen Sensor projizieren und anschließend daraus das digitale Signal erzeugen. Um die dabei stattfindenden komplexen physikalischen Vorgänge zu vereinfachen und mathematisch handhabbar zu machen, haben sich im Forschungsbereich der Bildanalyse Modelle für die geometrischen Zusammenhänge zwischen den Kameras sowie die Projektion etabliert. Ebenso sind die Vorgänge und Kenntnisse bestimmter Kenngrößen während der Bildwandlung von Bedeutung. Da innerhalb der Arbeit auf diese Modelle zurückgegriffen wird und bestimmte Kenngrößen von hoher Relevanz sind, werden sie in diesem Abschnitt ausführlich vorgestellt.

### 2.1.1. Projektive Geometrie

Das am meisten vorkommende Modell der Abbildung in Kameras ist die perspektivische Projektion. Um die geometrischen Relationen zwischen dem realen dreidimensionalen Raum, den darin enthaltenen Objekte und deren Abbildung mathematisch elegant zu beschreiben, wurde das Konzept der projektiven Geometrie und des projektiven Raums erdacht. Dieser Abschnitt gibt eine kurze Einleitung notwendiger Grundlagen, die zum Verständnis der Arbeit vonnöten sind.

Im Folgenden sollen zunächst verschiedene Arten von Transformationen im zweidimensionalen Raum $\mathbb{R}^2$ (euklidische Ebene) betrachtet werden. Dies sind Transformationen, die Punkte und Geraden von einer Ebene auf eine andere Ebene abbilden. Sie spielen eine wichtige Rolle bei der Kamerakalibrierung mittels planarer Kalibrierungskörper sowie der Bestimmung der Beziehung zwischen Abbildungen einer Ebene durch verschiedene Kameras. Die folgende Klassifizierung der Transformationen lässt sich anschließend einfach auf die Transformationen im $\mathbb{R}^3$ (euklidischen Raum) generalisieren.

Im Folgenden werden homogene Koordinaten und der dreidimensionale projektive Raum $\mathbb{P}^2$ eingeführt. Es wird die formelle Definition nach Hartley und Zisserman [Hartley und Zisserman, 2008] verwendet. Eine Gerade im $\mathbb{R}^2$ wird in der allgemeinen Form durch die Gleichung $ax + by + c = 0$ und somit durch $(a, b, c)^\top$ beschrieben. Dieser Vektor repräsentiert eine Gerade bis auf einen Skalierungsfaktor, da $ax + by + c = 0$ und $(ka)x + (kb)y + kc = 0$ oder anders geschrieben $(a, b, c)^\top$ und $k(a, b, c)^\top$ für alle für $k \neq 0$ dieselben Geraden repräsentieren. Die Äquivalenzklasse zweier durch einen Skalierungsfaktor ins Verhältnis gesetzten Vektoren heißt „homogener Vektor“, wobei jeder Vektor $(a, b, c)^\top$ ein Repräsentant dieser Klasse ist. Die Menge der Äquivalenzklassen dieser Vektoren in $\mathbb{R}^3 - (0, 0, 0)^\top$ (Nullvektor ausgenommen) bildet den projektiven Raum $\mathbb{P}^2$(vgl. [Hartley und Zisserman, 2008, S. 26f]), alternativ auch projektive Ebene genannt.

Punkte, im $\mathbb{R}^2$ als Koordinatenpaar $\mathbf{x} = (x, y)^\top$ dargestellt, werden durch die Erweiterung um eine dritte Koordinaten mit dem Wert 1 in homogene Koordinaten $\mathbf{x} = (x, y, 1)^\top$ überführt. Umgekehrt wird ein Punkt im $\mathbb{P}^2$ hier allgemein geschrieben als $\mathbf{x} = (x_1, x_2, x_3)^\top$ in $\mathbb{R}^2$ überführt, indem seine Koordinaten durch $x_3$ geteilt werden: $\mathbf{x} = (x_1/x_3, x_2/x_3)^\top$. Es wird ersichtlich, das ein Punkt im $\mathbb{P}^2$ als Strahl durch den Ursprung repräsentiert wird, wodurch der Punkt bis auf einen Skalierungsfaktor ($x_3$) definiert ist. Für $x_3 = 0$ kann wegen der Division durch 0 kein Punkt im $\mathbb{R}^2$ bestimmt werden. Diese Punkte werden als *points at infinity* oder „ideale Punkte“ bezeichnet und stellen einen Unterraum im Unendlichen dar (vgl. [Schreer u. a., 2005][Hartley und Zisserman, 2008]).

Aus den vorangegangen Definitionen lassen sich verschiedene Eigenschaften für Punkte und Geraden im $\mathbb{P}^2$ ableiten. Für die Herleitung sei auf [Hartley und Zisserman, 2008, S. 26ff] verwiesen.

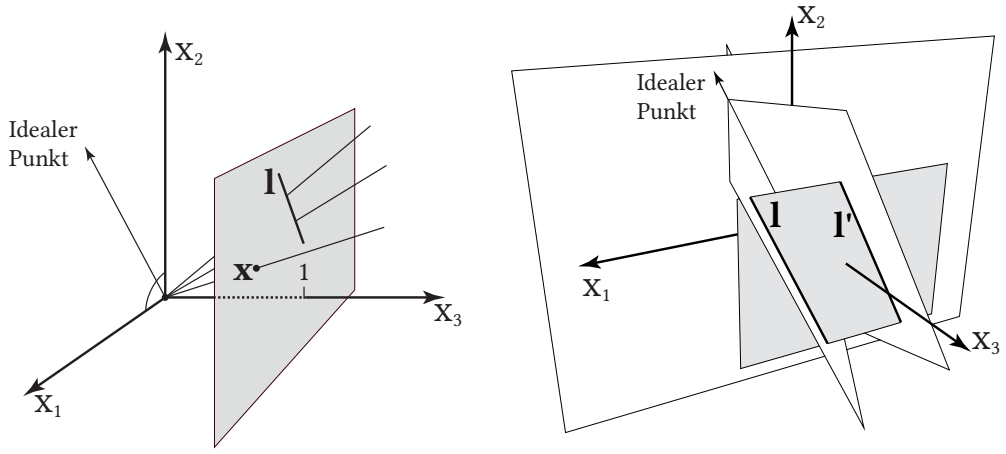
**Punkt auf einer Geraden:** Ein Punkt liegt auf einer Geraden, wenn gilt $\mathbf{x}^\top\mathbf{l} = 0$, wobei $\mathbf{l} = (a, b, c)^\top$ die Gerade darstellt.

**Schnittpunkt zwischen Geraden:** Der Schnittpunkt zweier Geraden ist bestimmt durch $\mathbf{x} = \mathbf{l} \times \mathbf{l}'$.

**Gerade zwischen zwei Punkten:** Eine Gerade durch zwei Punkte $\mathbf{x}$ und $\mathbf{x}'$ ist bestimmt durch $\mathbf{l} = \mathbf{x} \times \mathbf{x}'$.

Demnach sind Punkte und Gerade im $\mathbb{P}^2$ „austauschbar“, was als Dualität bezeichnet wird.

Die Definitionen der vorherigen Abschnitte sind für den dreidimensionalen projektiven Raum $\mathbb{P}^2$ in Abb. 2.1 veranschaulicht. Die idealen Punkte sind Vektoren der Ebene $x_1, x_2$. Diese Ebene repräsentiert eine einzelne Gerade, die als *line at infinity* - $\mathbf{l}_\infty$ bezeichnet wird. Eine Gerade schneidet diese *line at infinity* in einem idealen Punkt. Aus Abbildung 2.1 wird ersichtlich, dass




**(a)** Nach [Hartley und Zisserman, 2008] **(b)**

**Abbildung 2.1.:** Visualisierung des projektiven Raums $\mathbb{P}^2$ (projektive Ebene). Punkte werden als Strahlen, Geraden als Ebenen jeweils durch den Nullpunkt repräsentiert.(a) Der Schnittpunkt dieser Strahlen und Ebenen mit der Ebene $x3 = 1$ ergibt Punkt bzw. Ebene im $\mathbb{R}^2$. Die $\mathbf{l}$ repräsentierende Ebene schneidet die *line at infinity* in einem idealen Punkt. (b) Veranschaulichung des Schnittes zweier paralleler Geraden im $\mathbb{P}^2$
.

zwei parallele Geraden $\mathbf{l}$ und $\mathbf{l}'$ sich in der projektiven Ebene schneiden und zwar in einem idealen Punkt.

Die o. g. Definitionen werden auf den dreidimensionalen Fall erweitern. Ein Punkt im $\mathbb{R}^3$ wird durch einen homogenen Vektor mit vier Elementen im $\mathbb{P}^3$ beschrieben. Punkt $\mathbf{X} = (X_1, X_2, X_3, X_4)^\top$ mit $X_4 \neq 0$ entspricht dabei dem Punkt $(X, Y, Z)^\top$ im $\mathbb{R}^3$, wobei $X = X_1/X_4, Y = X_2/X_4$ und $Z = X_3/X_4$ sind. Ebenen werden im $\mathbb{R}^3$ durch die vier Parameter der allgemeinen Ebenengleichung repräsentieren: $\pi_1 X + \pi_2 Y + \pi_3 Z + \pi_4 = 0$ wobei der homogene Ebenenvektor $\pi = (\pi_1, \pi_2, \pi_3, \pi_4)^\top$ ist. Die Dualität zwischen Geraden und Punkten in der projektiven Ebene $\mathbb{P}^2$ ist im projektiven Raum $\mathbb{P}^3$ zwischen Ebenen und Punkten gegeben. Der *line at infinity* entspricht die *plane at infinity*.

Geraden haben im $\mathbb{R}^3$ vier Freiheitsgrade. Die Darstellung der Geraden als homogener Vektor entspräche einem fünf-Element-Vektor. Dieser lässt sich mathematisch schlecht mit der homogenen Repräsentation von Punkten und Ebenen kombinieren. Daher wurden verschiedene Formen der Repräsentation erdacht, deren Erklärung an dieser Stelle jedoch zu weit führen würden. Es sei auf [Hartley und Zisserman, 2008, S. 69ff] verwiesen.

### 2.1.2. Transformationen im projektiven Raum

Mittels des im vorangegangen Kapitel eingeführten Konzeptes homogener Koordinaten lassen sich Transformationen von einer Ebene auf eine andere als lineare Abbildungen innerhalb des $\mathbb{P}^2$ darstellen. Formal ist nach [Hartley und Zisserman, 2008] eine projektive Transformation eine „invertierbare Abbildung $h$ von Punkten von $\mathbb{P}^2 \mapsto \mathbb{P}^2$, so dass drei Punkte $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3$ auf derselben Geraden liegen genau dann, wenn $h(\mathbf{x}_1)$, $h(\mathbf{x}_2)$, $h(\mathbf{x}_3)$ auf einer Geraden liegen“. Diese lineare Abbildung wird mittels einer nichtsingulären $3 \times 3$ Matrix $\mathtt{H}$ geschrieben als:

$$\begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \tag{2.1}$$

oder kurz $\mathbf{x}' = \mathtt{H}\mathbf{x}$. Diese projektive Transformation, auch *Homographie* genannt, lässt sich in Gruppen verschiedener Transformationen unterteilen. Sie unterscheiden sich dabei in den Freiheitsgraden der Transformationsmatrizen und somit den möglichen Arten der Transformation. Die Freiheitsgrade bestimmen u. a., wie viele Punktkorrespondenzen auf zwei Ebenen notwendig sind, um die Abbildung zu bestimmen. An dieser Stelle wird kurz die Hierarchie für den $\mathbb{P}^2$ erläutert. Anschließend erfolgt eine ergänzende Erweiterung auf den $\mathbb{P}^3$.

**Euklidische Transformation / Isometrie:** Eine Transformation mit drei Freiheitsgraden. Es können Rotationen und Translation abgebildet werden. Die Transformationsmatrix lautet:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} \epsilon\ \cos\Theta & -\sin\Theta & t_x \\ \epsilon\ \sin\Theta & \cos\Theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{2.2}$$

mit $\epsilon = \pm 1$. Kompakt geschrieben:

$$\mathbf{x}' = \mathtt{H}_\mathtt{E}\mathbf{x} = \begin{bmatrix} \mathtt{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix} \mathbf{x} \tag{2.3}$$

$\mathtt{R}$ ist die Rotationsmatrix und $\mathbf{t}$ der Translationsvektor. Die wichtigsten Invarianten dieser Transformation sind Längen, Winkel zwischen Geraden sowie Größe von Flächen.

**Ähnlichkeitstransformation:** Die Erweiterung der euklidische Transformation um die Möglichkeit der Skalierung ergibt:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} s\ \cos\Theta & -s\ \sin\Theta & t_x \\ s\ \sin\Theta & s\ \cos\Theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{2.4}$$

mit $s$ als isotropischem Skalierungsfaktor. Kompakt geschrieben:

$$\mathbf{x}' = \mathtt{H}_\mathtt{S}\mathbf{x} = \begin{bmatrix} s\mathtt{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix} \mathbf{x} \tag{2.5}$$

mit vier Freiheitsgraden. Invarianten der Ähnlichkeitstransformation sind z. B. Winkel und Parallelität zwischen Geraden oder das Verhältnis von Längen, jedoch *nicht* die Längen selbst.

**Affine Transformation:** Die affine Transformation repräsentiert eine nichtsinguläre lineare Transformation $A$ mit anschließender Translation.

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{2.6}$$

Kompakt geschrieben:

$$\mathbf{x}' = \mathtt{H}_\mathtt{A}\mathbf{x} = \begin{bmatrix} \mathtt{A} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix} \mathbf{x} \tag{2.7}$$

Die Transformation `A` ist vorstellbar als eine Kombination aus einer Rotation und anschließender Rotation, Skalierung, Rückrotation dargestellt wird:

$$\mathtt{A} = \mathtt{R}(\theta)\mathtt{R}(-\phi)\mathtt{D}\mathtt{R}(\phi) \tag{2.8}$$

mit der Diagonalmatrix `D`

$$\mathtt{D} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \tag{2.9}$$

Die Freiheitsgrade erhöhen sich durch $\lambda_1$ und $\lambda_2$ auf sechs. Invarianten sind parallele Geraden, das Verhältnis von Flächen oder das Verhältnis der Länge paralleler Geradensegmente.

**Projektive Transformation:** Wird die Transformationsmatrix der affinen Transformation auf die eingangs erwähnte allgemeine Matrix `H` erweitert, erhält man die projektive Transformation. Sie besitzt acht Freiheitsgrade durch die Einführung des Vektors $\mathbf{v} = (v_1, v_2)^\top$:

$$\mathbf{x}' = \mathtt{H}_\mathtt{P}\mathbf{x} = \begin{bmatrix} \mathtt{A} & \mathbf{t} \\ \mathbf{v}^\top & v \end{bmatrix} \mathbf{x}' \tag{2.10}$$

Während die affine Transformation die Komposition einer nichtsingulären (regulären) Transformation inhomogener Koordinaten und einer Translation ist, ist die projektive Transformation eine generelle nichtsinguläre lineare Transformation homogener Koordinaten. Invarianten sind Geradlinigkeit (Geraden bleiben Geraden) und das Kreuzverhältnis (Verhältnis des Verhältnisses)

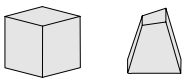
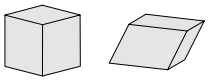
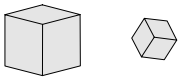
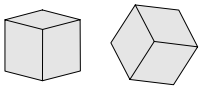
| Transformationstyp | Matrix | Visualisierung |
|---|---|---|
| Projektiv<br>15 Freiheitsgrade | $\begin{bmatrix} \mathtt{A} & \mathbf{t} \\ \mathbf{v}^\top & v \end{bmatrix}$ |  |
| Affin<br>12 Freiheitsgrade | $\begin{bmatrix} \mathtt{A} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix}$ |  |
| Ähnlichkeit<br>7 Freiheitsgrade | $\begin{bmatrix} s\mathtt{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix}$ |  |
| Euklidisch<br>6 Freiheitsgrade | $\begin{bmatrix} \mathtt{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix}$ |  |

**Tabelle 2.1.:** Transformationshierarchie im $\mathbb{P}^3$. Erklärungen siehe Text. Nach [Hartley und Zisserman, 2008].

von Längen auf einer Geraden.

Transformationen in der projektiven Ebene finden bei der Analyse und Korrektur von perspektivischen Abbildungen Anwendung. So lassen sich perspektivische Verzerrungen eines Bildes kompensieren. Die Nutzung von Punkten und Geraden auf einer Ebenen im Bild sowie die Kenntnis über deren Lagebeziehungen (rechte Winkel, Linien, Längenverhältnisse) können zur Ermittlung der Transformationsmatrix genutzt werden. Ebenso können die zwei Abbildungen einer Ebene durch eine Kamera mittels einer Homographie beschrieben werden. Die Ermittlung der Homographie erfolgt dabei über Punktkorrespondenzen in beiden Bildern. Da die Messung der Korrespondenzen üblicherweise fehlerbehaftet ist, werden oft iterative Lösungsverfahren zur Minimierung des Fehlers zur Bestimmung eingesetzt.

Die vorgestellte Transformationshierarchie wird nun auf den $\mathbb{R}^3$ und somit auf lineare Abbildungen im $\mathbb{P}^3$ erweitern. Die allgemeine projektive Transformationsmatrix ist demnach eine $4 \times 4$ Matrix. Die Schreibweise wird entsprechend den vereinfachten Schreibweisen im $\mathbb{P}^2$ übernommen. Somit besitzt eine euklidische Transformation im $\mathbb{P}^3$ sechs Freiheisgrade - die drei Winkel der Rotationsmatrix und drei Transformationsrichtungen. Die Ähnlicheitstransformation erweitert die Freiheitsgrade durch den Skalierungsfaktor $s$ auf sieben, die affine Transformation durch die nichtsinguläre $3 \times 3$ Matrix $\mathtt{A}$ und die drei Translationsrichtungen auf 12 und die projektive Transformation auf 15 Freiheitsgrade. Tabelle 2.1 gibt einen anschaulichen Überblick der Hierarchie.

Die Transformationsmatrizen im $\mathbb{P}^3$ werden häufig in der Computergrafik verwandt. So bieten die meisten 3D-Programme zumindest Ähnlichkeitstransformationen bis hin zu affinen Transfor-

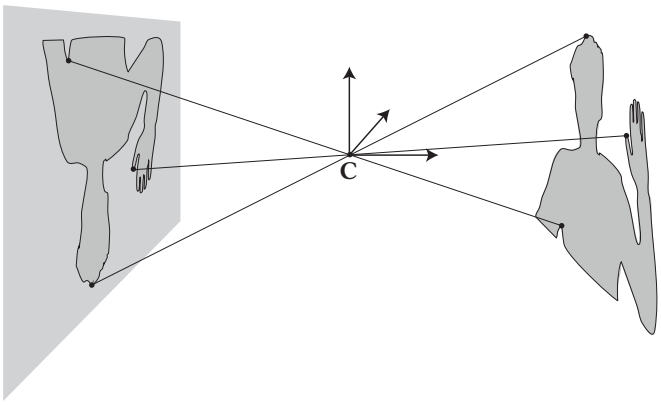



**Abbildung 2.2.:** Grundprinzip der Abbildung eines Objektes auf die Bildebene einer Lochkamera mit dem Kamerazentrum $\mathbf{C}$. Zur Veranschaulichung sind die Größenverhältnisse stark verzerrt.

mationen an. In dieser Arbeit finden sie u. a. Verwendung in der Beschreibung der Abbildung von Punkten im $\mathbb{R}^3$ auf die Bildebenen oder in allgemeinen euklidischen Kameratransformationen im Raum.

### 2.1.3. Lochkameramodell

Das Lochkameramodell ist das einfachste Modell der Abbildung des euklidischen Raums auf ein 2D-Bild ($\mathbb{R}^3 \mapsto \mathbb{R}^2$). Es reicht – meist in Kombination mit einen Linsenverzerrungsmodell (vgl. 2.1.4) – für viele Anwendungen im Bildverarbeitungsbereich aus. Mit seiner Hilfe wird eine perspektivische 3D-Projektion oder auch „Zentralprojekt“ mathematisch beschrieben. Abbildung 2.2 veranschaulicht das Prinzip. Von einem Objekt emittierte bzw. reflektierte Lichtstrahlen werden durch ein optisches Zentrum, den Brennpunkt, punktgespiegelt auf eine Bildebene (den Sensor der Kamera) projiziert. Die Ebene, in der sich der Brennpunkt befindet, wird als Brennebene und deren Abstand zur Bildebene als Brennweite $f$ bezeichnet [Kreibich, 2005] [Schreer u. a., 2005]. Die mathematischen Verhältnisse bei der Abbildung eine Punktes $\mathbf{X}$ des euklidischen Raums $\mathbb{R}^3$ auf eine Bildebene veranschaulicht Abbildung 2.3. Aus dem Satz ähnlicher Dreiecke ergeben sich die Koordinaten des abgebildeten Punktes zu $(fX/Z, fY/Z, f)^\top$. Unter der Verwendung der zuvor eingeführten homogenen Koordinaten wird der Zusammenhang geschrieben als:

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \longmapsto \begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} = \begin{bmatrix} f & & & 0 \\ & f & & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \tag{2.11}$$

oder allgemein:

$$\mathbf{x} = \mathrm{P}\mathbf{X} \tag{2.12}$$

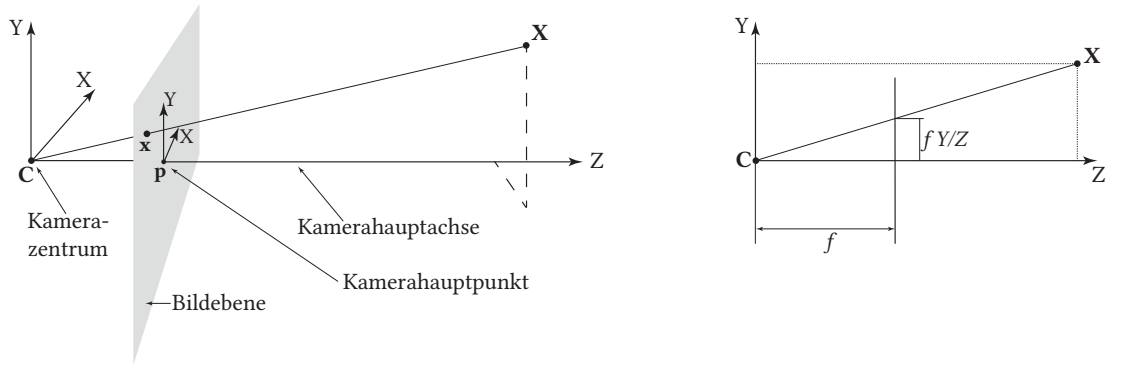



**Abbildung 2.3.:** Abbildung eines Punktes aus dem euklidischen Raum $\mathbb{R}^3$ auf die $\mathbb{R}^2$-Bildebene einer Lochkamera. Zur besseren Darstellung wird die Bildebene entgegen den realen Verhältnissen vor der Kamera platziert. Nach [Hartley und Zisserman, 2008]

mit

$$\mathtt{P} = diag(f, f, 1)[\mathtt{I}|0] \tag{2.13}$$

P ist die $3 \times 4$ homogene Kameraprojektionsmatrix [Hartley und Zisserman, 2008]. Bisher wurde angenommen, dass der Koordinatenursprung der (durch die Sensorgröße physisch begrenzten) Bildebene der Schnittpunkt der Kamerahauptachse mit der Bildebene ist, als *Kamerahauptpunkt* bezeichnet. Dies ist unter praktischen Verhältnissen meist nicht gegeben, so dass die Kameraprojektionsmatrix durch eine Verschiebung der Koordinaten auf der Bildebene in $X$- und $Y$-Richtung ergänzt wird. Wird das Koordinatensystem der Bildebene rechtshändig entsprechend Abb. 2.4a definiert, so ergeben sich die Bildkoordinaten des Bildpunktes indem zu dessen Kamerakoordinaten die Bildkoordinaten des Kamerahauptpunktes $\mathbf{p} = (p_x, p_y)^\top$ addiert werden. Dies ist auch die übliche Schreibweise innerhalb der homogenen Projektionsmatrix. Die Abbildung wird somit zu

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \longmapsto \begin{pmatrix} fX + Zp_x \\ fY + Zp_y \\ Z \end{pmatrix} = \begin{bmatrix} f & & p_x & 0 \\ & f & p_y & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \tag{2.14}$$

In der Praxis ebenso wie in dieser Arbeit wird das Koordinatensystem von Bildern meist linkshändig und somit in der linken oberen Ecke definiert (vgl. 2.4b). Dies bedeutet, dass vor der Anwendung der Translation um $p_x$ die y-Koordinate des abgebildeten Punktes im Kamerakoordinatensystem negiert werden muss ($y = -fY + Zp_y$). Dies wird in der Arbeit stets bedacht, wenn in den Matrizen auch nicht explizit modelliert. Die linke, obere $3 \times 3$ Matrix wird nach

dieser Erweiterung *Kamerakalibrierungsmatrix* $\mathtt{K}$ genannt und beinhaltet die internen Parameter einer Lochkamera, namentlich Brennweite $f$ und Kamerahauptpunkt $(p_x, p_y)^\top$.

$$\mathtt{K} = \begin{bmatrix} f & & p_x \\ & f & p_y \\ & & 1 \end{bmatrix} \tag{2.15}$$

Um weitere Eigenschaften realer Kameras zu modellieren, wird das Modell um zusätzliche Parameter erweitert. Bildkoordinaten werden in der Quantität $\mathtt{Pixel}$[1] oder $\mathtt{px}$ angegeben (zur genauen Definition vgl. Abschnitt 2.2.1). Bei CCD- oder CMOS-Sensoren kann es vorkommen, dass die Sensorelemente/Pixel nicht quadratisch sind und sich somit eine unterschiedliche Skalierung in $x$- und $y$-Richtung ergibt. Dem wird Rechnung getragen, indem für Umrechnung und Skalierung die Faktoren $m_x$ und $m_y$ eingeführt werden. Mit $\alpha_x = fm_x$ und $\alpha_y = fm_y$ ergibt sich damit

$$\mathtt{K} = \begin{bmatrix} \alpha_x & & p_x \\ & \alpha_y & p_y \\ & & 1 \end{bmatrix} \tag{2.16}$$

Mitunter sind auch die Bezeichnungen $f_x$ und $f_y$ zu finden, die meist ebenso die Brennweite in $\mathtt{Pixel}$ für die jeweilige Dimension enthalten. Um einen Sensor zu modellieren, der nicht senkrecht zur optischen Achse der Kamera liegt, wird $\mathtt{K}$ um den Parameter $s$ erweitert. Da dies in modernen Kameras selten der Fall ist, wird $s$ meist zu $0$ gesetzt [Szeliski, 2011]. Die vollständige Kamerakalibrierungsmatrix mit insgesamt 11 Freiheitsgraden ergibt sich zu:

$$\mathtt{K} = \begin{bmatrix} \alpha_x & s & p_x \\ & \alpha_y & p_y \\ & & 1 \end{bmatrix} \tag{2.17}$$

Die Koordinaten des Punktes $\mathbf{X}$ wurden bisher im Kamerakoordinatensystem angegeben, dessen Ursprung das Kamerazentrum ist. Zur Beschreibung der Lagebeziehungen zwischen mehreren Kameras, wird ein *Weltkoordinatensystem* eingeführt. Um die in Weltkoordinaten angegeben Punkte in Kamerakoordinaten zu überführen, bedarf es einer euklidischen Transformation die mit inhomogenen Koordinaten folgendermaßen ausgedrückt wird $\mathbf{X}_{kam} = \mathtt{R}(\mathbf{X} - \mathbf{C})$. $\mathtt{R}$ ist die Orientierung des Kamerakoordinatensystems und $\mathbf{C}$ die Position des Kamerazentrums in Weltkoordinaten. Im Folgenden wird die üblichere Schreibweise mittels eines Translationsvektors $\mathbf{t}$ als $\mathbf{X}_{kam} = \mathtt{R}\mathbf{X} + \mathbf{t}$ mit $\mathbf{t} = -\mathtt{R}\mathbf{C}$ verwendet. $\mathtt{R}$ und $\mathbf{t}$ werden als externe Kameraparameter bezeichnet. Die Zusammenfassung der internen Parameter repräsentiert durch die Matrix $\mathtt{K}$ und der externe Parameter in einer allgemeinen homogenen $3 \times 4$ Projektionsmatrix ergibt

$$\mathtt{P} = \mathtt{K}[\mathtt{R}|\mathbf{t}] \tag{2.18}$$

[1] aus dem Englischen abgeleitetes Kunstwort zusammengesetzt aus *Picture Element*

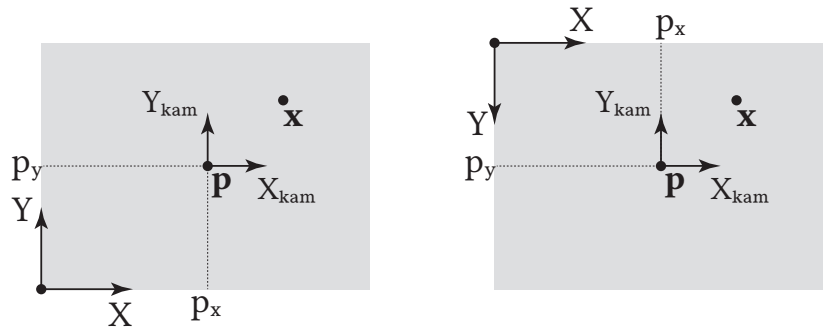



**Abbildung 2.4.:** Varianten von Koordinatensystemen der Bildebene und deren Bezug zum Kamerakoordinatensystem. Nach [Hartley und Zisserman, 2008]

Abbildung 2.5 veranschaulicht den Zusammenhang zwischen den Koordinatensystemen. Mittels der Matrix $\mathtt{P}$ lässt sich nun ein Raumpunkt $\mathbf{X}$ mit homogenen Koordinaten in den Bildpunkt $\mathbf{x}$ ebenfalls in homogenen Koordinaten transformieren. Dieser ist bis auf den Skalierungsfaktor definiert, der sich aus der projektiven Transformation ergibt, und wird im $\mathbb{P}^3$ als Strahl ausgehend vom Kamerazentrum repräsentiert:

$$\mathbf{x} = \begin{pmatrix} U \\ V \\ S \end{pmatrix} = \mathtt{P} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \mathtt{P}\mathbf{X} \tag{2.19}$$

Um die Position des Punktes auf der Ebene im $\mathbb{R}^2$ zu erhalten, muss dessen letzte Koordinate zu $1$ werden, was einer Division durch $Z$ gleich kommt, auch als perspektivische Division bezeichnet. Der Skalierungsfaktor wird somit eliminiert:

$$\mathbf{x} = \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} U/S \\ V/S \\ 1 \end{pmatrix} \tag{2.20}$$

### 2.1.4. Linsenverzerrungen

Die im vorherigen Abschnitt beschriebe *lineare Abbildung* durch die Projektionsmatrix $\mathtt{P}$ bildet reale Kameras oft nicht ausreichend nach. Insbesondere (billige) Optiken führen zu nichtlinearen

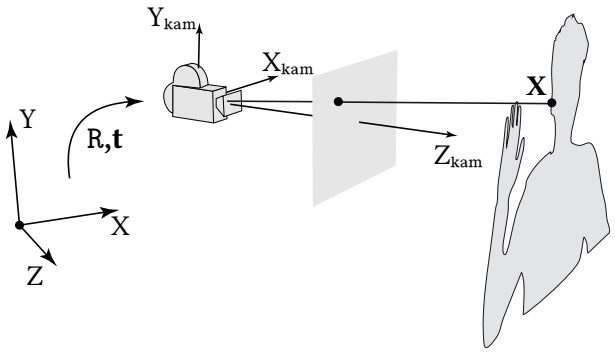



**Abbildung 2.5.:** Visualisierung der externen Kameraparameter. $\mathtt{R}$ und $\mathbf{t}$ überführen Punkt $\mathbf{X}$ vom Welt- in das Kamerakoordinatensystem.

Verzerrungen im Bild. Werden diese z. B. bei der Kamerakalibrierung im Modell nicht integriert, so kann dies zu Fehlern bei der Rektifizierung von Bildern und somit bei der Disparitätsanalyse (vgl. Abschnitte 5.1.2 und 5.2) führen. In dieser Arbeit wird für die Kalibrierung der verwendeten Kameras die „Camera Calibration Toolbox for Matlab“ eingesetzt [Bouguet, 2010]. Die oben vorgestellte Eliminierung des Skalierungsfaktors kann bereits vor der Anwendung der internen Kameramatrix, aber nach der externen Transformation durchgeführt werden. Es liegen dann die Bildkoordinaten im Kamerakoordinatensystem vor, auch *normalisierte Koordinaten* $\mathbf{x}_n$ genannt (vgl. [Hartley und Zisserman, 2008, S. 257]).

$$\mathbf{x}_n = \begin{pmatrix} X_{kam}/Z_{kam} \\ Y_{kam}/Z_{kam} \end{pmatrix} = \begin{pmatrix} x_n \\ y_n \end{pmatrix} \tag{2.21}$$

Auf diese Koordinaten wird ein Verzerrungsmodell angewandt, welches radiale und tangentiale Verzerrungen durch einen Vektor $\mathbf{K} = (K_1, K_2, K_3, K_4, K_5)^\top$ parametrisiert, der die Wichtungskoeffizienten eines Polynoms sechsten Grades enthält. Mit der Kreisgleichung $r^2 = x_n^2 + y_n^2$ ergeben sich die verzerrten Koordinaten zu:

$$\mathbf{x}_d = \begin{pmatrix} x_d \\ y_d \end{pmatrix} = (1 + K_1 r^2 + K_2 r^4 + K_5 r^6)\mathbf{x}_n + \mathbf{d}_x \tag{2.22}$$

Der Vektor $\mathbf{d}_x$ ist der tangentiale Verzerrungsvektor und folgendermaßen definiert:

$$\mathbf{d}_x = \begin{pmatrix} 2K_3 x_n y_n + K_4(r^2 + 2x_n^2) \\ K_3(r^2 + 2y_n^2) + 2K_4 x_n y_n \end{pmatrix} \tag{2.23}$$

Wird $\mathbf{d}_x$ wieder in homogenen Koordinaten geschrieben, kann nun die $3 \times 3$ Kalibrierungsmatrix $\mathtt{K}$ auf die verzerrten Bildkoordinaten angewandt werden:

$$\mathbf{x}_p = \begin{pmatrix} x_p \\ y_p \\ 1 \end{pmatrix} = \mathtt{K} \begin{pmatrix} x_d \\ y_d \\ 1 \end{pmatrix} \tag{2.24}$$

Das Modell wurden 1966 von Brown eingeführt [Brown, 1966]. Es bildet die meisten vorkommenden Verzerrungen realer Kameras sehr gut ab. Positive und negative Implikationen der Verwendung dieses nichtlinearen Kameramodells innerhalb der Arbeit werden an entsprechender Stelle diskutiert. Die in der Toolbox und somit in der Arbeit verwendeten Verfahren zur Bestimmung der Elemente von $\mathbf{K}$ werden hier nicht näher erläutert. Für weiter gehende Informationen sei an dieser Stelle auf [Tsai, 1987; Heikkila und Silven, 1997; Zhang, 2000] verwiesen.

### 2.1.5. Stereogeometrie

In diesem Abschnitt wird auf die geometrischen Zusammenhänge eingegangen, die sich aus der Nutzung von zwei Kameras ergeben. Grundsätzlich lässt sich die Anordnung von zwei Kameras in zwei Kategorien unterteilen – achsenparallele und konvergente Anordnungen. Bei der erstgenannten sind die Kamerahauptachsen parallel zueinander. D. h. die Beziehung der Kamerazentren besteht allein in einer Translation. Oft wird diese zusätzlich auf eine Translation in nur eine Richtung beschränkt: horizontal entlang der $X$-Achse oder vertikal entlang der $Y$-Achse. In konvergenten oder auch „allgemeinen“ Kameraanordnungen stehen die Kamerazentren durch eine euklidische Transformation (vgl. Abschnitt 2.1.2) mit Rotations- und Translationskomponenten in Beziehung. Über das für und wider der einzelnen Anordnungen für den konkreten Anwendungsfall der Arbeit wird an geeigneter Stelle diskutiert. Abbildung 2.6 veranschaulicht beide Prinzipien.

**Achsenparallele Stereogeometrie:** Wird der achsenparallele Fall geometrisch betrachtet, so ergeben sich wichtige Maße, die für die spätere Stereoanalyse von Bedeutung sind. Ausgehend von der Annahme, dass eine Translation nur in $X$-Richtung erfolgt, lässt sich der Zusammenhang zwischen Brennweite $f$, Basisabstand $b$, Tiefe $Z$ eines Raumpunktes $\mathbf{X}$ und der Position dessen Abbildungen $\mathbf{x}$ und $\mathbf{x}'$ einfach herleiten (vgl. Abb. 2.7). Die Koordinaten der Bildpunkte nach der Anwendung der internen Kameramatrix im $\mathbb{R}^2$ sind $\mathbf{x} = (x, y)^\top$ und $\mathbf{x}' = (x', y)^\top$. Die *Disparität* ergibt sich in dieser Anordnung zu

$$d = x - x' \tag{2.25}$$

in $\mathtt{px}$ angegeben[2]. Unter der Annahme identischer Sensorgrößen und -lagen ergibt sich aus dem

[2] Es sein darauf hingewiesen, dass $d$ in dieser Notation auch nicht-ganzzahlig sein kann.

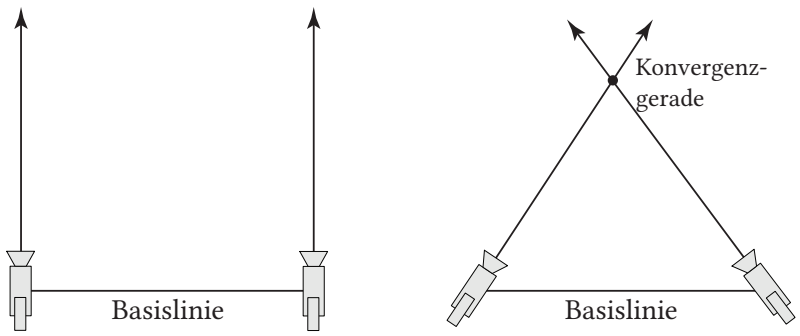



**Abbildung 2.6.:** Grundsätzliche Formen der Kameraanordnung. (a) Achsenparallele Anordnung. (b) Konvergente Anordnung. Die Konvergenzgerade ist die Schnittgerade der jeweiligen Ebenen aufgespannt aus Kamerahauptachse $Z$ und $Y$-Achse. Oft wird auch von einem Konvergenzpunkt geredet, in dem sich beide Hauptachsen schneiden, was jedoch mit realistischen Kameraaufbauten nur annähernd zu realisieren ist.

Strahlensatz das Verhältnis

$$\frac{Z}{f} = \frac{b}{d\,\delta_x} \tag{2.26}$$

wobei mit $\delta_x$ die Breite eines Sensorelementes in $X$-Richtung darstellt und somit die Umrechnung von Pixeln in das Maß des Kamerakoordinatensystems erfolgt (vgl. auch Gleichung 2.16, Seite 17 wonach sich $\delta_x = 1/m_x$ ergibt). Aus der Gleichung lässt sich ein direkter Zusammenhang zwischen der Tiefe $Z$ eines Raumpunktes $\mathbf{X} = (X, Y, Z)^\top$ und der Disparität herstellen:

$$Z = \frac{bf}{\delta_x} \cdot \frac{1}{d} \tag{2.27}$$

Es ist offensichtlich, dass sich die Tiefe umgekehrt proportional zur Disparität verhält und demnach ein nichtlinearer Zusammenhang besteht. Werden für alle Punkte einer Stereoabbildung jeweils die Disparitäten ermittelt (*dichte Disparitätskarte*), so lässt sich der Bereich der abgebildeten Szene rekonstruieren. Es sei noch einmal darauf hingewiesen, dass dieser Zusammenhang nur für die genannte Translation sowie identische Brennweiten $f$ gilt (vgl. auch [Schreer u. a., 2005, S. 67ff]). Die Anwendung auf eine vertikale Kameraanordnung (Translation nur in $Y$-Richtung) ist ohne Einschränkungen möglich.

**Allgemeine Stereogeometrie:** Beinhaltet die euklidische Transformation zwischen den beiden Kameras neben der Translation auch eine Rotation, wird dies „allgemeine Anordnung“ genannt. Die geometrischen Zusammenhängen – als *Epipolargeometrie* bezeichnet – verhalten sich wie in Abbildung 2.8 dargestellt. Die Disparität hat eine horizontale und eine vertikale Komponente.

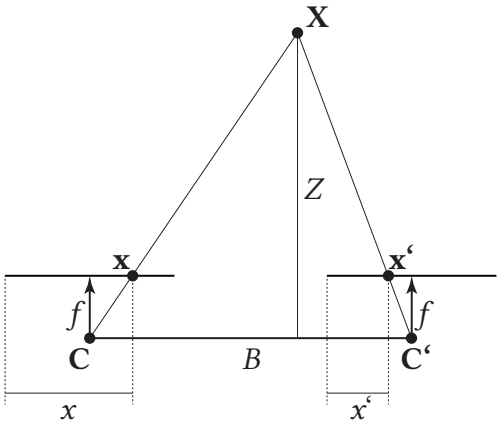



**Abbildung 2.7.:** Achsenparallele Kameraanordnung. Aufsicht zur Visualisierung der geometrischen Verhältnisse zwischen Brennweite $f$, Basisabstand $B$, Tiefe $Z$ eines Raumpunktes $\mathbf{X}$ und den Positionen von dessen Abbildungen $\mathbf{x}$ und $\mathbf{x}'$.

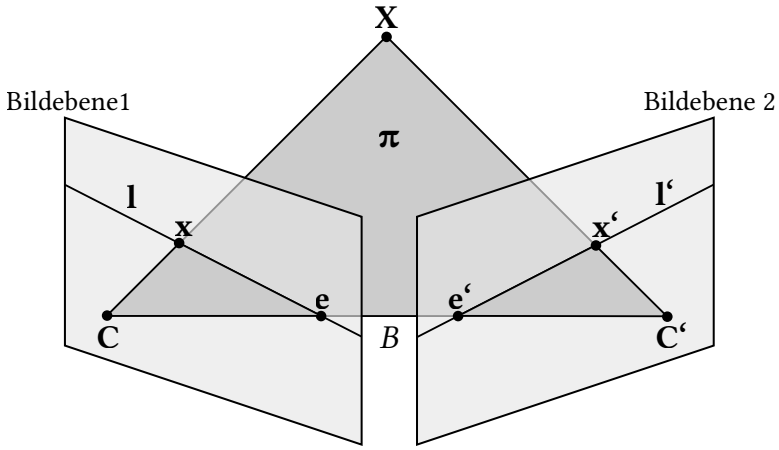



**Abbildung 2.8.:** Die Epipolargeometrie zwischen zwei Kameras

Der Raumpunkt $\mathbf{X}$ bildet mit den beiden Kamerazentren $\mathbf{C}$ und $\mathbf{C}'$ die *Epipolarebene* $\boldsymbol{\pi}$. Sie schneidet die beiden Bildebenen in den *Epipolarlinien*[3] $\mathbf{l} = (a, b, c)^\top$ und $\mathbf{l}'$, auf denen auch die Abbildungen von $\mathbf{X}$ – $\mathbf{x}$ und $\mathbf{x}'$ – liegen. Für jeden Raumpunkt, der nicht auf der Basislinie liegt, existiert eine Epipolarebene und somit zwei Epipolarlinien. Durch die konvergente Anordnung schneidet die Basislinie die Bildebenen in den Punkten $\mathbf{e}$ und $\mathbf{e}'$, den *Epipolen*. Diese müssen nicht gezwungenermaßen auf der aktiven Sensorfläche liegen, können sich also auch außerhalb des Bildes befinden. Alle Epipolarlinien eines Bildes schneiden sich im jeweiligen Epipol. Die Menge wird Epipolarlinienbüschel genannt (vgl. [Schreer u. a., 2005]). Ebenso ist leicht zu erkennen, dass alle Epipolarebenen um die Basislinie „rotieren“. Eine Epipolarlinie $\mathbf{l}'$ kann als Projektion

[3] Die korrekte Bezeichnung wäre eigentlich „Epipolargeraden“. Aufgrund des englischen „*epipolar lines*“ hat sich jedoch „-linien“ etabliert. Ebenso wird im Folgenden der Begriff Basis*linie* verwendet.

des Sichtstrahls von einem Raumpunkt $\mathbf{X}$ zu seiner Abbildung $\mathbf{x}$ auf die Bildebene von Kamera $\mathbf{C}'$ interpretiert werden. Ist nur $\mathbf{x}$ gegeben und wird seine korrespondierende Abbildung $\mathbf{x}'$ gesucht, so kann die Suche durch die Epipolargeometrie auf $\mathbf{l}'$ beschränkt werden, was für die Stereoanalyse von entscheidender Bedeutung ist, da so der Suchraum eingeschränkt wird. Mathematisch formuliert existiert eine Abbildung

$$\mathbf{x} \mapsto \mathbf{l}' \tag{2.28}$$

Diese Abbildung wird durch die Fundamentalmatrix $\mathtt{F}$ repräsentiert, unter der die Beziehung

$$\mathbf{l}' = \mathtt{F}\mathbf{x} \text{ und } \mathbf{l} = \mathtt{F}^\top \mathbf{x}' \tag{2.29}$$

gilt. $\mathtt{F}$ ist eine Matrix vom Rang 2. Sie lässt sich geometrisch aus einer projektiven Transformation von Bildebene 1 in die Bildebene 2 über eine Szenenebene, die den Raumpunkt $\mathbf{X}$ und den Schnittpunkt der Epipolarlinien mit dem Epipol verbindet, herleiten. Details dazu finden sich in [Hartley und Zisserman, 2008, S. 243ff]. Die Abbildung des Punktes $\mathbf{x}$ auf die Gerade $\mathbf{l}'$ ist eine projektive Abbildung $\mathbf{x} \longmapsto \mathbf{l}'$. Die Fundamentalmatrix ist eine homogene Matrix. Die Abbildung ist jedoch nicht invertierbar, da $\mathtt{F}$ nicht von vollem Rang ist. Einer der wichtigsten Zusammenhänge zwischen Fundamentalmatrix und Punktkorrespondenzen $\mathbf{x} \longleftrightarrow \mathbf{x}'$ in allgemeiner Stereoanordnung ist gegeben durch

$$\mathbf{x}'^\top \mathtt{F}\mathbf{x} = 0 \tag{2.30}$$

Dies ergibt sich aus der Bedingung, dass Punkt $\mathbf{x}'$ auf der Epipolarlinie $\mathbf{l}'$ liegt. Nach den Regeln der projektiven Geometrie gilt somit $0 = \mathbf{x}' \times \mathbf{l}'$. Wird für $\mathbf{l}'$ Gleichung (2.29) eingesetzt, ergibt sich Gleichung (2.30). Aus der Gleichung wird ersichtlich, dass sich die Fundamentalmatrix aus Punktkorrespondenzen bestimmen lässt. Die Anzahl notwendiger Punktkorrespondenzen ergibt sich dabei aus den Freiheitsgraden der Matrix. Eine allgemein projektive Abbildung im $\mathbb{P}^2$ besitzt acht Freiheitsgrade, da sie bis auf einen Skalierungsfaktor definiert ist (vgl. Abschnitt 2.1.2). Aus der Eigenschaft $rang(\mathtt{F}) = 2$ ergibt sich die Bedingung $det(\mathtt{F}) = 0$, welche die Freiheitsgrade auf sieben reduziert und sich somit die Matrix mit sieben Punktkorrespondenzen bestimmen lässt. Aufgrund von Fehlzuweisungen durch Rauschen oder Mehrdeutigkeiten sind zur robusten Bestimmung jedoch mehr Punktkorrespondenzen von Vorteil. Zur Berechnung kommen dann üblicherweise iterative Verfahren wie RANSAC (*Random Sample Consensus* [Fischler und Bolles, 1981]) zum Einsatz.

Die wichtigsten Eigenschaften der Fundamentalmatrix sind (vgl [Hartley und Zisserman, 2008]):

- $\mathtt{F}$ ist nur von den Projektionsmatrizen und somit vom gewählten Bildkoordinatensystem, jedoch nicht von der Wahl des Weltkoordinatensystems abhängig.

- Die Fundamentalmatrix ist invariant gegenüber einer projektiven Transformation des $\mathbb{P}^3$.

- Die Fundamentalmatrix lässt sich aus den Projektionsmatrizen ableiten, jedoch nicht umgekehrt. $\mathtt{F}$ definiert $\mathtt{P}$ und $\mathtt{P}'$ nur bis auf eine projektive Transformation.

Letztgenannte Eigenschaft soll an dieser Stelle kurz näher betrachtet werden. Bei einem kalibrierten Stereokamerapaar (Kalibrierungsmatrix $\mathtt{K}$ und Transformation $\mathtt{R}$, $\mathbf{t}$ sind bekannt) lässt sich aus der Annahme einer Kamera im Ursprung des Weltkoordinatensystems mit Projektionsmatrix $\mathtt{P} = \mathtt{K}[\mathtt{I}|\mathbf{0}]$ und der zweiten Kamera mit $\mathtt{P}' = \mathtt{K}'[\mathtt{R}|\mathbf{t}]$ über die Epipolarbeziehungen folgender Zusammenhang herleiten [Hartley und Zisserman, 2008, S. 244]:

$$\mathtt{F} = \mathtt{K}'^{-\top}\mathtt{R}[\mathtt{R}^\top\mathbf{t}]_\times\mathtt{K}^{-1} = \mathtt{K}'^{-\top}\mathtt{R}\mathtt{K}^\top[\mathbf{e}]_\times \tag{2.31}$$

wobei $\mathbf{e} = \mathtt{K}\mathtt{R}^\top\mathbf{t}$ der Epipol ist, dessen Darstellung sich aus der Projektion der Kamerazentren ableiten lässt:

$$\mathbf{e} = \mathtt{P}\begin{pmatrix}-\mathtt{R}^\top\mathbf{t}\\ 1\end{pmatrix} = \mathtt{K}\mathtt{R}^\top\mathbf{t} \qquad\qquad \mathbf{e}' = \mathtt{P}'\begin{pmatrix}\mathbf{0}\\ 1\end{pmatrix} = \mathtt{K}'\mathbf{t} \tag{2.32}$$

Die Notation $[\mathbf{a}]_\times$ beschreibt eine $3 \times 3$ antisymmetrischen Matrix, deren linker und rechter Nullvektor $\mathbf{a} = (a_1, a_2, a_3)^\top$ ist:

$$[\mathbf{a}]_\times = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}$$

Der Zusammenhang mit dem Kreuzprodukt besteht in:

$$\mathbf{a} \times \mathbf{b} = [\mathbf{a}]_\times\mathbf{b} = (\mathbf{a}^\top[\mathbf{b}]_\times)^\top$$

Mit Gleichung (2.31) lässt sich die Fundamentalmatrix aus den Projektionsmatrizen bestimmen.

Die Beziehung zwischen den zwei Abbildungen $\mathbf{x}$ und $\mathbf{x}'$ eines Punktes $\mathbf{X}$ lässt sich durch dessen Rückprojektion und Projektion in das zweite Bild herleiten (vgl. [Schreer u. a., 2005, S. 94][Hartley und Zisserman, 2008, S. 248]). Wird das Weltkoordinatensystem in die erste Kamera gelegt, so dass $\mathtt{P} = \mathtt{K}[\mathtt{I}|\mathbf{0}]$ und $\mathtt{P}' = \mathtt{K}'[\mathtt{R}|\mathbf{t}]$, ergibt sie sich zu:

$$\mathbf{x}' = \mathtt{K}'\mathtt{R}\mathtt{K}^{-1}\mathbf{x} + \frac{1}{Z}\mathtt{K}'\mathbf{t} \tag{2.33}$$

Diese Gleichung wird als *allgemeine Disparitätsgleichung* bezeichnet. $Z$ ist die orthogonale Distanz des Raumpunktes $\mathbf{X}$ zur ersten Kamera. Befindet sich Raumpunkt $\mathbf{X}$ auf der Ebene im Unendlichen (*plane at infinity*, vgl. Abschnitt 2.1.1) wird $Z$ unendlich groß und die allgemeine

Disparitätsgleichung vereinfacht sich zu:

$$\mathbf{x}' = \mathtt{H}_\infty \mathbf{x} \qquad \text{mit } \mathtt{H}_\infty = \mathtt{K}'\mathtt{R}\mathtt{K}^{-1} \tag{2.34}$$

Anschaulich ist dies durch sehr weit entfernte Punkte vorstellbar, die ihre translatorische Komponente ($\mathbf{t}$ im zweiten Term) bei einer Bewegung der Kameras verlieren.

Die Projektionsmatrizen $\mathtt{P}$ und $\mathtt{P}'$ sind, wie in Abschnitt 2.1.3 gezeigt, von der Wahl des Weltkoordinatensystems abhängig, was für $\mathtt{F}$ nicht zutrifft. Dies bedeutet, dass für eine beliebige projektive Transformation des dreidimensionalen Raums mittels einer $4 \times 4$-Homographie $\mathtt{H}$ die Fundamentalmatrizen für das Kamerapaar $(\mathtt{P}, \mathtt{P}')$ und das transformierte Paar $(\mathtt{PH}, \mathtt{P}'\mathtt{H})$ identisch sind. Für den Beweis sei auf [Hartley und Zisserman, 2008, S. 254] verwiesen. $\mathtt{F}$ ist somit bis auf eine projektive Transformation definiert. Durch diese Uneindeutigkeit lassen sich die Kameramatrizen basierend auf einer Fundamentalmatrix faktisch beliebig darstellen, so lange o.g. Beziehungen erhalten bleibt. Eine oft genutzte, da praktisch zu verwendende Form ist die kanonische Form der Projektionsmatrizen:

$$\mathtt{P} = [\mathtt{I}|\mathbf{0}] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{2.35}$$

und

$$\mathtt{P}' = [\mathtt{M}|\mathbf{m}] \tag{2.36}$$

wobei sich die Fundamentalmatrix zu $\mathtt{F} = [\mathbf{m}]_\times \mathtt{M}$ ergibt.

Die Epipolargeometrie bildet eine wichtige Basis für Teilbereiche der Arbeit. So wird die Suche nach korrespondierenden Bildpunkten in einer allgemeinen Stereoanordnung auf die Suche entlang einer Geraden im anderen Bild beschränkt. Wird die Fundamentalmatrix für die Rektifizierung der Bilder derart genutzt, dass die Epipolarlinien anschließend waagerecht im Bild liegen, so reduziert sich die Suche auf eine Dimension des Bildes. Zur Diskussion sei auf den Abschnitt 5.1.2 verwiesen. Auch lässt sich die Fundamentalmatrix zur Ableitung des Trifokalen Tensors nutzen, der in der Arbeit als eine Variante zur physikalisch korrekten Bildsynthese zum Einsatz kommt (vgl. Abschnitt 2.3).

### 2.1.6. Trifokale Geometrie

Die trifokale Geometrie beschreibt die geometrischen Relationen zwischen drei Kameras. Sie kann dabei helfen, zusätzliche Randbedingungen für die Tiefenanalyse zu liefern. Sie kann aber auch, wie in dieser Arbeit geschehen, zur Synthese neuer Ansichten verwendet werden. Die dafür notwendigen Grundlagen werden im Folgenden präsentiert.

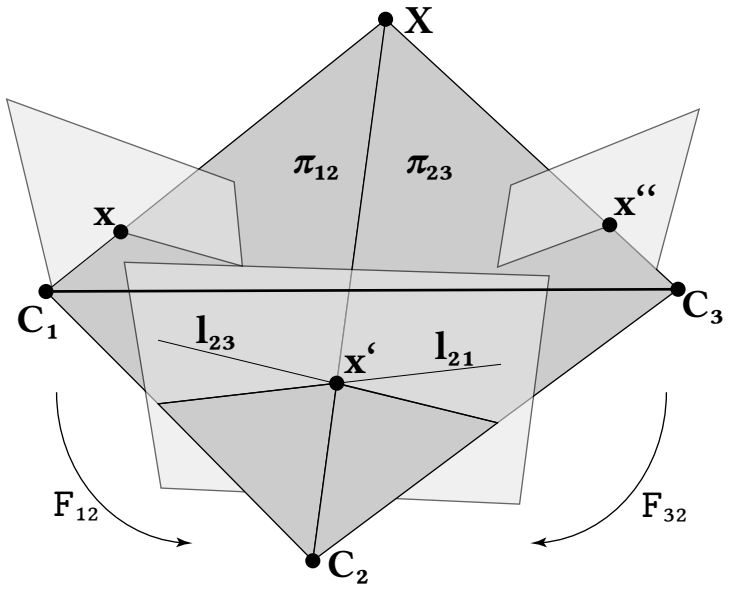



**Abbildung 2.9.:** Die Trifokalgeometrie zwischen drei Kameras ($\mathbf{C_1}$, $\mathbf{C_2}$, $\mathbf{C_3}$) vom Standpunkt der Epiploargeometrie. Punkt $\mathbf{x}'$ in Ansicht 2 kann durch den Schnittpunkt der Epipolarlinien ermittelt werden.

**Betrachtung vom Standpunkt der Epiploargeometrie:** Wie im vorigen Abschnitt beschrieben, bilden zwei Kamerazentren und ein Raumpukt die Epipolarebene $\pi$, deren Orientierung von der Lage des Raumpunktes abhängig ist. Wird nun ein drittes Kamerazentrum anstelle des Raumpunktes gewählt, so spannen die drei Kamerazentren $\mathbf{C_1}$, $\mathbf{C_2}$ und $\mathbf{C_3}$ die *trifokale Ebene* auf. Diese ist von der Lage von Raumpunkten unabhängig, existiert jedoch nur, wenn die Kamerazentren nicht kollinear angeordnet sind, also in allgemeiner Lage befinden [Kreibich, 2005]. Es existieren nun mehrere Fundamentalmatrizen $\mathtt{F}_{\mathtt{ij}}$, welche die Beziehung zwischen den drei Ansichten beschreiben. So lässt sich beispielsweise aus den korrespondierenden Abbildungen $\mathbf{x}$ und $\mathbf{x}''$ in Ansicht 1 und 3 die entsprechende Abbildung $\mathbf{x}'$ in Ansicht 2 ermitteln, indem der Schnittpunkt der beiden Epipolarlinien $\mathbf{l_{21}}$ und $\mathbf{l_{23}}$ bestimmt wird [Faugeras u. a., 2001, S. 412f] [Schreer u. a., 2005, S. 186f] (vgl. Abb. 2.9):

$$\mathbf{x}' = \mathbf{l_{21}} \times \mathbf{l_{23}} = (\mathtt{F}_{32}\mathbf{x}'') \times (\mathtt{F}_{12}\mathbf{x}) \tag{2.37}$$

Dieses Epipolartransfer genannte Prinzip gilt jedoch nur unter folgenden Bedingungen:

1. $\mathbf{x} \neq \mathbf{e_{13}}$ und $\mathbf{x}'' \neq \mathbf{e_{31}}$. D. h. die Abbildungen stimmen nicht mit den jeweiligen Epiolen zwischen Ansicht 1 und 3 überein.
2. Die Kamerazentren sind nicht kollinear angeordnet.
3. Raumpunkt $\mathbf{X}$ liegt nicht in der trifokalen Ebenen.

**Trifokaler Tensor:** Die o. g. singulären Fälle werden nun durch die Erweiterung der Epipolargeometrie ausgeschlossen. Ähnlich der Fundamentalmatrix wird dazu die Beziehungen zwischen drei Ansichten mit einem Tensor der Stufe 3, dem *trifokalen Tensor* beschrieben.

Der trifokale Tensor ist von der relativen Position der Kameras zueinander und von deren internen Parametern abhängig. Er wird mittels Kameraparametern oder anhand von Punktkorrespondenzen aus den drei Referenzansichten berechnet. Zur Schätzung des Tensors eignen sich wie zur Bestimmung der Fundamentalmatrix robuste Algorithmen [Kreibich, 2005].

Um eine kompaktere Darstellung der algebraischen Beziehungen zu ermöglichen, wird in der Literatur auf eine spezielle Tensor-Notation [Avidan und Shashua, 1998] zurückgegriffen. In Tensor-Notationen werden hochgestellte Indizes als *kontravariant*, tief gestellte Indizes hingegen als *kovariant* bezeichnet. Kontravariante Vektoren repräsentieren Punkte und deren Koordinaten: $x^i = (x^1, x^2, \ldots)$. Kovariante Vektoren repräsentieren Geraden und deren Elemente im $\mathbb{P}^2$: $l_j = (l_1, l_2, \ldots)$. Identisch auftretende kovariante und kontravariante Indizes in einem Produkt werden aufsummiert, was als *Kontraktion* bezeichnet wird. Beispielsweise gilt: $x^i l_i = x^1 l_1 + x^2 l_2 + x^3 l_3 + \ldots + x^n l_n$. Die Multiplikation zweier Tensoren erster Stufe (Vektoren) ergibt einen Tensor zweiter Stufe (Matrix), der mit hoch- und tief gestellten Indizes beziffert wird, z. B. $c^j_i = a_j b^i$. Die Gleichung $\mathbf{x}' = \mathtt{A}\mathbf{x}$ wird somit ausgedrückt als $x'^i = \sum_j a^i_j x^j$. Unter Verwendung der Summenkonvention wird die Beziehung vereinfacht zu $x'^i = a^i_j x^j$.

Die im vorherigen Abschnitt eingeführte kanonische Form der drei Projektionsmatrizen wird mittels Tensornotation folgendermaßen geschrieben:

$$\mathtt{P} = [\mathtt{I}|\mathbf{0}] \qquad \mathtt{P}' = [a^i_j] \qquad \mathtt{P}'' = [b^i_j] \tag{2.38}$$

Der trifokale Tensor ergibt sich dann zu

$$\mathcal{T}^{jk}_i = a^j_i b^k_4 - a^j_4 b^k_i \qquad \text{mit } i, j, k = 1, 2, 3 \tag{2.39}$$

Der trifokale Tensor ist demnach Tensor dritter Stufe mit zwei kontravarianten und einem kovarianten Index, der ein Array mit $3 \times 3 \times 3$ Einträgen repräsentiert und sich aus den Kameramatrizen aus den Ansichten 2 und 3 zusammensetzt [Avidan und Shashua, 1997a,b, 1998].

Die entscheidende Frage ist nun, wie der Tensor die Abbildungen $\mathbf{p} = (x, y, 1)$, $\mathbf{p}' = (x', y', 1)$, $\mathbf{p}'' = (x'', y'', 1)$ eines Raumpunktes $\mathbf{P}$ in eine mathematische Beziehung setzt, um so anhand von bekannten Korrespondenzen in zwei Ansichten die Position der Abbildung in der dritten Ansicht zu berechnen. Hierzu werden pro Abbildung zwei Geraden $s_j$, $r_k$ definiert, die jeweils die Abbildungen enthalten, d. h. $s_j p'^j = 0$ und $r_k p''^k = 0$. Für $\mathbf{p}'$ seien diese die waagerechte Gerade $(-1, 0, x')$ und die dazu orthogonale Gerade $(0, -1, y')$ (vgl. auch Geradennotation und

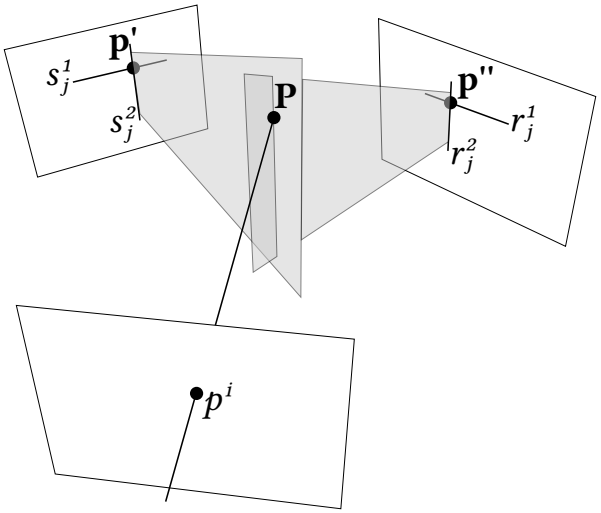



**Abbildung 2.10.:** Jede der Basistrilinearitäten beschreibt die Zuordnung eines Punktes $p^i$ zu Geraden in $s_j^\mu$ und $r_k^\rho$ durch die Punkte $\mathbf{p}'$ und $\mathbf{p}''$. Im Raum ist dies der Schnittpunkt zweier Ebenen mit einem Strahl im Raumpunkt $\mathbf{P}$. Nach [Avidan und Shashua, 1998]

Schnittpunktberechung im $\mathbb{P}^2$ in Abschnitt 2.1.1). Äquivalent dazu sind das für $\mathbf{p}''$ die Geraden $r_k$ mit $(-1, 0, x'')$ und $(0, -1, y'')$. Als Matrix in Tensornotation werden als

$$s_j^\mu = \begin{bmatrix} -1 & 0 & x' \\ 0 & -1 & y' \end{bmatrix} \quad \text{und} \quad r_k^\rho = \begin{bmatrix} -1 & 0 & x'' \\ 0 & -1 & y'' \end{bmatrix} \quad \text{mit } \mu, \rho = 1, 2\,;\, j, k = 1 \ldots 3$$

geschrieben. Abbildung 2.10 veranschaulicht die Zusammenhänge. Der Zusammenhang mit dem trifokalen Tensor und der ersten Abbildung $p^i = (p^1, p^2, p^3)$ (in Tensornotation) ist nun folgendermaßen:

$$p^i s_j^\mu r_k^\rho \mathcal{T}_i^{jk} = 0 \tag{2.40}$$

Die Herleitung dieses Zusammenhangs lässt sich über die kanonische Form der Projektionsmatrizen realisieren und ist u. a. in [Shashua, 1997] ausgeführt. Die freien Indizes $\mu$ und $\rho$ in Gleichung (2.40) sind nicht der Summenkonvention unterlegen und jeweils für den Bereich $1, 2$ definiert, wodurch sich vier trilineare Gleichungen ergeben, die jeweils die Abbildung von $p^i$ und den Geraden repräsentieren (vgl. Abb. 2.10). Diese Gleichungen werden Basistrilineariäten genannt und lauten folgendermaßen:

$$\begin{aligned}
x''\mathcal{T}_i^{13}p^i - x''x'\mathcal{T}_i^{33}p^i + x'\mathcal{T}_i^{31}p^i - \mathcal{T}_i^{11}p^i = 0, \\
y''\mathcal{T}_i^{13}p^i - y''x'\mathcal{T}_i^{33}p^i + x'\mathcal{T}_i^{32}p^i - \mathcal{T}_i^{12}p^i = 0, \\
x''\mathcal{T}_i^{23}p^i - x''y'\mathcal{T}_i^{33}p^i + y'\mathcal{T}_i^{31}p^i - \mathcal{T}_i^{21}p^i = 0, \\
y''\mathcal{T}_i^{23}p^i - y''y'\mathcal{T}_i^{33}p^i + y'\mathcal{T}_i^{32}p^i - \mathcal{T}_i^{22}p^i = 0.
\end{aligned} \tag{2.41}$$

Mittels der Basistrilinearitäten kann ein Punkt in der dritten Ansicht aus korrespondierenden Punkten der ersten und zweiten Ansicht ermittelt werden. Details zur Bestimmung des Tensors sowie der Lösung des Gleichungssystems sind im Abschnitt 5.3 zur Bildsynthese dargelegt. Es gibt noch eine Reihe weiterer trifokaler Transferbeziehungen und Eigenschaften des Tensors, die ausführlich in [Hartley und Zisserman, 2008] beschrieben werden.

## 2.2. Stereoanalyse

Verfahren der Stereoanalyse haben zum Ziel, Informationen über die Dreidimensionalität einer natürlichen Szene zu erfassen, sie in eine bestimmte Repräsentationsform zu überführen und diese direkt auszuwerten, zu übertragen oder zu speichern. Wie aus dem Namen ableitbar, kommen hierbei zwei Kameras zum Einsatz, die in verschiedenen Anordnungen Bild- oder Videoaufnahmen der Szene erstellen. Andere Verfahren mit demselben Ziel sind unter anderem Laserscanner [Blais, 2004], Tiefenkameras nach verschiedenen Prinzipien wie *Time of Flight* [Gokturk u. a., 2004], oder Musterprojektionen, welche jedoch an dieser Stelle nicht näher betrachtet werden. Eine häufige Repräsentationsform für Stereoanalyseverfahren ist die Speicherung der Tiefen- oder Disparitätswerte als monochromes Pixelbild. Diese Repräsentation lässt sich leicht speichern, komprimieren und manipulieren und im zweidimensionalen Kontext betrachten. Ordnen die Disparitätskarten jedem Farbpixel eines Kamerabildes einen entsprechenden ganzzahligen Tiefen- oder Disparitätswert zu, spricht man von *dichten Disparitätskarten*. Die Stereoanalyse ist essentieller Bestandteil der Arbeit, da auf Basis der dichten Disparitätskarten eine virtuelle Ansicht zur Blickkorrektur erzeugt wird.

### 2.2.1. Relevante Kenngrößen digitaler Bilderzeugung

An dieser Stelle sollen wichtige Sachverhalte bei der eigentlichen Bildentstehung thematisiert werden. Nahezu alle Methoden der Stereoanalyse verwenden direkt oder indirekt das erzeugte Signal und werden somit davon beeinflusst. Daher werden in diesem Abschnitt die für die Stereoanalyse und Bildsynthese relevanten Kenngrößen kurz dargestellt.

**Auflösung und Rauschen:** Digitale Kameras verwenden vornehmlich lichtempfindliche Sensoren nach dem CCD- oder CMOS-Prinzip. Einfallende Photonen werden durch deren Sensorelemente in elektrische Ladungen umgewandelt, welche durch verschiedene Ausleseprinzipien letztlich zu einem Spannungswert für jedes Pixel führen. Nach [Jähne, 2005] kann ein kontinuierliches Bild als „flächenhafte Verteilung der Bestrahlungsstärke in einer Ebene“ beschrieben werden. Dieses kontinuierliche Bild wird durch die Gitterstruktur örtlich zweidimensional abgetastet. Während ein Pixel eine infinitesimal kleine, diskrete Position angibt, so repräsentiert sein Wert die Fläche des jeweiligen Sensor- oder Elementarelementes bzw. die darauf auftreffende

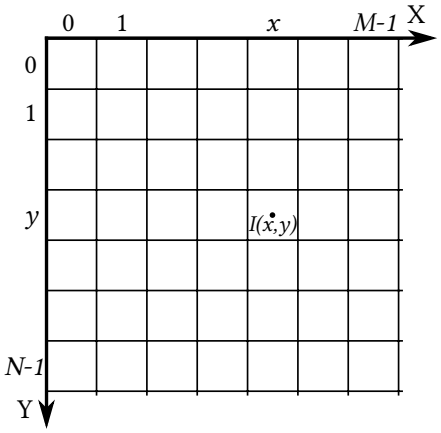
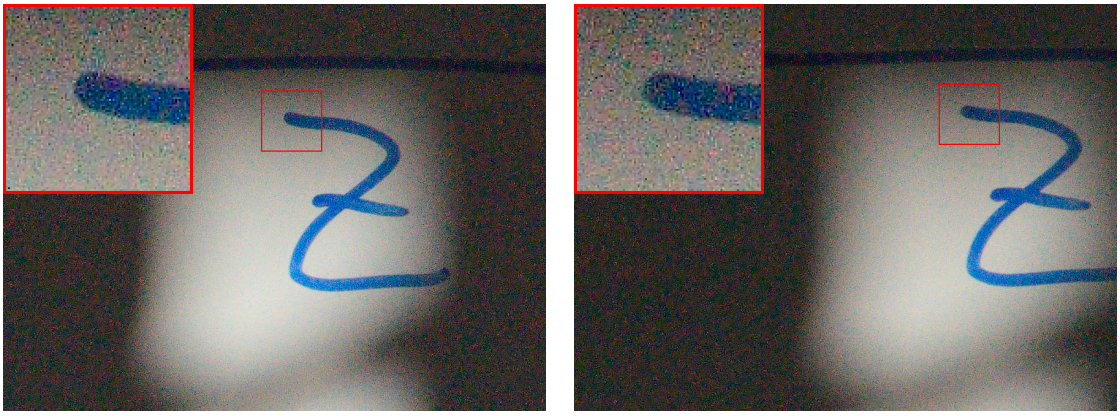



**Abbildung 2.11.:** Pixelraster bei digitalen Bildern (nach [Jähne, 2005]). Die Abbildung reeller 3D-Punkte als ein kontinuierlichen Bildes wird durch das Raster des Bildsensors abgetastet und ergibt das diskrete Bild $I(x, y)$.

Bestrahlungsstärke [Jähne, 2005]. Baubedingt kann in horizontaler und vertikaler Richtung nur eine bestimmte Anzahl von Sensoren platziert werden. Allgemein gilt, dass die Anzahl der aktiven Elemente auf der Sensorfläche der Anzahl der Pixel des Bildes in vertikaler und horizontaler Richtung entspricht. Oft wird in diesem Zusammenhang der Begriff *Auflösung* des Sensors verwendet. Dies kann zu Missverständnissen führen, da der Begriff mitunter auch für die Anzahl von Pixeln pro Längeneinheit ( `dpi`) verwendet wird. Entsprechend Abb. 2.4b wird das Bild auf ein rechteckiges diskretes Gitter $I$ der Größe $M \times N$ abgebildet, wobei $M$ die Anzahl der Spalten und $N$ die Anzahl der Zeilen des Bildes angibt. Die Position eines Pixels wird durch dessen ganzahlige Indizes $x = 0 \ldots M - 1$ und $y = 0 \ldots N - 1$ angegeben. Abbildung 2.11 veranschaulicht den Sachverhalt. Dieses Koordinatensystem entspricht dem Koordinatensystem in Abb. 2.4, jedoch sind $x$ und $y$ nun auf die eben eingeführten diskreten Positionen beschränkt. Spezielle Gitterstrukturen wie dreieckige oder hexagonale werden in dieser Arbeit nicht betrachtet.

Die Anzahl der Sensorelemente bestimmt die Abbildung von Details im Bild sowie deren Wahrnehmung durch den Menschen. Für die Stereoanalyse ist primär der erste Faktor von Bedeutung, da Aliasing-Effekte durch zu geringe örtliche Abtastung zur fehlerhafter Erkennung von Korrespondenzen führen können. Andererseits erzeugt eine höhere Anzahl von Sensoren bei gleichbleibender Fläche auch Probleme, da die Fläche eines Sensorelementes dadurch kleiner ist. Die Gesamtgröße des Sensors hat direkten Einfluss auf die Lichtmenge (Anzahl der Photonen), die jedes Sensorelement erreicht, da – einen idealen Sensor angenommen – über die Sensorfläche integriert wird [Jähne, 2005] [Clark, 2005]. Dieser Umstand ist von Bedeutung, da in jedem Kamerasystem verschiedene Arten von Rauschen auftreten können. Neben Photonen- / Schrotrauschen, Dunkelstrom- oder thermischem Rauschen ist auch das Verstärkerrauschen er-



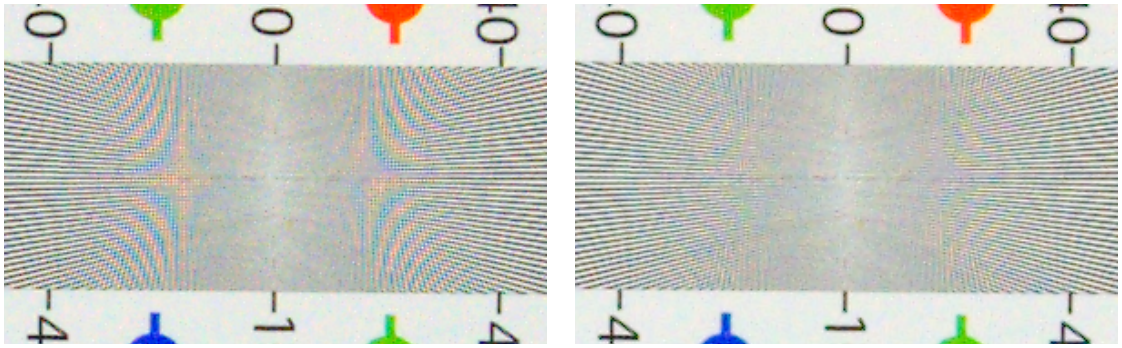
(a) (b)

**Abbildung 2.12.:** Stark verrauschte Stereoaufnahme. Das zufällige Rauschmuster kann zu Fehlerkennungen bei der Korrespondenzsuche (*Matching*) zwischen linkem (a) und rechtem (b) Bild führen.

wähnenswert [Holst, 1998] [Schmidt, 2009, S. 372]. So unterschiedliche Ursachen die Rauscharten haben, sie manifestieren sich alle in einer zufälligen Veränderung der Intensität der Bildkanäle. Wird der Signal-Rauschabstand durch längere Belichtung oder übermäßige Verstärkung zu gering, ist diese Intensitätsänderung von gravierendem Einfluss für die Bildverarbeitung im Allgemeinen und für die Stereoanalyse im Speziellen. Dieser Umstand wird in Abb. 2.12 deutlich.

**Quantisierung:** Nach der örtlichen Abtastung und dem Auslese- und Verstärkungsvorgang liegen für jedes Pixel kontinuierlich Signale vor, die der jeweiligen Beleuchtungsstärke entsprechen. Da ein digitales Bild erzeugt werden soll, müssen diese auf einen diskreten Wertebereich abgebildet werden. Die Analog-Digital-Wandler vieler Kameras unterstützen heutzutage bereits eine Quantisierung mit 10 `Bit` oder mehr. Die notwendige Anzahl von Quantisierungsstufen hängt auch hier wieder vom Nutzungskontext ab. Auf Basis umfangreicher Recherchen zum aktuellen Stand der Technik im Bereich der Stereoanalyse ist festzustellen, dass üblicherweise eine 8 `Bit` Quantisierung pro Bildkanal als ausreichend angenommen wird. Dieser Umstand scheint zunächst verwunderlich, da eine geringere Quantisierungstufenzahl mit einem Informationsverlust einhergeht und höheres Quantisierungsrauschen die Folge ist. Es müssen jedoch auch praktische Erwägungen wie erzeugte Datenmenge, Beschränkungen verarbeitender Werkzeuge, nutzbare Bilddateiformate oder Kosten für Testaufbauten in die Überlegung mit einbezogen werden.

**Farbaufnahme:** Die Sensorelemente können aufgrund Ihres Funktionsprinzips nur die Stärke des einfallenden Lichtes erfassen und in Intensitätswerte wandeln. Die Wellenlänge und somit die Farbwirkung müssen durch andere technische Methoden erfasste werden. Die technische



(a) (b)

**Abbildung 2.13.:** Hochfrequente Strukturen führen im Bild zu Interferenzmustern durch Aliasing. Ebenso erzeugt die Interpolation des Bayer-Patterns unerwünschte und in einem Stereosetup nicht deterministische Farbeffekte. (a) Verwendung eines einfachen Interpolationsalgorithmus zum Demosaicing. Bläuliche und rötliche Fehlfarben sind im Zentrum des Bildes klar zu erkennen. (b) Der in dieser Abb. verwendete hochqualitative Interpolationsalgorithmus erzeugt weniger Farbfehler, kann sie jedoch auch nicht vollends vermeiden.

Quantifizerung von Farbe und Farbeindrücken, *Farbmetrik* bezeichnet, ist ein weites und komplexes Wissenschaftsgebiet und kann hier nicht vollständig abgehandelt werden. Daher beschränkt sich dieser Teil auf relevante Fakten.

Abgeleitet aus der Dreifarbentheorie [Foley u. a., 1991, S. 576ff][Möllering und Slansky, 1993, S. 195ff] „teilen“ Aufnahmegeräte das Licht in die Farben Rot, Grün und Blau auf, um die Intensität der jeweiligen Grundfarbe, d. h. die jeweilige Wellenlänge, separat zu erfassen. Grundsätzlich existieren zwei Arten der Farberfassung. Insbesondere teurere Kameras nutzen für jede der drei Grundfarben eine eigenen Sensor. Der spektrale Rot-, Grün- oder Blauanteil des einfallenden Lichtes wird über Strahlenteilerprisma auf den jeweiligen Sensor gelenkt. Somit entstehen für jeden Farbkanal eigene Intensitätswerte. Die zweite Technik nutzt nur einen Sensor und eine Mosaikfilter, wobei vor jedem Sensorelement ein Filter für wellenlängenabhänge Transmission der RGB-Farbanteile zum Einsatz kommt. Oft wird dabei das *Bayer-Pattern* verwendet [Schmidt, 2009, S. 378f]. Die erste Methode erzeugt aufgrund der heutigen hohen Präzision solcher Bauteile eine sehr gute Trennung der einzelnen Farbkanäle und erzeugt somit ein Bild mit voller Farbauflösung. Bei Kameras mit einem Sensor werden die Farben faktisch örtlich unterabgetastet. Auf Basis der Annahme, dass die Farbe nahe beieinander liegender Pixel nahezu identisch sind, werden anschließen für jedes Sensorelement die jeweils fehlenden zwei Farben aus den Nachbarfarben mit verschiedenen Verfahren interpoliert (*Demosaicing*) wodurch ein Informationsverlust entsteht. Je nach Qualität des Interpolationsalgorithmus können an Kanten Farbsäume entstehen, die in Stereoaufnahmen aufgrund der unterschiedlichen Position der Kameras zu Ungleichheiten zwischen linken und rechtem Bild führen können (vgl. 2.13).

**Farbräume:** Liegen die quantisierten (und ggf. interpolierten) RGB-Werte für jedes Pixel vor, kann das Bild der Betrachtung oder Analyse zugeführt werden. Welchen Bereich die RGB-Komponenten des aufgenommenen Bildes im Bereich aller sichtbaren Farben abdecken, hängt von den Fähigkeiten des verwendeten Sensors ab. Auch wie die Farben bei einer Betrachtung letztendlich auf z. B. einem Monitor wirken, ist ein Spezifikum des Wiedergabegerätes. Für die Bildanalyse ist die realistische Wirkung von Farben nicht von so hoher Bedeutung wie beispielsweise für die Betrachtung oder bei der Druckausgabe. Der Farbraum in dem das analysierte Bild repräsentiert wird, hat dennoch einen wichtigen Einfluss. So hängen z. B. Eigenschaften wie die Dimensionalität von Merkmalsvektoren oder die Quantität von Abstandsmaßen vom gewählten Farbraum ab. Die Zerlegung von Farbsignalen in reine Helligkeits- und Farbinformationen hilft bei der Interpretation bestimmter Sachverhalte. Auch nutzen manche Verfahren Annahmen über die menschliche Wahrnehmung, weshalb dabei Farbräume zum Einsatz kommen, die dieser möglichst entsprechen.

### 2.2.2. Korrespondenzproblem und Nebenbedingungen der Stereoanalyse

Die Zusammenhänge zwischen einem 3D-Punkt und seiner Abbildung auf ein achsenparallel angeordnetes Stereosystem wurden bereits im Abschnitt 2.1.5 erläutert. Die folgenden Abhandlungen beziehen sich auf diesen Fall, da dieser fast ausschließlich als Voraussetzung für Algorithmen der Stereoanalyse angenommen wird. Methoden der Rektifizierung, die eine allgemeine Stereoansicht so transformieren, dass sie als achsenparallel behandelt werden kann, werden in Abschnitt 5.1.2 diskutiert. Um mit Gleichung (2.27) die Tiefe $Z$ zu bestimmen, muss die Disparität bekannt sein. Diese ergibt sich unter der Annahme einer achsenparallelen Anordnung entsprechen Gleichung (2.25) zu:

$$d = x - x'$$

Es wird ersichtlich, dass zur Bestimmung der Disparität korrespondierende Bildpunkte – also die Abbildung desselben 3D-Punktes in beiden Bildern – gefunden werden müssen. Bei dichten Disparitätskarten muss dies für jedes Bildpixel durchgeführt werden. Der naive Ansatz dazu besteht darin, für jedes Pixel eines Bildes das korrespondierende Pixel im *gesamten* anderen Bild mittels Intensitäts- bzw. Farbvergleich zu finden. Dieser Vorgang ist sehr aufwändig und es existieren viele Mehrdeutigkeiten (vgl. Abb. 2.14). Daher wird der Vorgang auch als *Korrespondenzproblem* bezeichnet. Ursachen für das Korrespondenzproblem können in szenen- und kamerainduzierte Ursachen unterteilt werden [Döhring, 2009, S. 24].

Szeneninduzierte Probleme sind schwach texturierte, farblich homogene Regionen, repetierende Muster, sowie nicht lambertsche Oberflächen (Transparenz, Reflexionen). Die Eigenschaft lambertscher Oberflächen in der Szene wird oft als gegeben angenommen und besagt, dass die Reflexion von Licht durch die Szene unabhängig von der Position des Betrachters konstant und

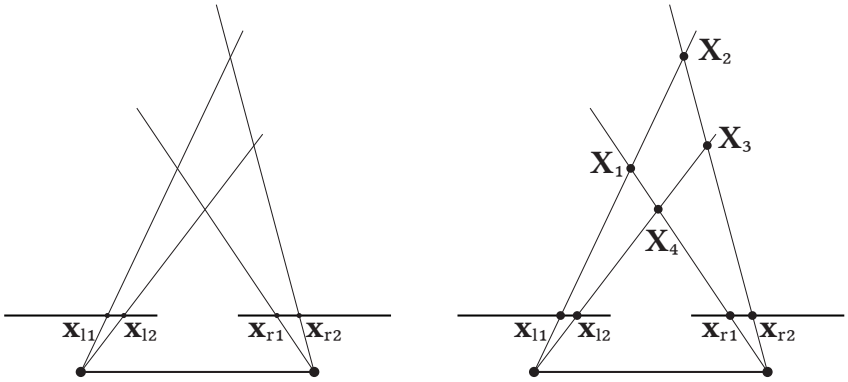



**Abbildung 2.14.:** Einfaches Beispiel zur Veranschaulichungen von Mehrdeutigkeiten bei der Stereoanalyse. Unter der Annahme, dass die kollinearen Punkte $\mathbf{x_{l1}}, \mathbf{x_{l2}}, \mathbf{x_{r1}}, \mathbf{x_{r2}}$ dieselben Intensitätswerte haben und somit als korrespondierend angenommen werden können, ergeben sich allein $2! = 4$ mögliche Raumpunkte $\mathbf{X_{1..4}}$ bei einer Rekonstruktion.

gleichmäßig in alle Richtungen stattfindet.[Szeliski, 2011, S. 57] (diffuse Reflexion). Auch die Kameraanordnung kann, da sie in der Szene stattfindet, zu dieser Gruppe gezählt werden. Durch den Versatz der Kameras in einem Stereoaufbau sind Szenenbestandteile im Bildrandbereich der einen Kamera enthalten, die in der anderen Kamera fehlen. Ebenso treten partielle Verdeckungen bzw. Aufdeckungen durch die Aufnahme von unterschiedlich tiefen Szenenobjekten aus unterschiedlichen Perspektiven auf.

Kamerainduzierte Probleme ergeben sich aufgrund von Objektiv- und Sensoreigenschaften (vgl. Abschnitt 2.2.1) sowie aus Unterschieden in Aufnahmeparametern wie Belichtung oder Weißabgleich. Auch die falsche Bestimmung interner Kameraparameter kompliziert die Problemstellung der intensitätsbasierten Korrespondenzsuche. Abbildung 2.15 veranschaulicht einige der genannten Fehlerquellen.

Um die Korrespondenzsuche einzuschränken, werden verschiedene Nebenbedingungen definiert, die sich aus der Stereo- und Szenengeometrie ableiten lassen (vgl. [Jiang und Bunke, 1997, S. 16ff]). Natürlich sind nicht alle diese Annahmen in allen möglichen Szenen und deren Abbildung gegeben, weshalb Stereoanalyseverfahren diese Fälle meist gesondert betrachten.

**Epipolarbedingung (*epipolar constraint*)**: Wie bereits in den vorherigen Abschnitten geschrieben, lässt sich aus Gleichung (2.29) erkennen, dass ein korrespondierendes Pixel auf der zum Ausgangspixel zugehörigen Epipolarline liegt. Dies reduziert das Problem auf ein eindimensionales und vermindert den Suchraum sowie die Möglichkeit von Mehrdeutigkeiten enorm. In Bildern aus achsenparallelen Anordnungen bzw. rektifizierten Bildern entsprechen die Epipolarlinien den Bildzeilen.

**Eindeutigkeitsannahme (*uniqueness constraint*)**: Die Eindeutigkeitsannahme besagt, dass

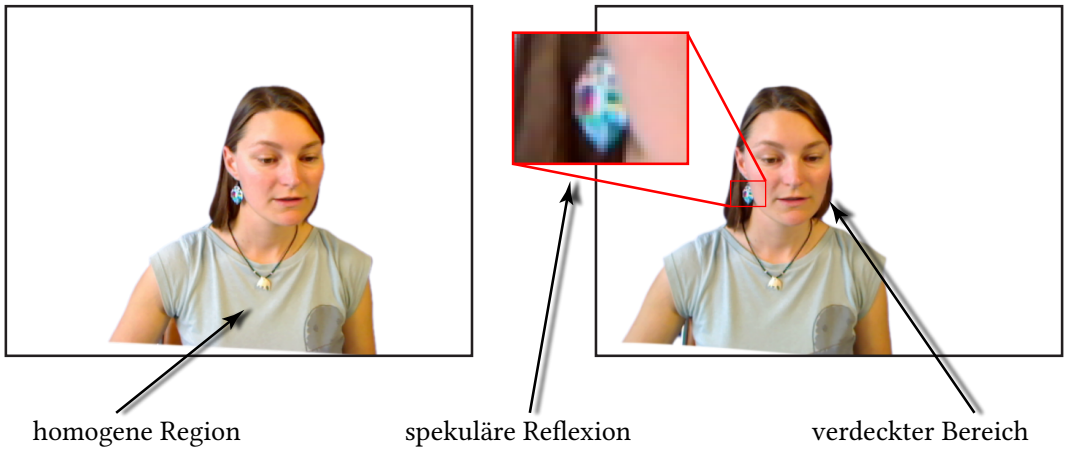



**Abbildung 2.15.:** Drei das Korrespondenzproblem verursachende szeneninduzierte Sachverhalte. Homogene Regionen führen zu Mehrdeutigkeiten bei der Korrespondenzsuche, spekuläre Reflexionen verletzen die Annahme lambertscher Oberflächen und können zu Fehlkorrespondenzen führen. Verdeckte Bereiche habe keine korrespondierendes Pixel im anderen Bild.

jedem Pixel maximal ein korrespondierender oder im Fall von Verdeckungen kein korrespondierendes Pixel zugordnet werden kann. Existiert diese Korrespondenz, so bildet sie denselben eindeutigen Szenenpunkt ab. Diese Annahme wird insbesondere durch Transparenzen verletzt [Marr und Poggio, 1976].

**Reihenfolgeneinschränkung (*ordering constraint*)**: Dieser Einsschränkung liegt die Annahme zugrunde, dass die Reihenfolge der abgebildeten Szenepunkte in beiden Stereobildern identisch ist. Sie kann beispielsweise durch kleine Objekte im Vordergrund verletzt werden.

**Glattheitsbedingung (*smootheness/continuity constraint*)**: Diese Bedingung besagt, dass sich im großen Teil des Bildes benachtbarte Disparitäten nur gering ändern, was seinen Grund in der meist kontinuierlichen Szenestruktur hat. Die Bedingung wird verletzt durch die Verdeckung eines Objektes durch ein Objekt geringerer Tiefe, wodurch ein Disparitätssprung stattfindet.

**Eingeschränkter Disparitätsbereich (*disparity limit*)**: Sind Szenentiefe und Kameraparameter bekannt, so ist eine Berechnung oder Abschätzung der minimalen und maximalen Disparität möglich. Neben der Einschränkung des Suchbereiches ist ein weiterer Vorteil die geringere Anzahl von Rechenschritten bei kleinere Disparitätsbereichen.

### 2.2.3. Taxonomie der Stereoanalysealgorithmen

Daniel Scharstein und Richard Szeliski entwickeln in [Scharstein und Szeliski, 2002] eine Taxonomie, nach der sich nahezu alle modernen Stereoanalysealgorithmen einteilen und vergleichen lassen. Aus den Bildkoordinaten und der Disparität lässt sich der Disparitätsraum (*disparity space*)

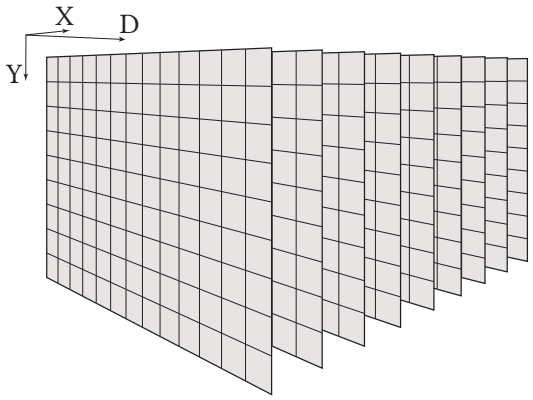



**Abbildung 2.16.:** Visualisierung des diskreten Disparitätsraumes $(X, Y, D)$. $X$ und $Y$ beschreiben die Achsen der Pixelkoordinaten, $D$ die Dimension der Disparität. Der Disparitätsraum wird durch die Ausdehnungen des Bildes sowie durch die Annahme eines eingeschränken Disparitätsbereiches begrenzt.

$(X, Y, D)$ definieren, der in Abbildung 2.16 dargestellt ist. Dieser dient wiederum der Definition eines wichtigen Konzeptes der Stereoanalyse: dem Disparitätsraumbild (*disparity space image*) - DSI [Bobick und Intille, 1999][Szeliski und Golland, 1999]. Das DSI ist, allgemein formuliert, eine Funktion, die über den diskreten oder kontinuierlichen Disparitätsraum definiert ist. Im Kontext der Stereoanalyse beschreibt diese Funktion die *Zuordnungskosten* $C$ für ein bestimmtes Korrespondenzpaar $(x, y, d)$. Ziel der Analysealgorithmen ist es nun, eine Funktion $d(x, y)$ im Disparitätsraum zu finden, die am besten die Oberfläche der analysierten Szene beschreibt [Scharstein und Szeliski, 2002]. Zur Veranschaulichung sind in Abbildung 2.17 Ausschnitte eines DSI als Pixelbilder visualisiert.

Die Vielzahl von Algorithmen zur Stereoanalyse lässt sich in vier grundlegende Schritte einteilen [Scharstein, 1999, S. 33][Scharstein und Szeliski, 2002]:

- Berechnung der Zuordnungskosten
- Kostenaggregation
- Disparitätsberechnung / -optimierung
- Disparitätsverfeinerung / -nachverarbeitung

Die Reihenfolge der Schritte kann je nach Algorithmus variieren, wobei mit den lokalen und globalen Algorithmen zwei große Gruppen der Analysemethoden existieren, die in den Absätzen 2.2.5 und 2.2.6 behandelt werden.

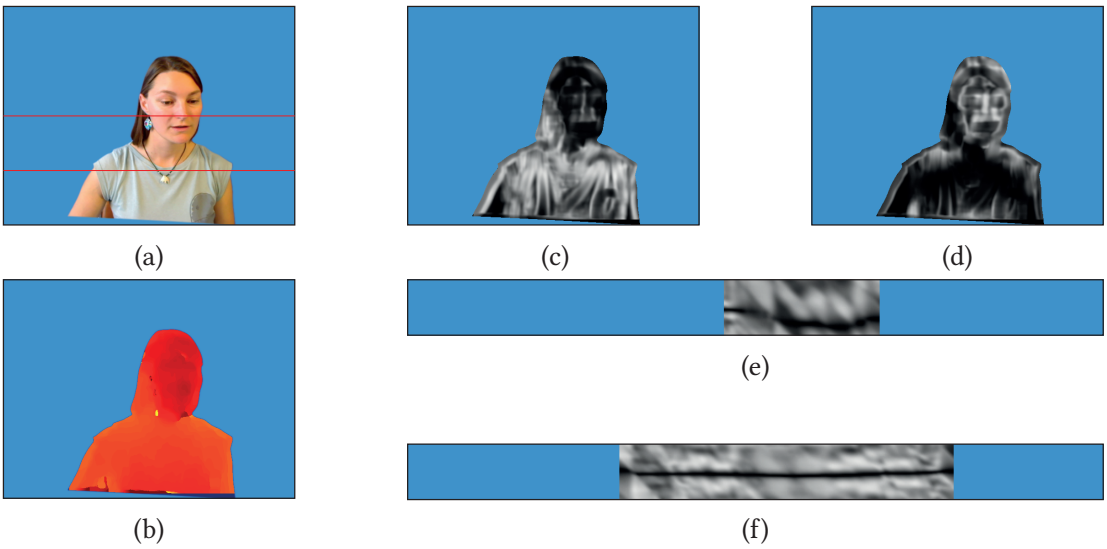



**Abbildung 2.17.:** Auschnitte aus einem DSI. Helle Regionen repräsentieren hohe Zuordnungskosten (niedrige Wahrscheinlichkeit). Blaue Flächen markieren nicht analysierte Bereiche (Segmentierung). (a) Linkes Bild des Stereopaares. (b) Ermitteltes Disparitätsbild für die linke Ansicht. (c) $x, y$-Schnitt durch das DSI bei der Disparität $d = 108$. (d) $x, y$-Schnitt durch das DSI bei der Disparität $d = 116$. Gesichtregionen werden bei höherer Disparität geringere Kosten zugeordnet. (e)&(f) Vergrößerte $x, d$-Schnitte durch das DSI der Zeilen $y = 240$ und $y = 360$ (rote Zeilen in (a)). Die jeweiligen Pfade der geringsten Kosten und somit die Repräsentationen der Form der Oberflächen sind als horizontale dunkle Streifen zu erkennen.

### 2.2.4. Zuordnungskosten

In den letzten Jahren haben sich verschieden oft verwendete Funktionen für die Berechnung von Zuordnungskosten etabliert. Die wichtigsten Maße seien an dieser Stelle kurz genannt. Einfache und schnell berechenbare Kostenmaße sind die Absolute Intensitätsdifferenz (AD) [Kanade u. a., 1995] sowie die quadrierte Intensitätsdifferenz (*squared difference*-SD) [Anandan, 1989]. Sie werden als *parametrische Maße* bezeichnet. Sie sind empfindlich gegenüber Bildunterschieden wie Rauschen oder unterschiedlichen Helligkeiten von linkem und rechtem Bild. Gegenüber (gaußschem) Rauschen zeigt sich die insbesondere die normalisierte Kreuzkorrelation (NCC) unempfindlicher. Sie vereint durch ihr Prinzip bereits Eigenschaften der Kostenbestimmung und Aggregation. Birchfield und Tomasi haben ein Kostenmaß eingeführt, dass auf Subpixelebene arbeitet und das aktuelle Pixel mit einem interpolierten Wert aus dem anderen Bild vergleicht. Es liefert in Bereichen mit Tiefensprüngen bessere Ergebnisse [Birchfield und Tomasi, 1998].

Diesen gegenüber stehen *nicht parametrische* Kostenmaße. Sie transformieren die Intensitätswerte und bestimmen die Kosten anhand der transformierten Signale. Oft verwendete Maße sind gradientenbasierte Methoden [Scharstein, 1994] sowie die Rang- uns Censustransformationen [Zabih und Woodfill, 1994] [Cyganek und Siebert, 2009, S. 209ff]. Sie führen die Kostenberechnung nicht

| Name | Typ | Term |
|---|---|---|
| Absolute Intensitätsdifferenz | parametrisch | $C_{AD}(\mathbf{p}, \hat{\mathbf{p}}) = \lvert I_L(\mathbf{p}) - I_R(\hat{\mathbf{p}})\rvert$ |
| Quadrierte Intensitätsdifferenz | parametrisch | $C_{SD}(\mathbf{p}, \hat{\mathbf{p}}) = (I_L(\mathbf{p}) - I_R(\hat{\mathbf{p}}))^2$ |
| Normalisierte Kreuzkorrelation | parametrisch | $C_{NCC}(\mathbf{p}, \hat{\mathbf{p}}) = \frac{\sum_{\mathbf{q}\in N_\mathbf{p}} I_L(\mathbf{q}) I_R(\hat{\mathbf{q}})}{\sqrt{\sum_{\mathbf{q}\in N_\mathbf{p}} I_L(\mathbf{q})^2 \sum_{\mathbf{q}\in N_\mathbf{p}} I_R(\hat{\mathbf{q}})^2}}$ |
| Rangtransf. (meist mit AD) | nicht-parametrisch | $I_{Rg}(\mathbf{p}) = \sum_{\mathbf{q}\in N_\mathbf{p}} T[I(\mathbf{q}) < I(\mathbf{p})]$ |
| Censustransf. (Hamming Distanz) | nicht-parametrisch | $I_{Ce}(\mathbf{p}) = \bigotimes_{\mathbf{q}\in N_\mathbf{p}} T[I(\mathbf{q}) < I(\mathbf{p})]$ |
| | | $T[v] = \begin{cases} 1 & \text{wenn v=wahr} \\ 0 & \text{wenn v=falsch} \end{cases}$ |
| | | $\bigotimes$ - Aneinanderreihung |

**Tabelle 2.2.:** Verschieden Kostenmaße und ihre Berechnung im Überblick. Zur einfacheren Darstellung wurde $C(x, y, d)$ durch $C(\mathbf{p}, \hat{\mathbf{p}})$ ersetzt. $\mathbf{p}$ und $\mathbf{q}$ stehen für Pixel eines Bildes $I$. $\hat{\mathbf{p}} = \mathbf{p} - \mathbf{d}$ und $\hat{\mathbf{q}} = \mathbf{q} - \mathbf{d}$ mit $\mathbf{d} = [d\,0]^\top$. $N_\mathbf{p}$ ist die Nachbarschaft von $\mathbf{p}$, meist ein quadratisches Fenster.

auf den Intensitätswerten, sondern auf deren Transformierten aus. Bei der Rangtransformation ergibt sich der transformierte Wert aus der Anzahl der Pixel, die kleiner als das gerade betrachtete Pixel innerhalb des Fensters sind. Die Censustransformation erzeugt eine Bit-Kette aus diesem Vergleich. Die Zuordnungkosten ergeben sich dann aus der Hamming-Distanz zweier Bit-Ketten des linken und rechten Bildes. Nicht parametrische Methoden sind unempfindlicher gegen Intensitätsschwankungen als parametrische. Eine Übersicht ausgwählter Maße für Zuordnungskosten findet sich in Tabelle 2.2. Ein aktueller Vergleich ist in [Hirschmüller und Scharstein, 2009] zu finden.

### 2.2.5. Lokale Stereoanalysealgorithmen

Lokale Stereoanalysealgorithmen bestimmen die Disparität für ein bestimmtes Pixel nur durch Miteinbeziehung von dessen näherer Umgebung - des *Fensters* oder auch der *Support-Region*. Die Nutzung des Fensters dient der Erhöhung der Robustheit der Bestimmung und entspricht implizit der Annahme der Glattheitsbedingung. Als Beispiel für einen lokalen Algorithmus sei die Verwendung der absoluten Intensitätsdifferenzen (AD) als Kostenfunktion genannt. In Schritt eins werden für jedes Pixel für jede mögliche Disparität die absoluten Intensitätsdifferenzen gebildet (vgl. Tabelle 2.2). Die anschließende Aggregation wird durch Aufsummierung dieser

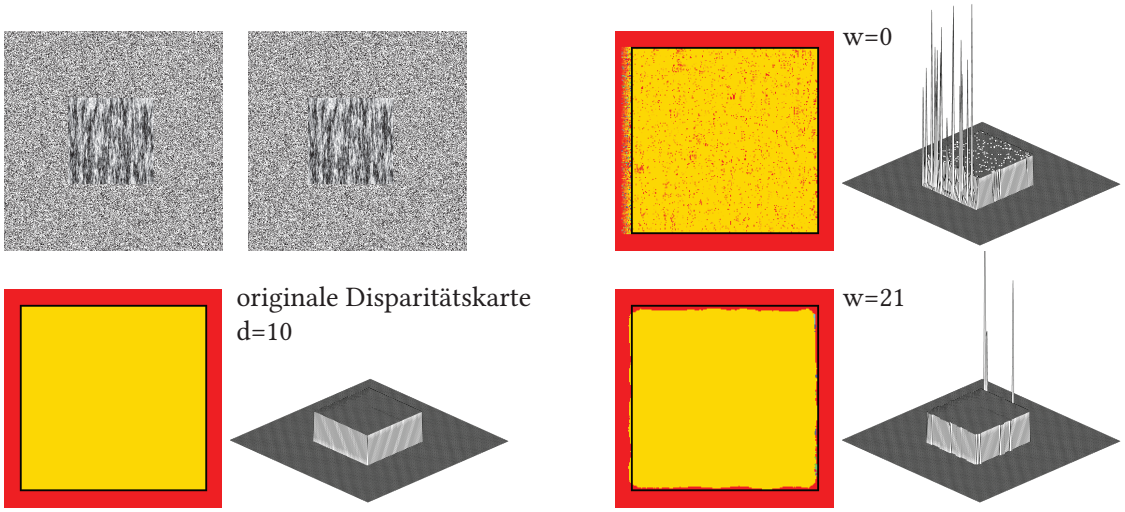



**Abbildung 2.18.:** Auswirkungen der Fenstergröße bei lokalen Stereoanalysealgorithmen. Dargestellt sind das Eingangs-Stereopaar sowie vergrößerte Falschfarbenausschnitte der Disparitätskarten und ein Plot der Oberfläche. Ohne Aggregation ($w = 0$) führen Mehrdeutigkeiten und Verdeckungen zu Fehlschätzungen. Die Kontur des Objektes (schwarzer Rahmen) ist jedoch gut erhalten. Große Fenster ($w = 21$) erzeugen innerhalb des Objektes kaum Fehler, im Randbereich kommt es jedoch zu Vordergrunderweiterungen und -reduzierungen.

Werte über eine quadratische Fensterfunktion realisiert. Dieser Vorgang wird mathematisch als zweidimensionale Faltung der $d$-Ebenen des Kostenraumes dargestellt. Werden beide Schritte zusammengefasst, wird oft von SAD (*Sum of Squared Differences*) gesprochen. Die Bestimmung der besten Zuordnung und somit die Wahl der Disparität erfolgt durch die Suche nach den geringsten Kosten für jedes Pixel $x, y$ entlang der $D$-Dimension des Kostenraumes, was häufig als *winner-takes-all* – WTA bezeichnet wird. Ist das Disparitätsbild gefunden, so kann es abschließend verfeinert werden. Übliche Methoden sind die Anwendung von 2D-Filtern (insbesondere des Median-Filters) zur Beseitigung von „Ausreißer"-Disparitäten oder die Interpolation fehlender Disparitäten, z. B. in verdeckten Bereichen. Mit steigender Fenstergröße werden lokale Algorithmen robuster gegenüber Ausreißern, die u. a. durch Rauschen und Mehrdeutigkeiten entstehen können. Jedoch werden dadurch auch Tiefensprünge an Objektgrenzen verwischt bzw. Objekte im Vordergrund in der Disparitätskarte über ihren Begrenzung hinaus vergrößert was auch als *foreground fattening* bezeichnet wird (vgl. Abbildung 2.18). Um diesen Effekt zu verringern, werden z. B. adaptive Fenster eingesetzt, deren Form und Größe z. B. durch Kantendetektoren gesteuert wird [Okutomi und Kanade, 1992; Kanade und Okutomi, 1994; Kang u. a., 2001; Wang, 2004]. Ein weiterer populärer Ansatz ist die Verwendung adaptiver Wichtung bei der Aggregation [Yoon und Kweon, 2005, 2006, 2007]. Abgeleitet aus den Gesetzen der Nähe und der Ähnlichkeit der Gestaltpsychologie wird für jedes Aggregationsfenster ein separater Filterkernel berechnet. Die Filterkoeffizienten sind umso höher, je näher (euklidischer Abstand

zum Aufpunkt) und je ähnlicher (euklidischer Abstand im CIE-Lab Farbraum) das jeweilige Pixel im Fenster dem Aufpunkt ist. Dies führt insbesondere an Objektkanten und somit an Tiefensprüngen zu besseren Ergebnissen. Abbildung 2.19 veranschaulicht die Wichtung an einem Beispiel. Die Berechnungsvorschrift nach [Yoon und Kweon, 2006] lautet unter Nutzung der Notation in Tabelle 2.2:

$$C_{ASW}(\mathbf{p}, \hat{\mathbf{p}}) = \frac{\sum_{\mathbf{q} \in N_{\mathbf{p}}, \hat{\mathbf{q}} \in N_{\hat{\mathbf{p}}}} w(\mathbf{p}, \mathbf{q}) w(\hat{\mathbf{p}}, \hat{\mathbf{q}}) c(\mathbf{q}, \hat{\mathbf{q}})}{\sum_{\mathbf{q} \in N_{\mathbf{p}}, \hat{\mathbf{q}} \in N_{\hat{\mathbf{p}}}} w(\mathbf{p}, \mathbf{q}) w(\hat{\mathbf{p}}, \hat{\mathbf{q}})} \tag{2.42}$$

mit

$$w(\mathbf{p}, \mathbf{q}) = e^{-\frac{\Delta c_{\mathbf{pq}}}{\gamma_c}} \cdot e^{-\frac{\Delta g_{\mathbf{pq}}}{\gamma_p}} \tag{2.43}$$

$\Delta c$ und $\Delta g$ sind die Maße für Farbähnlichkeit bzw. örtliche Nähe, $\gamma$ die Parameter, die die Steilheit des Filters bestimmen. Die Farbähnlichkeit wird aus dem euklidischen Abstand im CIE-Lab Farbraum bestimmt, da dieser Abstand dort der Wahrnehmung von Unterschieden beim Menschen (weitestgehend) entspricht:

$$\Delta c_{\mathbf{pq}} = \sqrt{(L_{\mathbf{p}} - L_{\mathbf{q}})^2 + (a_{\mathbf{p}} - a_{\mathbf{q}})^2 + (b_{\mathbf{p}} - b_{\mathbf{q}})^2} \tag{2.44}$$

Die Koeffizienten für die Nähe werden aus dem euklidischen Abstand zum Aufpunkt berechnet:

$$\Delta g_{\mathbf{pq}} = \sqrt{(x_{\mathbf{p}} - x_{\mathbf{q}})^2 + (y_{\mathbf{p}} - y_{\mathbf{q}})^2} \tag{2.45}$$

$c(\mathbf{q}, \hat{\mathbf{q}})$ ist ein Kostenmaß. In [Yoon und Kweon, 2006] wurde die begrenzte absolute Differenz (*Truncated Absolute Difference*-TAD) über alle RGB-Kanäle gewählt. Nachteil der ASW-Methode ist ein wesentlich erhöhter Berechnungs- und Speicheraufwand. Fehlschätzungen können bei sich wiederholenden Mustern durch einen dann sehr geringen Support-Bereich auftreten, was u. a. in [Tombari u. a., 2007] untersucht wurde.

### 2.2.6. Globale Stereoanalysealgorithmen

Globale Methoden modellieren Zusammenhänge zwischen den Pixeln der Eingangsdaten und leiten daraus die Tiefe der Szene ab (*inference*). Ziel ist es, mittels des Modells jedem Pixel des Bildes ein bestimmtes *Label* aus einer Menge von Labeln zuzuweisen (*labeling*). Im Fall der Stereoanalyse ist dies die Menge aller möglichen Disparitäten.

Dieses *Labeling* wird formal mittels *Sites*[4] und Labeln und deren Mengen spezifiziert [Li, 2009]. Es sei $\mathcal{S} = \{1, \ldots, m\}$ die Menge aller Sites, welche im konkreten Fall Bildpixeln oder Bildregionen entsprechen, wobei $\{1, \ldots, m\}$ Indizes sind. Unter der Annahme von Pixeln in einem Bild der

[4] auf eine Übersetzung wird an dieser Stelle bewusst verzichtet

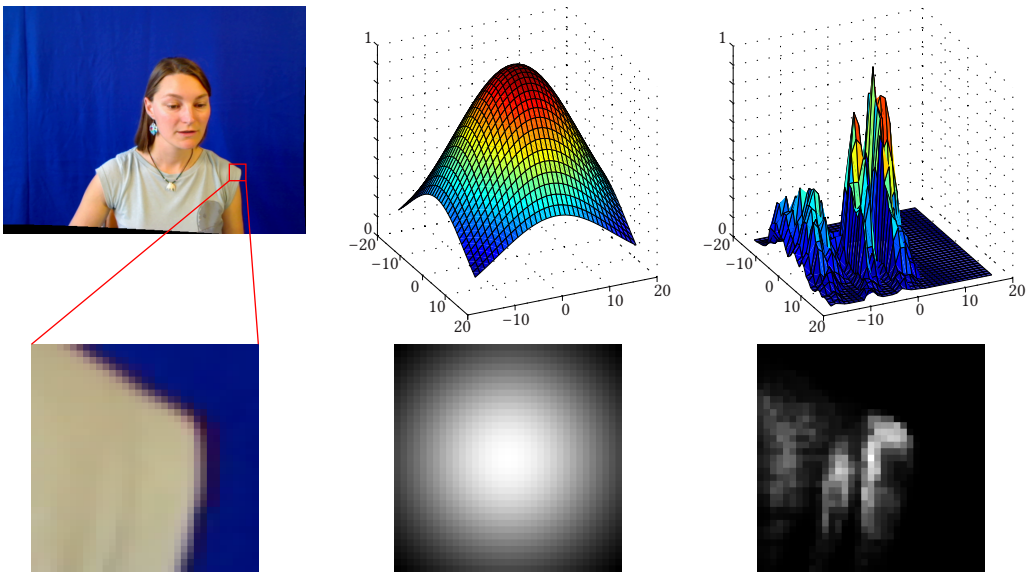



**Abbildung 2.19.:** Visualisierung der ASW-Wichtungskoeffizienten des links abgebildeten Fensters für die korrekte Disparität des Vordergrundens. Allein die Wichtung entsprechend des Abstandes (Mitte) bezöge den blauen Hintergrund mit ein. Eine adaptive Wichtung beschränkt die hohen Filterkoeffizienten auf ähnliche Bereiche der Schulter (rechts). Fenstergröße $35 \times 35$, $d = 105$, $\gamma_c = 5$ und $\gamma_p = 17.5$.

Größe $n \times n$ sind diese „örtlich regulär“ auf einem Gitter angeordnet.

$$\mathcal{S} = \{x, y | 1 \leq x, y \leq n\} \tag{2.46}$$

Im Folgenden wird ein eindimensionaler Index verwendet. Die Menge aller möglichen Label wird mit $\mathcal{L}$ bezeichnet. Bei der Stereoanalyse handelt es sich um die diskrete Menge aller möglichen Disparitäten $\mathcal{L} = \{d_{min}, \ldots, d_{max}\}$ Mit diesen Definitionen lässt sich das Labeling-Problem mathematisch beschreiben. Die Menge

$$f = \{f_1, \ldots, f_m\} \tag{2.47}$$

ist ein bestimmtes Labeling, also die Zuweisung eines Labels zu jedem Pixelindex. Dies kann auch als Funktion $f_i = f(i)$ aufgefasst werden, deren Bildmenge $\mathcal{L}$ und deren Definitionsmenge $\mathcal{S}$ ist. Die Funktion ist eine Abbildung $f : \mathcal{S} \longmapsto \mathcal{L}$. Die Menge aller möglichen Labelings wird als Konfigurationsraum bezeichnet und ergibt sich zu $\mathbb{F} = \mathcal{L}^m$. Abbildung 2.20 veranschaulicht den Sachverhalt für ein konkretes Beispiel. Um aus dieser hohen Anzahl möglicher Konfigurationen die gesuchte bzw. optimale zu finden, bedarf es

- eines Modells zur Erfassung der Eingangsdaten und quantitativen Bewertung der jeweiligen Konfiguration,

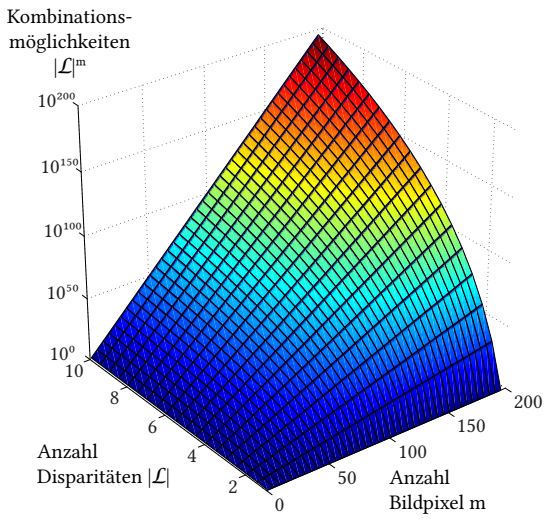



**Abbildung 2.20.:** Visualisierung der Mächtigkeit der Lösungsmenge eines Konfigurationsraumes $\mathbb{F}$ für 200 Pixel und 10 mögliche Disparitäten. Es ist ersichtlich, dass die mögliche Anzahl von Labelings allein für dieses Beispiel sehr hoch ist.

- kontextueller Randbedingungen, die die rechnerisch unmögliche Suche nach dem Optimum durch Ausprobieren vereinfachen und sich durch das Modell darstellen lassen.

Ein beliebtes graphisches Modell zur Lösung eines solchen Problems sind paarweise Markowsche Zufallsfelder (*Markov Random Fields* - MRF). Die zentrale Idee von MRFs ist die Nutzung bedingter Wahrscheinlichkeiten und die Einbeziehung der unmittelbaren Nachbarschaft der Sites eines Systems.

Es folgt die formale Definition unter der Nutzung der Notation aus [Li, 2009]. Es sei $F = \{F_1, \ldots, F_m\}$ eine Familie von Zufallsvariablen, definiert über die Menge aller Sites $\mathcal{S}$ (die Indizes aller Bildpixel). Jede Zufallsvariable $F_i$ kann einen Wert aus der Menge $\mathcal{L}$ annehmen. $F$ wird als Zufallsfeld bezeichnet. Das Ereignis, dass $F_i$ einen Wert $f_i$ annimmt wird mit $F_i = f_i$ geschrieben, das Verbundereignis $(F_1 = f_1, \ldots, F_m = f_m)$ kurz als $F = f$. $f$ ist entsprechend Gleichung (2.47) eine Konfiguration von $F$. Die Modellierung von MRFs bedient sich der bayesschen Statistik. Die Wahrscheinlichkeit repräsentiert hierbei den Grad der „Glaubwürdigkeit“ eines Ereignisses und nicht, wie in der frequentistischen Wahrscheinlichkeitsauffassung dessen relative Häufigkeit. Unter der Verwendung einer diskreten Menge von Labels $\mathcal{L}$ wird die Wahrscheinlichkeit des Ereignisses $F_i = f_i$ als $P(f)$ geschrieben, die Verbundwahrscheinlichkeit als $P(f)$.

Unter der Annahme von stochastisch unabhängigen Sites ergibt sich die Verbundwahrscheinlichkeit einer bestimmten Konfiguration aus dem Produkt der einzelnen Label-Wahrscheinlichkeiten

[Li, 2009, S. 7].

$$P(f) = \prod_{i \in \mathcal{S}} P(f_i) \tag{2.48}$$

Entsprechend der Idee der Einbeziehung anderer Labels wird der Gedanke der gegenseitigen Unabhängigkeit aufgehoben. Die Wahrscheinlichkeit einer bestimmen Konfiguration wird nun als Produkt von bedingten Wahrscheinlichkeiten ausgedrückt:

$$P(f) = \prod_{i \in \mathcal{S}} P(f_i | f_{\mathcal{S}-\{i\}}) \tag{2.49}$$

$P(f_i|f_{\mathcal{S}-\{i\}})$ ist die bedingte Wahrscheinlichkeit für Label $f_i$ in Abhängigkeit der Labels aller anderen Sites außer der gerade betrachteten $\mathcal{S} - \{i\}$. Eine direkte Lösung verbietet sich auch hier, da die Berechnung aller Kombinationen von Verbundwahrscheinlichkeiten noch immer zu aufwändig wäre und zudem ein Konstrukt fehlt, diese Wahrscheinlichkeiten anhand von Beobachtungen zu berechnen. Weitere Randbedingungen sind notwendig [Li, 2009, S. 25]. Die Markow-Eigenschaft ist eine solche Randbedingung:

$$P(f_i|f_{\mathcal{S}-\{i\}}) = P(f_i|f_{\mathcal{N}_i}) \tag{2.50}$$

Sie beschränkt die bedingte Wahrscheinlichkeit auf die direkte Nachbarschaft $\mathcal{N}_i$ einer Site. Eine weitere Eigenschaft ist die Positivität:

$$P(f) > 0, \forall f \in \mathbb{F} \tag{2.51}$$

Erfüllt das Zufallsfeld $F$ die Eigenschaften (2.50) und (2.51) wird es als *Markowsches Zufallsfeld* bezeichnet. Ziel ist nun die Maximierung der Verbundwahrscheinlichkeit, und somit die Bestimmung der Konfiguration, die zu dieser Verbundwahrscheinlichkeit führt.

Die Nachbarschaft in MRFs wird formal mittels so genannter Cliquen definiert. Um den Rechenaufwand gering zu halten, wird zumeist nur die unmittelbare Nachbarschaft der aktuellen Site (eines Pixels) betrachtet. Abbildung 2.21 zeigt das Beispiel einer Vierer-Nachbarschaft (auch Nachbarschaft erster Ordnung genannt). Aus der Nachbarschaft ergibt sich die Menge möglicher Cliquen. Im Beispiel einer Vierer-Nachbarschaft sind das die einzelne Clique definiert durch

$$\mathcal{C}_1 = \{i | i \in \mathcal{S}\} \tag{2.52}$$

bzw. die paarweisen Cliquen:

$$\mathcal{C}_2 = \{\{i, i'\} | i \in \mathcal{S}, i' \in \mathcal{N}_i\} \tag{2.53}$$

mit der Gesamtmenge für alle Cliquen des gewählten Beispiels $\mathcal{C} = \mathcal{C}_1 \cup \mathcal{C}_2$. Es stellt sich nun die

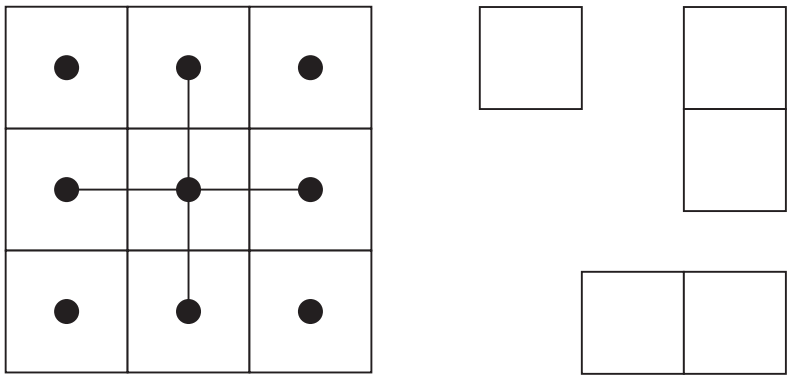



**Abbildung 2.21.:** Die in (a) dargestellte Vierer-Nachbarschaft ermöglicht die drei Cliquen in (b).

Frage, wie die Wahrscheinlichkeiten durch lokale berechenbare Funktionen ausgedrückt werden können. Ein Ansatz ist, die Äquivalenz zwischen MRFs und Gibbs-Zufalls-Feldern zu nutzen, welche durch das Hammersley-Clifford-Theorem gegeben ist. Gibbs-Zufalls-Felder sind durch eine globale Eigenschaft, die Gibbs-Verteilung, bestimmt [Li, 2009; Hörchens, 2005]:

$$P(f) = \frac{1}{Z}\, e^{-\frac{1}{T}U(f)} \tag{2.54}$$

$T$ wird als Temperatur[5] bezeichnet und in den betrachteten Anwendungen mit 1 angenommen. $Z$ ist hierbei die *Partitionsfunktion*, $U(f)$ wird als *Energiefunktion* bezeichnet. Der Zusammenhang wurde erstmals von Geman & Geman [Geman und Geman, 1984] im Kontext der Bildverarbeitung genutzt. Die Partitionsfunktion ist normalisierend und ergibt sich aus

$$Z = \sum_{f \in \mathbb{F}} e^{-\frac{1}{T}U(f)} \tag{2.55}$$

Die Energiefunktion

$$U(f) = \sum_{c \in \mathcal{C}} V_c(f) \tag{2.56}$$

ist die Summe aller möglichen Cliquenpotentiale $V_c(f)$ über die Menge der möglichen Cliquen $\mathcal{C}$ der Nachbarschaft $\mathcal{N}_i$. Mittels dieser Formulierung lassen sich die im MRF modellierten Wahrscheinlichkeiten durch die Energiefunktion quantifizieren. Aus (2.54) wird ersichtlich, dass die Energie minimiert werden muss, um die Verbundwahrscheinlichkeit $P(f)$ zu maximieren. Es fehlt jedoch noch die Einbeziehung der Beobachtung in das Konstrukt. Hierzu wird sich eines Bayesschen Schätzers, meist des MAP-Schätzers (Maximum A-posteriori) bedient.

[5] Die Gibbs-Boltzmann-Verteilung hat Ihren Ursprung in der Thermodynamik.

Gesucht ist die Wahrscheinlichkeit für ein bestimmtes Labeling $f$ unter einer Beobachtung $b$. Sie wird *a-posteriori Wahrscheinlichkeit* genannt und als $P(f|b)$ geschrieben. Nach dem Bayes-Theorem lässt sich diese mittels

$$P(f|b) = \frac{P(b|f)P(f)}{P(b)} \tag{2.57}$$

ausdrücken. $P(f)$ ist die *A-priori-Wahrscheinlichkeit* für ein Labeling $f$ vor der Beobachtung. Die *bedingte Wahrscheinlichkeit* $P(b|f)$ wird *likelihood* genannt und beschreibt die Wahrscheinlichkeit der beobachteten Daten unter der Annahme, dass $f$ eingetreten ist. $P(b)$ beschreibt die Wahrscheinlichkeit für die Beobachtung $b$ selbst und ist konstant, da $b$ konstant ist.

Diese Wahrscheinlichkeit gilt es zu maximieren. Entsprechend des MAP-Schätzers ergibt sich das wahrscheinlichste Labeling zu

$$f* = \underset{f \in \mathbb{F}}{\operatorname{argmax}}\, P(f|b) \tag{2.58}$$

Zur genauen Herleitung dieses und anderer Schätzer sei auf [Li, 2009] verwiesen. Unter der Annahme einer konstanten Beobachtung $b$ kann der Nenner in (2.57) weg gelassen werden und es gilt

$$f* = \arg\max_{f \in \mathbb{F}} \{P(b|f)P(f)\} \tag{2.59}$$

Unter der Hinzunahme von (2.54) lassen sich die Wahrscheinlichkeiten im Modell eines MRFs durch die Energiefunktionen quantifizieren:

$$P(f) = \frac{1}{Z}\, e^{-U(f)} \tag{2.60}$$

und

$$P(b|f) = \frac{1}{Z}\, e^{-U(b|f)} \tag{2.61}$$

$U(f)$ ist die A-priori-Energie und $U(b|f)$ die Likelihood-Energie. Entsprechend eingesetzt ergibt sich der Zusammenhang

$$P(f|b) \propto e^{-U(f|b)} \tag{2.62}$$

mit

$$U(f|b) = U(b|f) + U(f) \tag{2.63}$$

Ziel aller globalen Stereoanalysealgorithmen ist es nun, diese Energiefunktionen entsprechend des MRF-Modells geeignet zu spezifizieren und mittels verschiedener Algorithmen ein angenähertes Optimum zu finden, indem die Energie minimiert wird (wodurch $P(f|b)$ maximiert wird). Da $U(f)$ benachbarte Beziehungen quantifiziert, wird sie oft als Glattheitsterm bezeichnet, während $U(b|f)$ die Beziehung des Labelings zu den beobachteten Daten herstellt und daher als Datenterm bezeichnet wird. Die Nachbarschaft für den Datenterm wird meist auf die Site

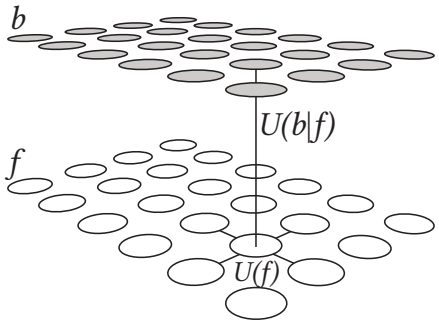



**Abbildung 2.22.:** Die Beziehung zwischen Beobachtung $b$ und unbekanntem Zustand (Labeling) $f$ wird durch die Energiefunktionen hergestellt, die es zu minimieren gilt. $U(b|f)$ bewertet die Beziehung zwischen Beobachtung und aktuellem Label (im Bsp. eine $\mathcal{C}_0$-Clique), $U(f)$ quantifiziert den Zusammenhang innerhalb der Nachbarschaft von Labels (im Bsp. eine $\mathcal{C}_1$-Clique).

selbst beschränkt, jene für den Glattheitsterm auf dessen Vierer-Nachbarschaft so dass für dieses Beispiel gilt:

$$U(f|b) = \beta \sum_{c \in \mathcal{C}_0} V_0(b, f) + \gamma \sum_{c \in \mathcal{C}_1} V_1(f) \tag{2.64}$$

wobei $\beta$ und $\gamma$ eine Wichtung ermöglichen. Abbildung 2.22 veranschaulicht den Sachverhalt.

Die in der Literatur auf die Stereoanalyse konkretisierte und gängigere Schreibweise für die Energiefunktion folgt der Notation in [Scharstein und Szeliski, 2002] und lautet

$$E(d) = E_{daten}(d) + \lambda E_{glatt}(d) \tag{2.65}$$

mit $d$ als Disparitätsfunktion, welche einer Konfiguration $f$ in obigen Auführungen entspricht. $E_{daten}$ entspricht $U(b|f)$ und $E_{glatt}$ ist das Äquivalent zu $U(f)$.

Der Datenterm $E_{Daten}$ misst die Übereinstimmung mit den Eingangsstereobildern und nutzt dafür üblicherweise die im Abschnitt 2.2.5 durch das DSI $C(x, y, d)$ repräsentierten Kostenmaße [Scharstein und Szeliski, 2002]:

$$E_{daten}(d) = \sum_{(x,y)} C(x, y, d(x, y)) \tag{2.66}$$

Der Glattheitsterm lässt sich z. B. schreiben als [Scharstein und Szeliski, 2002]:

$$E_{glatt}(d) = \rho(d(x, y) - d(x + 1, y)) + \rho(d(x, y) - d(x, y + 1)) \tag{2.67}$$

wobei hier eine sehr kleine Nachbarschaft genutzt wurde. $\rho$ ist hier eine monoton steigende Funktion von Disparitätsdifferenzen. Der Anspruch bei der Formulierung des Glattheitsterms liegt darin, ihn so zu formulieren, dass einerseits eine „glatte“ Disparitätskarte entsteht, andererseits

jedoch an Diskontinuitäten die Tiefengrenzen nicht „verwaschen“ werden [Scharstein und Szeliski, 2002]. So wurde bereits in [Geman und Geman, 1984] eine Diskontinuitäten-erhaltende Energiefunktion im Kontext der Bildrestauration formuliert. Die Einbeziehung der Intensitäten der Beobachtung in den Glattheitsterm ist eine weitere Idee:

$$E_{glatt}(d) = \rho_d(d(x,y) - d(x+1,y)) \cdot \rho_I(||I(x,y) - I(x+1,y||) \tag{2.68}$$

Beispiele für eine derartige Verwendung finden sich in [Bobick und Intille, 1999; Gamble und Poggio, 1987; Fua, 1993; Boykov u. a., 2001].

Das Finden der minimalen Energie durch Ausprobieren aller Möglichkeiten ist derzeit rechnerisch unmöglich. Approximierende Algorithmen zur Bestimmung eines angenäherten aber nicht zwingenderweise globalen Minimums sind beispielsweise ICM (Iterative Conditional Modes) [Besag, 1986], *(Loopy) Belief-Popagation* [Pearl, 1988; Sun u. a., 2003; Felzenszwalb und Huttenlocher, 2006] oder *Graph Cuts* [Kolmogorov und Zabih, 2001; Boykov u. a., 2001; Kolmogorov und Zabih, 2002, 2004]. Sie unterscheiden sich in der Modellierung der Sites und der Art und Weise, wie die Labels geändert/aktualisiert bzw. wie die kumulativen Nachbarschaftsenergien propagiert werden. Während insbesondere ICM-Methoden, aber auch Belief-Propagation auf ein gutes intitiales Labeling angewiesen sind [Li, 2009, S. 248], so hat das initiale Labeling auf das Ergebnis bei Graph Cuts kaum Einfluss [Boykov u. a., 2001]. Für einen kurzen Vergleich von Belief Propagation und Graph Cuts sei auf [Tappen und Freeman, 2003] verwiesen.

**Dynamische Programmierung:** Eine teilglobale Methode ist die Disparitätsschätzung mittels Dynamischer Programmierung (DP). Das Verfahren basiert auf der Eindeutigkeitsannahme und Reihenfolgeneinschränkung. Auch dieses Problem kann als Minimierung einer Energiefunktion beschrieben werden. Anstelle des gesamten zweidimensionalen Bildraumes bzw. des dreidimensionalen Kostenraumes (DSI) wird die Optimierung jeweils auf den korrespondierenden Bildzeilen rektifizierter Stereobilder durchgeführt. Diese spannen ihrerseits eine Ebene im Disparitätsraum auf. Anschaulich entspricht dies der zeilenweisen Suche nach dem optimalen Pfad durch eine (neu angeordnete) DSI-Ebene wie Abb. 2.23 zeigt. Zunächst allein für Kanten durchgeführt (vgl. [Baker und Binford Thomas, 1981; Ohta und Kanade, 1985]) fand das Verfahren auch Anwendung zur Erzeugung dichter Disparitätskarten [Belhumeur und Mumford, 1992; Geiger u. a., 1995; Belhumeur, 1996; Birchfield und Tomasi, 1999]. In jüngerer Vergangenheit wurden Verfahren mit mehreren Zuständen im Kontext Videokommunikation [Criminisi u. a., 2003, 2007] oder in Kombination mit adaptiver Wichtung [Wang u. a., 2007] vorgestellt. Fehlschätzungen der DP-Verfahren machen sich als typische Streifen bemerkbar, die über Objektgrenzen mit Tiefensprüngen hinaus propagiert werden. Auch sind oft Disparitätsunterschiede zwischen einzelnen Zeilen zu beobachten, wenn der Algorithmus diese Nachbarschaften nicht mit einbezieht (vgl. Abb. 2.24).

Das Prinzip der dynamischen Programmierung basiert auf der Zerlegung eines Gesamtpro-

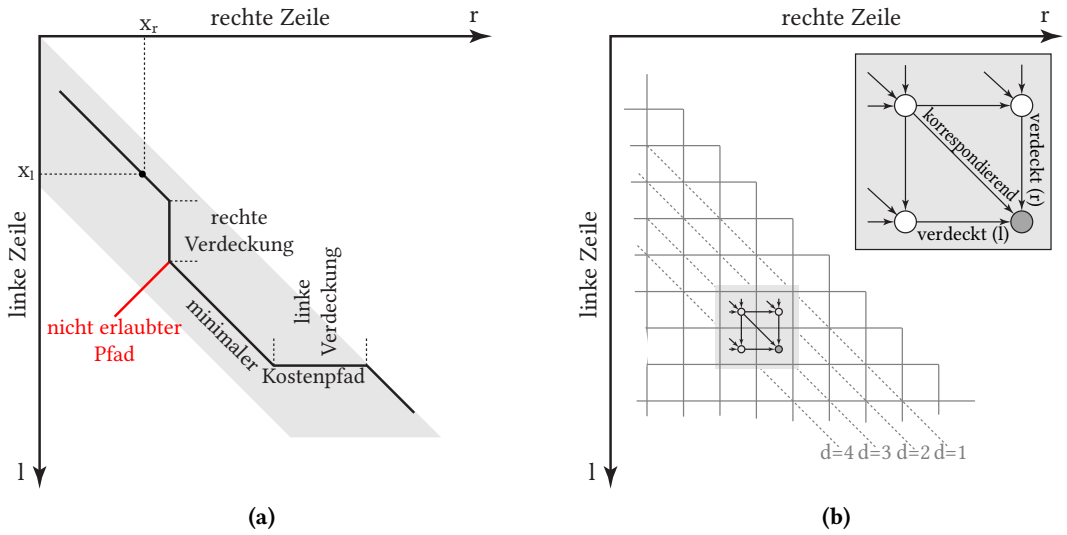



**Abbildung 2.23.:** Schematische Darstellung der einfachen Dynamischen Programmierung. (a) Möglicher Kostenpfad. Disparitätssprünge werden als linke bzw. rechte Verdeckung interpretiert. (b) Darstellung der drei möglichen Transfers zwischen den Elementen der Kostenebene.

blems (Finden des optimalen Pfades durch den Kostenraum) in Teilprobleme. Die durch die jeweiligen Zeilen aufgespannte Kostenebene wird in kleinen Schritten (Transfer) durchlaufen, wobei die Kosten kumuliert werden und sich der jeweils günstigste Schritt „gemerkt wird“. Im ursprünglichen Verfahren nach Cox [Cox u. a., 1996] wird für jedes Element des Kostenraumes ein Kostenwert berechnet, der sich aus den Kosten dreier benachbarter Elemente ergibt. Die schematische Darstellung eines möglichen Pfades zeigt Abb. 2.23. Die Reihenfolgeneinschränkung verbietet dabei bestimmte Richtungen des Pfades wie in der Abbildung als Beispiel rot markiert. Außerdem beschränkt sie den Suchbereich auf die untere Hälfte der Kostenmatrix. Die initialen Kosten $M(x_l, x_r)$ werden mit einem der bekannten Kostenmaße (vgl. Tabellen 2.2) bestimmt. Die kumulierten Kosten werden beispielsweise wie folgt berechnet:

$$C(x_l, x_r) = min \begin{cases} C(x_l - 1, x_r) & + c_{ver} \\ C(x_l - 1, x_r - 1) & + M(x_l, x_r) \\ C(x_l, x_r - 1) & + c_{ver} \end{cases} \tag{2.69}$$

Die Akkumulation der Kosten nach dieser Gleichung wird auch als *Vorwärts-Durchlauf* bezeichnet, bei dem der jeweils minimale Transfer-Schritt gespeichert wird. $c_{ver}$ ist eine im Vergleich zu den realen Kosten meist höher gewählte Konstante und symbolisiert die Kosten innerhalb eines verdeckten Bereiches. Der „Rückwärts-Durchlauf“ folgt den gespeicherten Schritten, wodurch der minimale Kostenpfad erzeugt wird.

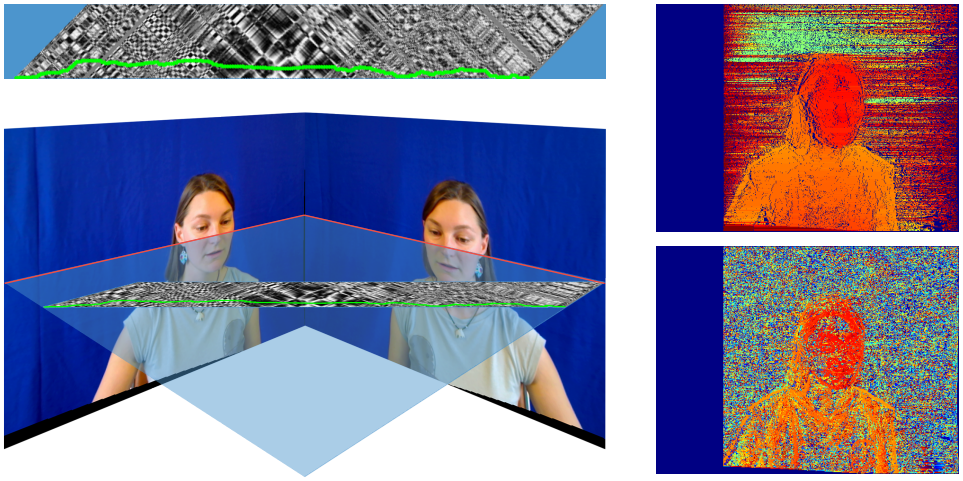


**Abbildung 2.24.:** Stereoanalyse mittels Dynamischer Programmierung. Für jede Zeile (im Beispiel Zeile 232) wird der optimale Pfad (grün) durch den Kostenraum ermittelt. Das Ergebnis mit den typischen DP-Artefakten ist r. o. dargestellt. Im Vergleich dazu die WTA-Disparitätskarte mit gleichem Kostenmaß und derselben Aggregation r. u. (3x3 NCC mit 3x3 Gauß-Aggregation).

Es ist offensichtlich, dass Gleichung (2.69) die Glattheitsannahme formuliert, indem Disparitätsänderungen mittels eines konstanten Terms „bestraft“ werden, was in realen Szenen natürlich selten der Fall ist. Sie werden von vornherein als eine Verdeckung betrachtet. Daher wurde das Modell in [Criminisi u. a., 2003] um weitere mögliche Schritte ergänzt. Dabei wir u. a. zwischen der Änderung der Disparität durch eine Übereinstimmung bzw. durch eine Verdeckung unterschieden. Dies führt zur realistischeren Disparitätskarten und besserer Verdeckungserkennung, andererseits erhöht sich der Berechnungsaufwand.

## 2.3. Bildbasiertes Rendering

Die Synthese neuer Ansichten zur Blickkorrektur ist das finale Ziel des in dieser Arbeit entwickelten Verfahrens. Dazu werden Algorithmen aus dem Bereich des bildbasierten Renderings genutzt. Die Definition des bildbasierten Renderings (*image based rendering*) ist nicht immer einheitlich in der Forschungsgemeinschaft. Die vorliegende Arbeit folgt der Systematik von Shum et. al [Shum u. a., 2003, 2007] und definiert den Begriff wie folgt: Bildbasiertes Rendering bezeichnet die Erzeugung neuer Ansichten einer dreidimensionalen Szene auf Basis einer oder mehrerer Bildaufnahmen dieser Szene.

Allen Verfahren ist gemein, dass sie Methoden beschreiben, die die dreidimensionale Struktur einer Szene aus Kameraaufnahmen extrahieren und in einer bestimmten Repräsentationsform speichern. Dabei lassen sich drei große Kategorien unterteilen, deren Unterschiede in der Anzahl

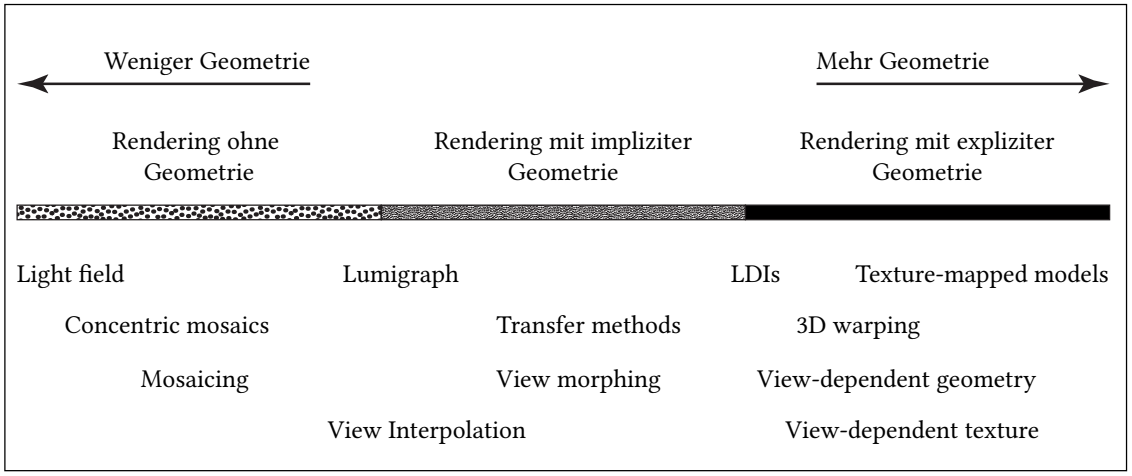


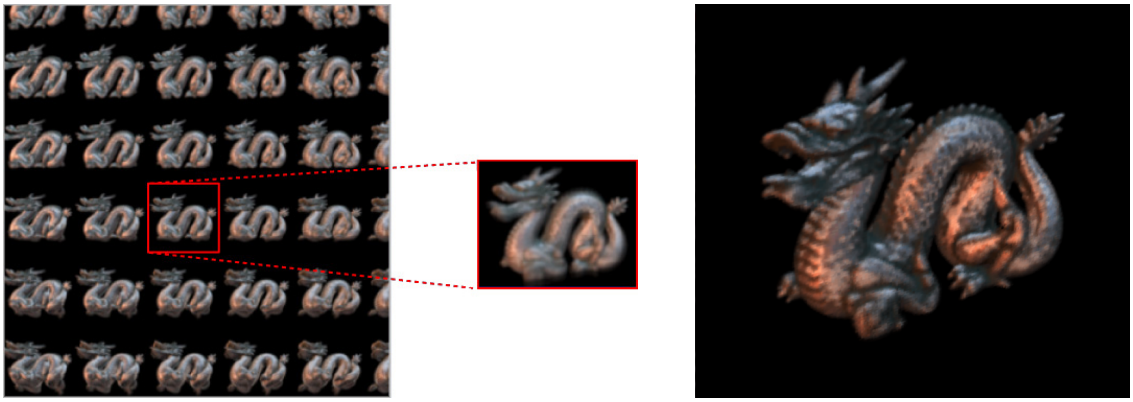

**Abbildung 2.25.:** Kategorien des bildbasierten Rendering. Die Verfahren reichen von Methoden ohne Geometrie bis zu Methoden, die Geometrie explizit modellieren. Die Bezeichnung der Beispielverfahren wurde bewusst in Englisch belassen, da sie eindeutiger sind, als etwaige Übersetzungen. Nach [Shum u. a., 2007].

der Kameras, deren Position, der Extraktionsmethode und vor allem in der Repräsentationsform bestehen [Shum u. a., 2003] [Schreer u. a., 2005, S. 154ff]. Dieser Abschnitt präsentiert lediglich einen Auszug aus den aktuellen Verfahren und nennt die prominentesten Vertreter sowie die wichtigsten Referenzen. Die für die vorliegende Arbeit wichtigen Verfahren werden an entsprechender Stelle diskutiert. Eine Übersicht über die in dieser Arbeit verwendete Kategorisierung zeigt Abbildung 2.25. Eine klare Abgrenzung ist kaum möglich, da sich verschiedene Verfahren oft verschiedener Techniken und Repräsentationsformen bedienen bzw. diese vermischen.

Rendering *ohne Geometrie* bezeichnet Verfahren, welche weder geometrische Informationen über Szene und Kameras nutzen noch diese als Repräsentationsform modellieren. Es sind Methoden, welche die Intensitätsverteilung des Lichtes innerhalb eines Raumes abtasten. Als mathematische Beschreibung dieser Intensitäten hat sich die plenoptische Funktion nach Adelson/Bergen [Adelson und Bergen, 1991] etabliert:

$$P = P(\theta, \phi, \lambda, t, V_x, V_y, V_z) \tag{2.70}$$

$P$ ist die Intensitätsverteilung der Lichtstrahlen. Die Argumente der Funktion lassen sich durch ein „ideales Auge“ veranschaulichen. Diese erfasst die Verteilung unter jedem Blickwinkel $(\theta, \phi)$, jeder Wellenlänge $(\lambda)$ zu jeder Zeit $(t)$ an jedem Ort im Raum $(V_x, V_y, V_z)$. Ziel von Methoden ohne Nutzung von Geometrie ist es, die plenoptische Funktion abzutasten. Es ist offensichtlich, dass dies nur über Einschränkungen der Anzahl der Argumente oder deren Definitionsbereich geschehen kann. So werden oft die zeitliche Komponente sowie die Wellenlänge eliminiert [McMillan und Bishop, 1995]. Auch sind Position und Blickwinkel fest oder nur eingeschränkt



(a) (b)

**Abbildung 2.26.:** (a) Ausschnitt aus den Bilddaten, die für ein Lightfield gespeichert werden (aus [Levoy, 2000]). Rendering einer neuen Ansicht des Lightfields (b).

definiert. *Lightfield Rendering* und *Lumigraph* erzeugen durch geschickte Parametrisierung und Begrenzung auf ein Sichtvolumen eine vier-parametrige Funktion [Levoy und Hanrahan, 1996; Gortler u. a., 1996]. Die Abtastung erfolgt mittels einer sehr dichten Kameraanordnung (vgl. Abb. 2.26). Die Synthese nutzt Interpolation um unvermeidbare Aliasingeffekte zu reduzieren. Lumigraph nutzt zusätzlich explizite geometrische Randbedingungen zur weiteren Aliasingvermeidung [Shum u. a., 2007, S. 13], weshalb die Methode auch in den Bereich der Verfahren mit expliziter Geometrie zu zählen ist. Auf Basis der ursprünglichen Methoden haben sich abgeleitete Methoden mit unterschiedlicher Parametrisierung und verschiedenen Verfahren zum Rendering entwickelt. Allen gemein ist eine immense Datenmenge der Repräsentationsform, bedingt durch die hohe Abtastdichte. Deshalb beschränken sich viele Verfahren zunächst auf Standbilder. Konzentrische Mosaike (*Concentric Mosaics*) bilden eine weitere Gruppe innerhalb der Verfahren ohne Geometrie. Sie beschränken die Kamerabewegung auf konzentrische Kreise auf einer Ebene. Somit ist eine plenoptische Funktion mit 3 Parametern abzutasten [Shum und He, 1999]. Bildmosaike fixieren die Kameraposition gänzlich. Die plenoptische Funktion kann dann auch aus einer unvollständigen Abtastung rekonstruiert werden [Shum u. a., 2007, S. 18]. Beispiele sind sphärische Panoramen, die aus verschiedenen Aufnahmen einer Szene zusammengesetzt werden [Szeliski, 1996; Szeliski und Shum, 1997]. Das populärste Beispiel ist QuickTime VR [Chen, 1995].

Demgegenüber stehen Verfahren unter der Nutzung expliziter Geometrie. Sie sind stark mit der Computergrafik verbunden. Einige Methoden dieser Kategorie ermitteln aus räumlich spärlich angeordneten Kameras ein 3D-Drahtgittermodell (*Mesh*) und dazugehörige Texturen. Die Repräsentationsform gleicht somit synthetischen Computergrafikmodellen, die anschließend mittels üblicher Verfahren (Echtzeitrendering, Raytracing etc.) gerendert werden. Vertreter dieser

**Abbildung 2.27.:** Teil des Eingangsdatensets und mittels verfeinertem Shape-from-Silhouette Algorithmus erzeugte Ansichten des 3D-Modells. Aus [Starck und Hilton, 2007].

Verfahren sind *Shape from Silhouette*-Methoden wie beispielsweise in [Grau, 2003; Smolic u. a., 2004; Starck und Hilton, 2007] genutzt (vgl. Abb. 2.27). Die Texturinformation der Szene wird aus den Farbinformationen der Aufnahmen gewonnen und durch Einfärben der Volumenelemente, statische oder blickabhängige dynamische Texturierung (*dynamic texturing*) des Modells realisiert. Andere Methoden nutzen ein vordefiniertes 3D-Modell und passen dieses an die aufgenommen Szene an [Carranza u. a., 2003]. Eine vielbeachtetes Verfahren sind *Layered Depth Images - LDI* [Shade u. a., 1998]. Sie nutzen eine *Sprite*-ähnliche Repräsentation zur Vermeidung von Löchern durch Aufdeckungen. Für jeden Pixel der Referenzkamera werden mehrere Farb- und Tiefenwerte gespeichert. So kann je nach Position der neuen Ansicht der entsprechende Pixel direkt ausgelesen und dargestellt werden. Jedoch benötigt diese Methode mehrere Eingangsbilder der Szene zur Erzeugung der Repräsentation. Eine umfangreiche Betrachtung weiterer Methoden findet sich in [Cheung u. a., 2005a,b].

Andere Verfahren speichern die 3D-Information ebenfalls explizit, jedoch nicht immer als Mesh. Dazu gehören Verfahren wie *Billboards*, die eine Textur auf einer zum Betrachter ausgerichteten Ebene aufbringen und somit eine Pseudo-Ansicht erzeugen. Aufwändigere Verfahren wie Billboard-Clouds verwenden mehrere geschickt angeordnete Billboards, um einen besseren 3D-Eindruck zu vermitteln [Décoret u. a., 2003]. Beim *3D-Warping* werden Tiefeninformation verwendet, um Punkte in den Raum zu projizieren und diese dann wieder in die neue Ansicht zurück zu projizieren [Würmlin u. a., 2004, 2005; Waschbüsch u. a., 2005]. Die Repräsentation kann dabei die Tiefenkarte oder aber die Punktwolke jeweils in Kombination mit der Textur sein. Diese Methode wird u. a. auch in dieser Arbeit adaptiert.

Die letzte Kategorie bilden Methoden, die eine implizite Geometrieannahme nutzen. Das heißt, es werden geometrische Beziehungen über Szene und Kameras genutzt (vgl. Abschnitte 2.1.3, 2.1.5, 2.1.6). Die Repräsentation und Synthese neuer Ansichten erfolgt jedoch auf Pixelebene, was

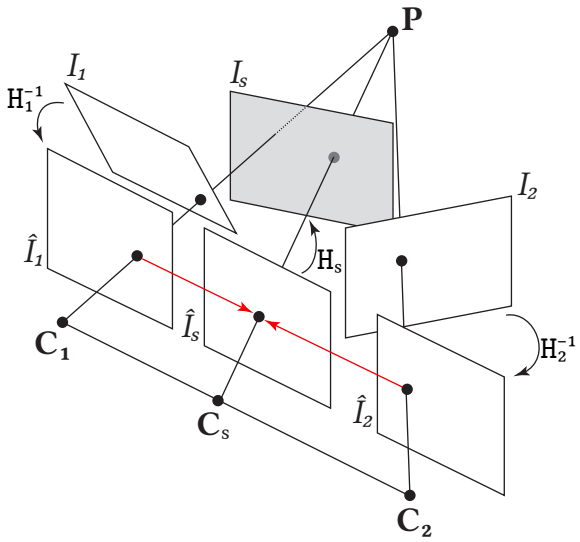



**Abbildung 2.28.:** Prinzip des View Morphings: Bilder $I_1$ und $I_2$ werden durch die Homographien $H_1^{-1}$ und $H_2^{-1}$ in die rektifizierten Bilder $\hat{I}_1$ und $\hat{I}_2$ transformiert. Anschließend erfolgt eine Interpolation entlang der Basislinie (rote Pfeile) gefolgt von der Derektifizierung (Post-Warping) der synthetisierten Zwischenansicht $\hat{I}_s$ durch die Homographie $H_s$, was zur finalen synthetisierten Ansicht $I_s$ führt (Nach [Seitz und Dyer, 1996]).

allgemein als Transfer oder *Warping* bezeichnet wird. Die einfachste Form ist die Ansichteninterpolation (*view interpolation*), ursprünglich von Chen vorgestellt [Chen und Williams, 1993]. Sie erzeugt keine physikalisch korrekten Bilder, kann jedoch bei dicht beieinander liegenden Kameras gute Resultate erzeugen. Notwendig sind Flussfelder, die für jeden Pixel die Korrespondenz im anderen Bild repräsentieren. Sie entsprechen letztlich der allgemeinen Disparität (vgl. Gl. (2.33)).

Eine Weiterentwicklung dieses Ansatzes stellen Seitz und Dyer mit dem Verfahren des *View Morphing* vor. Es basiert auf einer projektiven Transformation der Eingangsbilder (*Pre-Warping*), die mit einer Rektifizierung zu vergleichen ist. Anschließend wird eine einfache Interpolation auf Basis von Korrespondenzen, gefolgt von der Rücktransformation der interpolierten Ansicht, durchgeführt [Seitz und Dyer, 1996]. So werden theoretisch physikalisch korrekte Ansichten aus beliebigen Kamerapositionen erzeugt. Deren Position ist jedoch auf die Basislinie zwischen den Ursprungskameras beschränkt. Abb. 2.28 erklärt den Vorgang. Problematisch bei dem Verfahren ist zudem die Bestimmung der Homographie $H_S$ für den Post-Warping-Schritt.

Xiao und Sha erweitern das View Morphing Prinzip auf drei Ansichten und nennen das Verfahren *Tri-View Morphing* [Xiao und Shah, 2003, 2004]. Basierend auf der trifokalen Ebene bestimmen sie die Rektifizierungshomographien für die drei Ansichten. Auf Basis der rektifizierten Ansichten nutzen sie einen semi-automatischen Prozess zur Bestimmung dichter Korrespondenzen (Disparitäten). Das Kostenmaß ist die sich aus den drei Ansichten ergebende Kombination der Summe

absoluter Differenzen (SAD). Mittels dieser Korrespondenzen wird eine neue Ansicht auf der trifokalen Ebene durch Blending synthetisiert.

Eine freie Wahl der virtuellen Ansicht erlauben Verfahren, die auf Basis der Fundamentalmatrix oder des Trifokalen Tensors einen Pixeltransfer durchführen. Das Prinzip des Trifokaltransfers über Basistrilinearitäten wurde bereits im Abschnitt 2.1.6 erläutert. Die Fundamentalmatrix kommt im Verfahren von Laveau und Faugeras zum Einsatz [Laveau und Faugeras, 1994]. Zunächst werden das Kamerazentrum und die Bildebenen der virtuellen Ansicht durch die manuelle Spezifizierung der Epipole zur dritten Ansicht in den gegeben Referenzkameras bestimmt. Sind die Kameras nicht kalibriert, erfolgt die Bestimmung nur bis auf eine projektive Transformation genau (vgl. Abschnitt 2.1.5). Unter der Nutzung der Epipolargeometrie kann anschließend die Position eines Bildpunktes aus dem Schnittpunkt der Epipolarlinen in den gegebenen Referenzbildern bestimmt werden, was natürlich bekannte Korrespondenzen zwischen den Referenzansichten voraus setzt (vgl. Abb. 2.9 in Abschnitt 2.1.6). Die Mehrdeutigkeit durch die projektive Transformation kann durch kalibrierte Kameras eliminiert werden. Mehrdeutigkeit in der Abbildung von Raumpunkten (zwei Raumpunkte werden in einen Bildpunkt in der virtuellen Ansicht projiziert) lösen Laveau und Faugeras durch einen dem Ray Tracing ähnlichen Algorithmus. Das Verfahren kann beliebige Ansichten erzeugen, unterliegt aber den bereits beschriebenen singulären Fällen in bestimmten Kameraanordnungen (vgl. Abschnitt 2.1.6).

Allgemeine Probleme bei allen Verfahren sind Aufdeckungen und Überdeckungen durch die neue Ansicht. Erstere resultieren in Löchern, da Szenebereiche sichtbar werden, die durch keine der Referenzkameras abgedeckt wurden. Überdeckungen beschreiben den Fall, dass zwei Szenepunkte auf einen Bildpunkt in der virtuellen Ansicht abgebildet werden. Dadurch entstehende Artefakte werden oft mittels Tiefenpuffer-ähnlicher Verfahren gelöst. Eine Methode zur Lösung des Problems der Aufdeckung, die auch den Verfahren mit impliziter Geometrie zuzuordnen ist, ist die *Joint View Triangulation* [Lhuillier und Quan, 1999]. Auf Basis robuster Korrespondenzen zwischen den gegebenen Referenzansichten werden mittels Region Growing Verfahren [Otto und Chau, 1989] weitere Korrespondenzen generiert. Anschließend wird durch Delaunay Triangulierung ein zweidimensionales Gitter auf der Bildebene gebildet, welches die Abbildung planarer Flächen der Szenen repräsentiert. Diese wird dann in neue Ansichten auf der Basislinie überführt. Durch die implizite Verknüpfung der Korrespondenzen durch das Gitter werden so auch aufgedeckte Stellen ohne Bildinformation gefüllt. Eine Weiterentwicklung bezieht die Epipolarbedingung bei der Korrespondenzsuche und die Nutzung des Disparitätsgradienten als Kostenmaß beim Region-Growing mit ein[Lhuillier und Quan, 2002, 2003].

Es existieren viele Erweiterungen und Abwandlungen der oben genannten Verfahren [Alatan u. a., 2007]. Unterschiede bestehen in der Korrespondenzbestimmung und der Behandlung von Aufdeckungen oder von Artefakten durch Fehler in den Korrespondenzfeldern. Nennenswert ist die Arbeit von Zitnick et al. [Zitnick u. a., 2004]. Eine aufwändige segmentbasierte Stereoanalyse

mit Konsistenzüberprüfung über mehreren Kameras, gepaart mit einer separaten Behandlung verdeckter Randbereiche, führt zu einer hochqualitativen Ansichteninterpolation entlang der Basislinie zwischen den Kameras. Eine Repräsentationsform, die örtliche und zeitliche Zusammenhänge modelliert, präsentieren Vedula et al. [Vedula u. a., 2005]. Aus einer Voxelrepräsentation der 3D-Szene bestimmen sie mittels des Optischen Flusses in den einzelnen Ansichten der Referenzkameras den so genannten Szenenfluss (*Scene Flow*), der die Geschwindigkeit der Elemente über die Zeit als 3D-Vektoren repräsentiert. Daraus lassen sich virtuelle Ansichten nicht nur zu den aufgenommen Zeitinstanzen, sondern auch zu dazwischen liegenden Zeitpunkten synthetisieren. Lipski et al. erweitern das View-Morphing für die Erzeugung beeindruckender visueller Effekt [Lipski u. a., 2010a, 2011]. Auf der durch mehrere Kameras aufgespannten Oberfläche können virtuelle Ansichten unter Nutzung örtlicher und zeitlicher Informationen erzeugt werden. Durch eine geschickte Parametrisierung des Bewegungsbereiches der virtueller Kamera mittels eines tetraederförmigen Netze, lassen sich einfach virtuelle Kamerapfade realisieren. Die Interpolation durch *Forward-Warping* erfolgt anhand von Homographien, die aus den Lagebeziehungen der Kameras gewonnen werden. Entgegen des originalen View Morphings kann die Interpolation dabei örtlich wie auch zeitlich geschehen. Sie erfolgt anhand dichter Korrespondenzen, die mittels eines globalen Matching-Algorithmus unter der Verwendung von Farb- und SIFT-Merkmalen [Lowe, 2004] gewonnen werden [Lipski u. a., 2010b]. Als Besonderheit ist hervorzuheben, das sowohl die Kamerakalibrierung als auch deren zeitliche Synchronisation bildbasiert erfolgen.

## 2.4. Blickkorrektur mittels Bildsynthese

Im Bereich von Technologien und Algorithmen zur Blickkorrektur mittels Bildsynthese finden bereits seit Beginn der neunziger Jahre Arbeiten statt. Auch hier unterscheiden sich primär Analyseverfahren, Repräsentationsform und Rendering. In dieser Arbeit liegt der Fokus auf Methoden, die eine implizite Repräsentationsform und entsprechendes Rendering nutzen. Gründe dafür werden im Abschnitt 3.2 dargelegt. Modellbasierte Verfahren nutzen zur Darstellung das (angepasste) Kopf- oder Körpermodell einer Person oder einen Avatar. Diese Methoden sollen, abgesehen von nennenswerten Ausnahmen, nicht Betrachtungsgegenstand dieses Abschnitts sein.

Erste Arbeiten wurden 1993 von Ott et al. durchgeführt [Ott u. a., 1993]. Die Veröffentlichung sowie ein von denselben Autoren im selben Jahr eingereichtes Patent beschreiben recht allgemein ein Videokonferenzsystem [Lewis u. a., 1994]. Im beschriebenen Verfahren werden zwei Kameras über und unter bzw. links und rechts vom Monitor konvergent befestigt. Nach einem Rektifizierungsschritt erfolgt ein zeilenbasiertes Korrespondenzverfahren, welches mittels Dynamischer Programmierung gelöst wird [Cox, 1992]. Die Synthese erfolgt in der „Mitte“ der beiden Kameras durch Interpolation entsprechend der Disparität. Das Patent versucht das Prinzip der Blickkorrek-

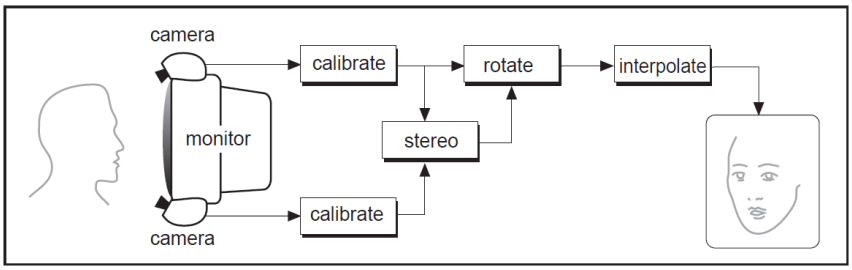
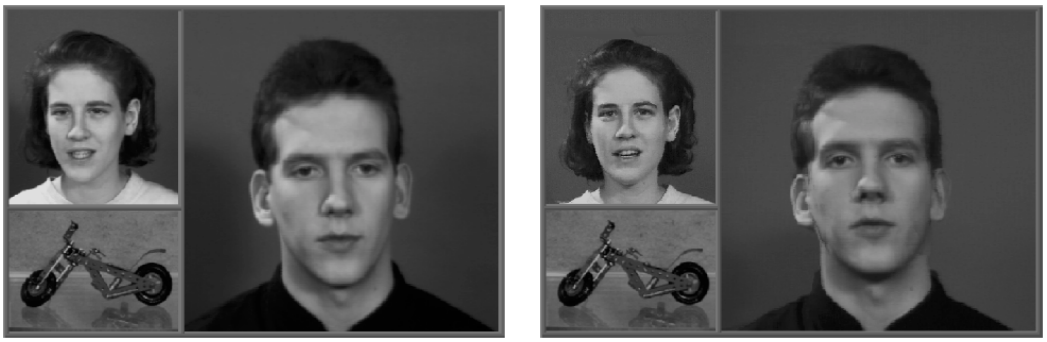



**Abbildung 2.29.:** Virtuelles Kamerasystem aus Ott 1993 [Ott u. a., 1993].

tur basierend auf Disparitätsanalyse zu schützen. Konkrete Ausführungen gibt es kaum, um die Generalität zu wahren. Einzig die Anordnung der Kameras ist etwas genauer genannt. Abbildung 2.29 zeigt ein Blockdiagramm des Prinzips. Die Rechenzeiten lagen im Jahr des Erscheinens der Veröffentlichungen selbst auf Workstations jenseits der Echtzeit-Anwendung. Heutzutage wäre mittels Dynamischer Programmierung eine Echtzeitanwendung bei entsprechender Implementierung denkbar, jedoch müssen die üblichen Artefakte bedacht werden (vgl. Abb. 2.24). Die Anordnung der Kameras erzeugt einen großen Basisabstand, was zu entsprechend großen verdeckten Bereichen im Randbereich und einem großen Disparitätsbereich führt. Letzterer macht die Stereoanalyse aufwändiger und erhöht die Möglichkeit von Fehlschätzungen.

Liu et al. beschreiben 1995 ebenfalls ein System zur Blickkorrektur in Videokonferenzanwendungen [Liu u. a., 1995]. Sie nutzen ein festes Setup mit drei Kameras (links, rechts, oben). Anschließend erfolgt eine mehrstufige Merkmals-Korrespondenzsuche (*Feature-Matching*) auf Basis von Richtung und Helligkeitsdifferenz mit adaptivem Fenster für jede Bildzeile in allen Kamerakombinationen, welche durch dynamische Programmierung gelöst wird. Dabei werden zusätzlich Interline-, Interlevel- und Interframe-Informationen genutzt. Erstere beschreibt ein Kostenmaß zwischen benachbarten Bildzeilen, die zweite zwischen verschiedenen Auflösungspyramiden im topologischem Matching (vgl. [Liu und Huang, 1993]) und letztere zwischen verschiedenen Zeitinstanzen. Die von links nach rechts und umgekehrt bestimmten Disparitäten werden über die Zeit geglättet. Die Autoren erwähnen, das die Disparitätsinformation nur an Merkmalspunkten gewonnen wird, so dass interpoliert werden muss, lassen aber offen, wie viele Merkmale genutzt werden. Die Synthese erfolgt in Abhängigkeit der Position der Person zur Kameraachse. Interessant an dieser Arbeit ist die Evaluation mittels Human-Factors-Experimenten. Mittels 70 Testsequenzen (35 synthetisiert, 35 von der oberen Kamera) wurden Probanden nach der Blickrichtung in verschiedenen Fenstern gefragt (vgl. Abb. 2.30). Die Erkennungsrate des Augenkontaktes konnte mittels der synthetischen Ansichten klar gesteigert werden. Über die wahrgenommene visuelle Qualität der Synthese wurde jedoch keine Erhebung gemacht.

Gemmel et al. erzeugen in ihrer Arbeit aus dem Jahr 2000 eine Pseudo-Korrektur der Blickrichtung [Gemmell u. a., 2000]. Im Analyseteil werden Augenkontur, Kopfpose und Blickrichtung aus einem monoskopischen Video bestimmt. Mit diesen Informationen werden beim Rendering



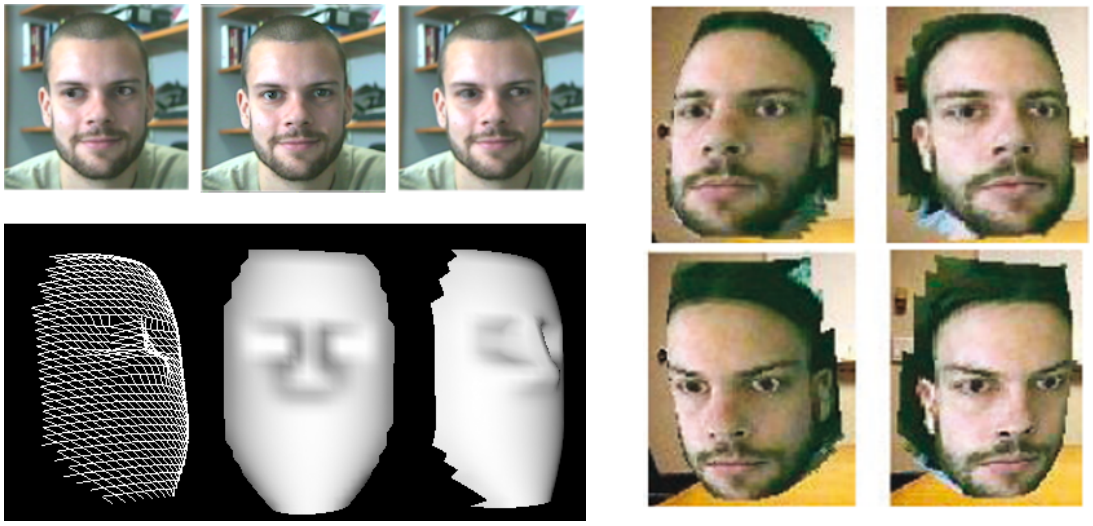
(a) (b)

**Abbildung 2.30.:** Anordnung der Testfenster bei Liu. Aus [Liu u. a., 1995].

zunächst die Augen alleine korrigiert. Anschließend erfolgt eine Kopfposenkorrektur basierend auf einem vereinfachten Kopfmodell. Die Detektion der Augen (genauer der *Sclera*[6]) erfolgt u. a. über die aktive Konturen-Methode [Kass u. a., 1988]. Die Synthese des korrigierten Auges nutzt A-priori-Annahmen zu Größenverhältnissen des Auges und Kreise und Zufallssrauschen zur Texturierung. Das Auge, oder besser die Pupille wird ohne Krümmung als Fläche modelliert. Diese Textur wird auf Basis der Daten aus der Augendetektion auf die originale Augenposition positioniert und belässt, ersetzt bzw. interpoliert die originalen Farbwerte. Die Ausrichtung der künstlichen Pupille wird über ihren Radius, ihre relative Position und die Ausrichtung des Kopfmodells bestimmt. Für das vereinfachte Kopfmodell werden auf Basis von Kopfposen-Daten Texturkoordinaten generiert. Die Größe des Modells wird in Abhängkeit der Augen- und Nasenposition angepasst. Anschließen wird die Textur orthogonal auf das Modell projiziert. Die Kopfposenbestimmung erfolgt über die Lösung eines Minimierungsproblems über die Abstände von bestimmen Merkmalen des Gesichtes. Abb. 2.31a zeigt beispielhaft Ergebnisse. Nachteil der Methode ist, dass nur der Kopf betrachtet wird, der Torso jedoch außen vor bleibt. Auch ist die Verwendung des vereinfachten Modells in den synthetischen Ansichten erkennbar. Den Bilder ist die Synthese mittels des einfachen Meshes anzusehen.

Ein ähnlicher Ansatz, der jedoch allein die Pupillen korrigiert, wurde 2010 von Wolf vorgestellt [Wolf u. a., 2010]. Er modelliert ein Auge mittels eines sechs Parameter-Modells (Zentrum, Ausdehnung, Winkel etc.). Die Videokommunikationsteilnehmer müssen zunächst in einer Trainingsphase in die Kamera blicken. Für diesen Blick werden die Parameter des linken und rechten Auges bestimmt. Dies geschieht durch den Vergleich mit einer zuvor manuell annotierten Datenbank unter der Nutzung von SIFT-Merkmalen [Lowe, 2004] mittels einer Nächster-Nachbar-Suche. Wurde das beste Parameter-Set für jedes Auge gefunden, erfolgt eine Verfeinerung der Parameter über ein Template-Matching durch normalisierte Kreuzkorrelation (NCC). Die geschieht für mehrere Ansichten in der Trainingsphase, womit mehrere Modelle der

[6] Lederhaut - weißer Bereich des Auges



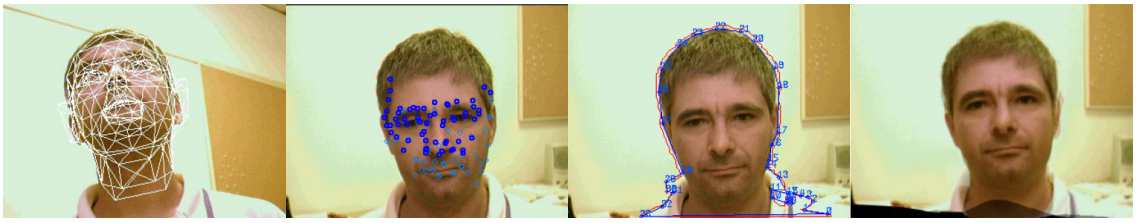
(a) (b)

**Abbildung 2.31.:** Blickkorrektur nach Gemmel [Gemmell u. a., 2000]. (a) Oben links und rechts: Ersetzte künstliche Pupillen. Mitte: Originalbild. Unten: Vereinfachtes Kopfmodell . (b) Texturiertes Modell in neuen Posen. All Beispiele zur Demonstration ohne Blickkontakt herzustellen.

**Abbildung 2.32.:** Beispiele für Augenersetzung nach Wolf 2010. Jeweils links das originale, rechts das korrigierte Bild. Aus [Wolf u. a., 2010].

zur Kamera gerichteten Augen vorliegen. In der eigentlichen Kommunikationssituation wird das Auge mittels Template-Matching über SSD verfolgt. Das ideale Auge aus der Trainingsphase wird mittels NCC ausgewählt, angepasst und ersetzt das real aufgenommene Auge. Dies geschieht durch Tiefpassquotientenbilder [Shashua und Riklin-Raviv, 2001] und linearem Überblenden im Randbereich. Die Qualität der erzeugten Beispielbilder ist, subjektiv bewertet, sehr hoch. Dennoch bleibt der Diskussionspunkt, ob eine reine Augenkorrektur reicht, um den Eindruck des Angesehen-Werdens herzustellen.

Ein simplifiziertes personalisiertes Kopfmodell zur Unterstützung einer Blickkorrektur kommt 2002 bei Yang et al. zum Einsatz [Yang und Zhang, 2002; Zhang und Yang, 2004]. Im ersten Schritt des Algorithmus wird eine Kopfposenverfolgung (*Head-Pose Tracking*). durchgeführt, die mit manuell gesetzten Merkmalspunkten initialisiert wird. Die Merkmalspunkte werden mittels eines KLT-Trackers [Lucas und Kanade, 1981; Shi und Tomasi, 1994] unter Einbeziehung der Epipolarbedingung zwischen den beiden Kameras verfolgt. Unter Verwendung der verfolgten

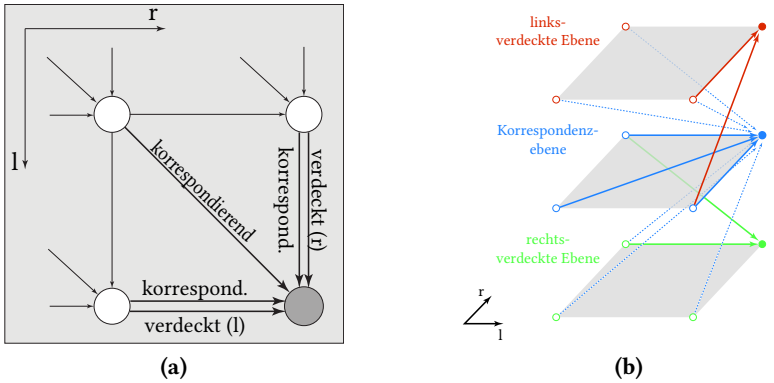


(a) (b) (c) (d)

**Abbildung 2.33.:** Blickkorrektur nach Zhang. (a) Visualisierung des personalisierten Kopfmodells. (b) Merkmalspunkte für rigide (dunkelblau) und nicht rigide (hellblau) Bereiche. (c) Kontur für eine Ansicht. (d) Synthetisierte Zwischenansicht. Aus [Zhang und Yang, 2004].

und von Ausreißern bereinigten Punkte wird die aktuelle Kopfpose bestimmt. Dies geschieht, indem der Projektionsfehler der jeweiligen 3D-Modellpunkte über die Translation $\mathbf{t}$ und Rotation $\mathtt{R}$ mittels Levenberg-Marquardt-Algorithmus minimiert wird. Anschließend erfolgt ein Stereo-Matching für nicht-rigide Teile des Gesichtes und die Kontur der Person. NCC dient dabei als Kostenmaß mit zusätzlicher Überprüfung mittels Disparitätsgradientenlimit, welches als Maß für eine kontinuierliche Oberfläche einer Szene herangezogen werden kann [Pollard u. a., 1986]. Die Kontur wird über eine nicht näher spezifizierte Hintergrundsubtraktion bestimmt und in eine polygonale zweidimensionale Form überführt. Korrespondenzen zwischen den Segmenten der Konturen des oberen und unteren Bildes werden mittels Dynamischer Programmierung ermittelt. Das dabei eingesetzte Kostenmaß setzt sich aus drei Komponenten zusammen: Einem Kostenmaß auf Basis der Epipolarbedingung, dem Disparitätsgradientenlimit und der Differenz in der Orientierung zweier Segmente. Durch diese Vorgehensweise ist das Ergebnis eine spärliche Disparitätskarte sowie die Korrespondenzen zwischen den Konturen. Die Synthese der Ansicht zwischen den Kameras erfolgt nach zwei Methoden. Die erste Methode nutzt View Morphing (vgl. vorheriger Abschnitt 2.3). Die zweite nutzt ein texturiertes 2D-Drahtgittermodell. gebildet aus den Konturen und den ermittelten Korrespondenzen, und wendet darauf eine gewichtete Texturierung an. Einzelne Ergebnisse des Verfahrens sind in Abb. 2.33a dargestellt. Das Verfahren ist als recht aufwändig zu bezeichnen. Diverse Algorithmen mit unterschiedlichen Parametern und jeweiliger Fehleranfälligkeit kommen zum Einsatz. Die erzeugten Beispielbilder wirken qualitativ gut. Die Kameraanordnung mit großem Basisabstand erzeugt die bereits erläuterten Probleme der Verdeckung und des großen Disparitätsbereiches.

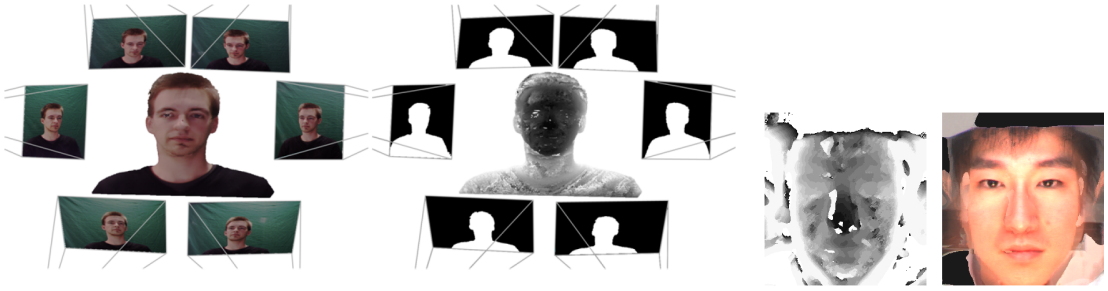
Ein Verfahren unter ausschließlicher Verwendung impliziter Geometrien präsentieren Criminisi et al. [Criminisi u. a., 2003, 2005]. Sie platzieren zwei Kameras vertikal zentriert links und rechts vom Bildschirm. Für die rektifzierten Bilder wird ein modifizierter Ansatz zur Stereoanalyse mittels Dynamischer Programmierung vorgestellt. Fokus liegt dabei nicht nur auf der korrekten Bestimmung von Disparitäten, sondern auch auf einer korrekten Verdeckungsbestimmung um anschließend Vordergrund und Hintergrund zu separieren. Anstelle der üblichen drei erlaubten




(a) (b)

**Abbildung 2.34.:** Dynamische Programmierung bei der Blickkorrektur nach Criminisi. (a) *Five-move model.* Diese Erweiterung des klassischen DP unterscheidet zwischen Disparitätswechseln aufgrund von Verdeckungen und schrägen Flächen. (b) Das 3-Ebenen-Modell erlaubt nun 13 anstelle von 3 möglichen Übergängen (vgl. auch Abb. 2.23). Nach[Criminisi u. a., 2003].

Schritten (vgl. Abschnitt 2.2.6) wird ein so genanntes *5-Move Model* entwickelt (vgl. Abb. 2.34a). Es vermeidet die Mehrdeutigkeit zwischen verdeckten Disparitätssprüngen und Sprüngen, die z. B. durch geneigte Oberflächen entstehen können. Die Kosten werden mittels eines 3x3 NCC Fensters berechnet und der Kostenraum vor dem Optimierungsschritt mittels Gaußfilter geglättet. So wird u. a. Konsistenz zwischen benachbarten Zeilen erzeugt. Aus dem neuen DP-Modell ergeben sich durch die Modellierung des Kostenraumes auf drei Ebenen 13 mögliche Schritte, die über zwei Strafterme parametrisiert werden (vgl. Abb. 2.34b). Die Synthese der zentralen Ansicht zwischen beiden Kameras geschieht zeilenweise direkt mittels des minimalen Kostenpfades. Die Farbwerte korrespondierender linker und rechter Pixel werden gemittelt und entsprechend der bestimmten Disparität platziert. Als verdeckt markierte Bereiche erhalten die Farbe des entsprechenden Pixel in der anderen Ansicht und die Disparität des letzten korrespondierenden Pixelpaares auf dem Pfad. So wir ein automatischen Füllen der Verdeckungen durch Propagierung des Hintergrundes realisiert. Um dabei auftretendes Flackern über die Zeit zu vermeiden, wird auf Basis der Verdeckungserkennung eine Vordergrund/Hintergrundsegmentierung vorgenommen. Diese setzt sich aus durch einen Abschwächungsfaktor gewichteten Werten vorheriger Hintergründe und des aktuellen Hintergrundes zusammen. So können aktuell verdeckte Bereich ggf. durch früher sichtbare Bereiche gefüllt werden. Translatorische Bewegungen der virtuellen Kamera in alle Richtungen können durch einfache Projektionsvorschriften, die im Paper hergeleitet werden, direkt aus dem Kostenraum erzeugt werden. Rotationen sind nicht möglich. In [Criminisi u. a., 2007] verändern die Autoren 2007 das DP-Schema zu einem reduzierten *four-state*-Modell, welches allerdings mehr Parameter besitzt. Auch nutzen sie nun als alternatives Kostenmaß die normalisierte Summe der absoluten Quadrate und analysieren detailliert die Leistungen ihres Algorithmus in Bezug auf Verdeckungsdetektion. Das Verfahren ist das universellste. Es stellt keinerlei Bedingungen an den Bildinhalt.



(a) (b)

**Abbildung 2.35.:** Blickkorrektur mittels Plane Sweep. (a) Tiefenkarte und synthetisierte zentrale Ansicht aus Dumont [Dumont u. a., 2008]. (b) Tiefenkarte und synthetisierte Ansicht aus Muarayama/Mukai [Murayama u. a., 2010].

Ebenfalls auf Basis einer Repräsentation, die implizite Geometrie nutzt, arbeitet das Verfahren von Mukai/Murayama et al. [Mukai u. a., 2009; Murayama u. a., 2010]. Zur Bestimmung der Disparität wird ein *Plane Sweep* ähnliches Verfahren genutzt. Die Bildebenen aus drei um einen Fernseher angeordneten Kameras werden mittels Homographie in die virtuelle Bildebene transformiert. Dies geschieht für mehrere verschiedene Tiefenwerte der virtuellen Bildebene. Die Projektionen werden paarweise mittels SAD verglichen. Objektoberflächen (der Person), die in der jeweils untersuchten z-Ebene liegen, werden korrekt projiziert und erzeugen so einen geringen Kostenwert. In einer anderen Tiefenebene liegende Oberflächen werden falsch projiziert, was zu einem hohen Kostenwert führt. Die Homographien zur Ebenentransformation werden aus den kalibrierten Kameramatrizen abgeleitet. So entsteht eine Tiefenkarte, auf deren Basis die virtuelle Ansicht aus den gewichtetet Anteilen der drei Bilder synthetisiert werden kann (vgl. Abb. 2.35b). Eine zweite, vereinfachte Methode basiert auf der Annahme der bekannten Distanz des Nutzers zum Bildschirm, projiziert das Bild einer Kamera und ersetzt anschließend die Augen durch einen einfachen Algorithmus.

Ein *Plane Sweep*-Ansatz kommt auch bei Dumont et al. zum Einsatz [Dumont u. a., 2008, 2009]. Unter der Verwendung von sechs um den Bildschirm angeordneten Kameras werden die Videos zunächst segmentiert, um den Vordergrund (die Person) frei zu stellen. Das Kostenmaß für die anschließende Tiefen- und Farbbestimmung durch *Plane Sweep* orientiert sich an dem von Yang und anderen [Yang u. a., 2003]. Es entspricht der SSD unter Einbeziehung des Mittelwertes der gerade betrachteten Pixel aller Bilder. Zu Verbesserung der Tiefenkarten erfolgt eine fensterbasierte Konsistenzüberprüfung in der Nachbarschaft eines jeden Tiefenwertes. Als Ausreißer markierte Bereiche werden durch morphologische Operationen gefüllt. Abschließend wird die Tiefenkarte mittels Gauß-Filter geglättet. Die durch den *Plane Sweep* bereits interpolierten Farbwerte werden entsprechend der verbesserten Tiefenkarte aktualisiert. Durch eine einfache Bewegungsanalyse wird der mögliche Tiefenbereich eingeschränkt. Die komplette Algorithmenkette wird effizient auf der GPU implementiert. Eine mit dem Verfahren gewonnene Tiefenkarte ist in Abb. 2.35a dar-

gestellt. Beide *Plane Sweep*-Methoden bedürfen mehrerer um den Monitor angeordneter Kameras, um eine robuste Tiefenschätzung zu ermöglichen. Diese müssen außerdem exakt kalibriert sein. Dies ist insbesondere für die Videokommunikation im Heimbereich aber auch für den mobilen Einsatz wenig praktikabel.

Am Fraunhofer HHI wird bereits seit geraumer Zeit an virtuellen Konferenzsystemen geforscht. Ein erster Prototyp eines so genannten *shared virtual table environments* – SVT – wurde 2002 von Kauff und Schreer vorgestellt [Kauff und Schreer, 2002]. Zur Disparitätsanalyse wird ein rekursives Verfahren genutzt (*Hybrid Recursive Matching*-HRM) [Schreer u. a., 2001; Atzpadin u. a., 2004]. Ähnlich der Bewegungsanalyse bei der Videocodierung basiert der Algorithmus auf der Annahme, dass sich die Disparität über die Zeit und den Ort nicht stark ändert und arbeitet auf $4 \times 4$-Blöcken. Zum Bestimmen der aktuellen Disparität werden zwei rekursive Prozesse genutzt, die letztlich einen finalen Disparitätswert für jeden Block bestimmen (vgl. Blockschaltbild in Abb. 2.36a). Der erste Prozess, die Block-Rekursion, nutzt die örtliche und zeitliche Nachbarschaft eines Blockes zur Schätzung seiner aktuellen Disparität. Die Abarbeitung erfolgt dabei abwechselnd mäanderartig für jede zweite Zeile. Der beste Kandidat für den aktuellen Block wird mittels eines SAD-basierten Maßes (*Displaced Block Distance*) zwischen linkem und rechtem Bild in Abhängigkeit dreier Nachbarblöcke ausgewählt. Um zeitliche und örtliche Veränderungen der Szene zu modellieren, wird in diesen Prozess jeweils ein neuer Disparitätswert injiziert. Dieser wird durch den zweiten rekursiven Prozess, die Pixel-Rekursion, bestimmt. Diese findet nun auf Pixelbasis innerhalb eines $4 \times 4$-Blockes statt. Rekursiv wird eine Disparität aus der absoluten Distanz zwischen linkem und rechtem Bild unter Annahme der vorherigen Disparität sowie einem lokal approximierten Gradienten ähnlich dem optischen Fluss berechnet. Für Details sei auf [Atzpadin u. a., 2004] verwiesen.

Auf Basis einer links-rechts-Konsistenzprüfung mit einem vorbestimmten Schwellenwert, werden anschließend unsichere Disparitäten ausgeschlossen. Die dadurch entstehenden Löcher werden mittels verschiedener Verfahren abhängig von einem Entscheidungsschema gefüllt. Primär wird dabei zwischen Fehlern durch Verdeckung und anderen Fehlern unterschieden. Es kommt dann z. B. ein Füllverfahren auf Basis der Segmentierung der Person und der Hände, oder eine einfache bilineare Interpolation zum Einsatz. Ähnlich erfolgt die Berechnung des dichten Disparitätsfeldes auf Basis der $4 \times 4$-Blöcke. Vorteil des HRM ist, dass es sehr schnell berechenbar und seine Komplexität nicht vom möglichen Disparitätsbereich abhängig ist, da keine lokale Suche über den kompletten Bereich durchgeführt wird. Durch die verwendete Rekursion unter Einbeziehung der Nachbarschaften kann der Algorithmus als spezieller globaler Algorithmus zur Stereoanalyse eingestuft werden.

Die Synthese bei obigem Ansatz basiert auf einem in mehrere Schritte unterteilten Interpolationsalgorithmus [Lei und Hendriks, 2001, 2002]. Unter der exakten Kenntnis der externen und internen Kameraparameter wird basierend auf den rektifizierten Eingangsbilder zunächst eine

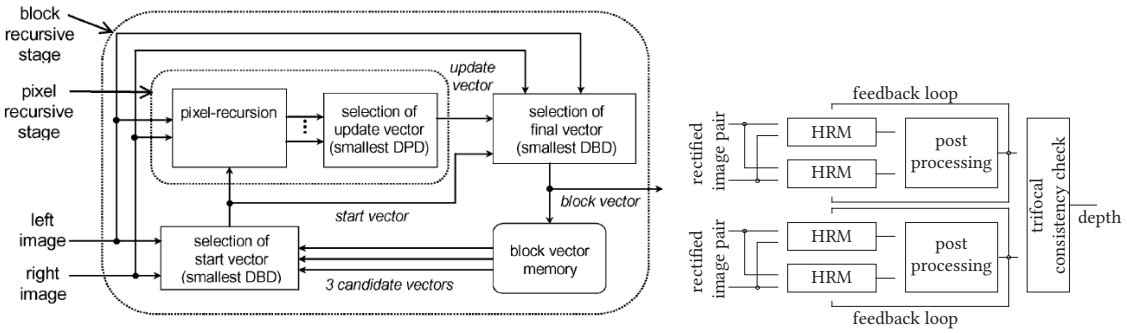



**Abbildung 2.36.:** (a) Blockschaltbild des Hybrid Recursive Matching nach Atzpadin. Aus [Atzpadin u. a., 2004]. (b) Nutzung des HRM in der Weitereintwicklung der Tiefenbestimmung. Nach [Feldmann u. a., 2010].

Zwischenansicht auf der Basislinie berechnet, was dem View-Morphing entspricht (vgl. Abschnitt 2.3). Um Positionen der virtuellen Kamera auch jenseits der Basislinie zu erzeugen, folgen weitere Schritte. Eine $Y$-Extrapolation „bewegt“ die Ansicht der virtuellen Kamera an die entsprechende $Y$-Position. Ebenso wird die Disparitätskarte entsprechend transformiert. Anschließend erfolgt eine $Z$-Translation, die vereinfacht als Zoom (lineare Skalierung) implementiert wird. Abschließend wird die so platzierte Kameraansicht derektifiziert. Die so synthetisierten Bilder sind geometrisch nicht korrekt, jedoch zeigen die Autoren durch einen Vergleich, dass sie nur geringe Differenzen zu einer Referenzansicht (*Ground Truth*) aufweisen.

Die konsequente Weiterentwicklung des Konferenzsystems erfolgt im Projekt „3D Presence“ [Divorra, 2011]. Für die Disparitätsanalyse kommt nach wie vor der HRM zum Einsatz (vgl. Abb. 2.36b). Er wird jedoch auf zwei Kamerapaare (eines vertikal, eines horizontal) angewandt und durch eine trifokale Konsistenzprüfung ergänzt [Feldmann u. a., 2009b,a; Schreer u. a., 2009; Feldmann u. a., 2010]. Im Projekt wird ein größeres Setup genutzt, so dass auch *Visual Hull*-Algorithmen (vgl. Shape from Silhouette in Abschnitt 2.3) für die weit auseinander liegenden Kameras genutzt werden können. Die Daten des extrahierten 3D-Modells werden für die nochmalige Kontrolle sowie Verbesserung der Disparitätskarten des HRM genutzt. Eine alternativen Ansatz, der direkt auf die Blickkorrektur abzielt, stellen die Autoren in [Waizenegger u. a., 2011] vor. Anstelle des HRM wird ein so genannter *Patch Sweep*-Ansatz verwendet. Dabei wird, ausgehend von vier Kameras, ein schnell zu berechnender Plane Sweep durch eine Patch Sweep-basierten Optimierung erweitert. So kann die bei der Stereoanalyse übliche Annahme frontaler Flächen vermieden werden. Durch die geschickte Einschränkung können die Orientierungen der Patches reduziert und eine aufwändige Verdeckungsbehandlung vermieden werden. Die in der Veröffentlichung exemplarisch gezeigten Bilder der Tiefendaten können subjektive überzeugen. Die beschriebenen Systeme erzeugt mit hohem technischen Aufwand bereits überzeugende korrigierte Ansichten. Kameraanordnung und Synthespositionen sind in allen Veröffentlichungen

auf den konkreten Anwendungsfall eines großen Videokonferenzsystems zugeschnitten, der für den Heimbereich eher ungeeignet ist. Dennoch sind die verwendeten Methoden als innovativ und die Ergebnisse als qualitativ hochwertig einzustufen.

# 3. Systematisierung

Zur Lösung der Aufgabenstellung erfolgt in diesem Kapitel eine Systematisierung der notwendigen Systemkomponenten und deren Interaktionen untereinander. Als Grundlagen dient das erstmals von Rittermann vorgestellte [Rittermann, 2004, 2007] und in dieser Arbeit erweiterte Modell der 3D-Videoobjektgenerierung. Als 3D-Videoobjekt wird die Darstellung von vom Hintergrund freigestellten, zeitveränderlichen Szenenelementen wie z. B. Menschen bezeichnet, wobei der Blickpunkt frei wählbar ist. Die Realisierung von 3D-Videoobjekten erfolgt durch bildbasiertes Rendering (vgl. Abschnitt 2.3). Wenn auch in einem anderen Anwendungskontext entwickelt, so bildet das Modell die notwendigen Bestandteile der Blickkorrektur gut ab. Ausgehend von diesem Modell werden das in dieser Arbeit entwickelte Verfahren eingeordnet und zwei mögliche Anwendungsszenarien entwickelt. Anschließen werden die notwendigen Arbeitsschritte definiert. Für jeden Verarbeitungsschritt in der Kette werden abschließend Qualitätsparameter und deren Auswirkungen diskutiert.

## 3.1. Modell der Verarbeitungskette

Rittermann entwickelt in [Rittermann, 2007] ein Modell der 3D-Videoobjektgenerierung, welches in Abb. 3.1 dargestellt ist. Erster Block ist die aufzunehmende Szene selbst. Diese wird durch die Aufnahme in eine so genannte *primäre Repräsentation* überführt. Diese Repräsentation kann bereits Szeneinformationen enthalten (z. B. durch Tiefenkameras generiert). Die folgende *sekundäre Repräsentation* wird durch weitere Verarbeitungsschritte in der Kette erzeugt. Die anschließende *Bildsynthese* erzeugt letztlich das 3D-Videoobjekt. Diese Darstellung stellt die Repräsentationsformen in den Mittelpunkt. Sie bildet die eigentlichen Arbeitsschritte für den Zweck dieser Arbeit unzureichend ab. Das Modell wird deshalb in ein methodenzentriertes Modell überführt, welches in Abbildung 3.2a dargestellt ist. Der Vorgang der *Aufnahme* besitzt Parameter, welche die folgenden Blöcke beeinflussen. Er hat damit bereits Auswirkungen auf die visuelle Qualität der erzeugten Ansicht und ist wichtiger Bestandteil der Verarbeitungskette. Durch die Aufnahme wird die primäre Repräsentation erzeugt, welche äquivalent mit der des Modells von Rittermann ist. Der anschließende Verarbeitungsblock überführt die primäre Repräsentation in die sekundäre. Im einfachsten Fall ist der Block funktionslos und beide Repräsentationsformen identisch. Dies träfe z. B. bei Tiefenkameras zu, welche direkt die sekundäre Repräsentationsform erzeugen. Im

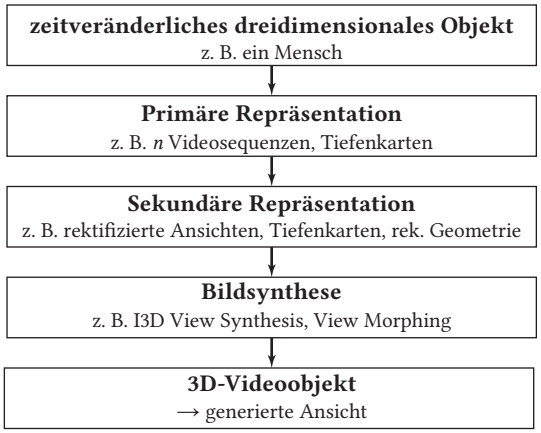



**Abbildung 3.1.:** Modell der 3D-Videoobjektgenerierung nach Rittermann [Rittermann, 2007].

Fall von Stereoaufnahmen repräsentiert er den funktionalen Bereich der Stereoanalyse und ist von entscheidendem Einfluss. Um eine Allgemeingültigkeit beizubehalten, wird der Block in dieser Arbeit als *Vorverarbeitung* bezeichnet.

In einem Videokommunikationsszenario, aber auch bei anderer Nutzung von 3D-Videoobjekten – z. B. im Kontext virtueller Welten – sind neben den bildverarbeitenden Blöcken auch Komponenten zur digitalen Übertragung oder Speicherung der Repräsentationsdaten notwendig. Zeitveränderliche Multimediadaten werden dabei aufgrund ihrer Datenmengen üblicherweise effizient komprimiert. Daher wird das Modell um Blöcke für *Kodierung* und *Dekodierung* ergänzt. Da sowohl die Übertragung als auch die Kodierung die Daten verändern kann, entsteht nach der Dekodierung die *primäre Zielrepräsentation.* Durch die Vielfalt von möglichen Repräsentationsformen können Verfahren zur Kodierung von Computergrafik-Netzen (*Meshes*) (z. B. in [Stefanoski und Ostermann, 2008]) ebenso zum Einsatz kommen, wie beispielsweise Techniken zur effizienten Kodierung und Übertragung von Multi-View-Daten [Merkle u. a., 2007a] oder Tiefen- bzw. Disparitätskarten [Merkle u. a., 2007b]. Geschehen Kodierung sowie Übertragung/Speicherung verlustlos und erhalten die Daten der sekundären Repräsentation vollständig, so ist die primäre Zielrepräsentation mit der sekundären Repräsentation identisch.

Der Vorgang der Bilderzeugung aus der primären Zielrepräsentation wird als *Synthese* bezeichnet. Da auch dieser Vorgang mittels verschiedener Verfahren realisiert werden kann, wird eine letzte Präsentationsform, die *Zielrepräsentation* erzeugt. Diese kann z. B. ein monoskopisches Video oder auch ein Stereo- oder Multi-View-Video sein. Die Zielrepräsentation wird letztlich zur *Anzeige* gebracht.

Das Modell wird nun auf den konkreten Fall der Blickkorrektur mittels Stereoanalyse und Bildsynthese angewendet, wobei zwei grundsätzliche Szenarien definiert werden:

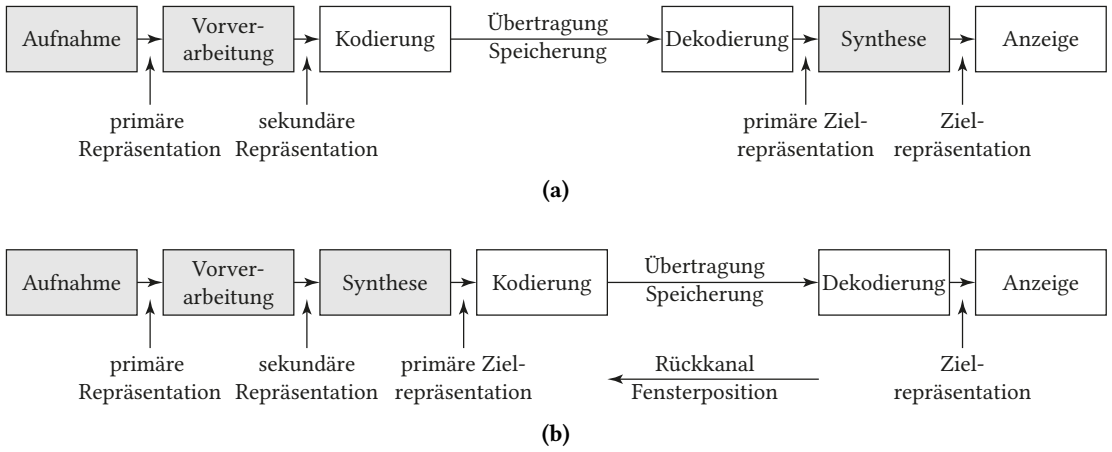



**Abbildung 3.2.:** (a) Szenario 1: Allgemeines Modell der Verarbeitungskette für die Blickkorrektur mittels Bildsynthese. (b) Szenario 2: Modifiziertes Modell für die Videokommunikation, wobei die Synthese bereits vor der Übertragung geschieht. Grau hinterlegte Blöcke sind die für diese Arbeit relevanten. Details im Text.

**Szenario 1:** Entsprechend der klassischen Übertragung von multimedialen Inhalten finden Aufnahme und Analyse und somit die Erzeugung der sekundären Repräsentation jeweils auf dem lokalen Gerät der Kommunikationsteilnehmer statt. Die sekundäre Repräsentation wird kodiert, übertragen und auf Empfängerseite dekodiert. Da diese Repräsentationsform die dreidimensionalen Szeneninformationen enthält, kann auf Empfängerseite die blickkorrigierende virtuelle Ansicht synthetisiert und angezeigt werden. Der Vorteil der Übertragung der kompletten sekundären Repräsentation ist, dass die Position der virtuellen Kamera auch interaktiv verändert werden könnte. Ebenso kann die Synthese direkt an vorhandene Anzeigegeräte angepasst werden und so z. B. monoskopisch oder stereoskopisch erfolgen. Der offensichtliche Nachteil besteht in der größeren Datenmenge, welche die sekundäre Repräsentation durch die dreidimensionalen Szeneninformationen enthält. Auch kann sich die Übertragung und die Kodierung negativ auf die Synthesequalität auswirken, da z. B. Kodierungsartefakte in der primären Zielrepräsentation z. B. zu Fehlplatzierungen oder Löchern führen könnten (vgl. auch den folgenden Abschnitt 3.3). Das erste Szenario entspricht Abb. 3.2a.

**Szenario 2:** Der komplette Vorverarbeitungs- und Synthesevorgang wird bereits vor der Übertragung durchgeführt. Diese Vorgehen würden die Nachteile von Szenario 1 beseitigen. Ist die Zielpräsentation das synthetisierte, blickkorrigierte, monoskopische Bild, so wäre die Datenmenge geringer als bei der sekundären Repräsentation. Auch die Kodierung kann keinen Einfluss mehr auf die Synthese nehmen, da die Übertragung erst anschließend stattfindet. Jedoch muss in diesem Szenario Position und Lage der virtuellen Kamera bereits auf der Sendeseite bekannt sein. Dies wäre nur durch die Übertragung mittels Rückkanal möglich, welcher jedoch bei der Videokommunikation ohnehin vorhanden ist. Auch eine interaktive Betrachtung auf

Empfängerseite sowie die Anpassung an bestimmte Displaytypen würde voraussetzen, dass die dazu notwendigen Informationen zuerst an den Sender übertragen werden. Die Integration in vorhandene Software zur Videokommunikation ist mit diesem Szenario einfacher, da im Falle einer monoskopischen Ansicht vorhandene Videoencoder und -decoder genutzt werden können. Lediglich die Sendung besagter Zusatzinformationen müsste realisiert werden. Szenario 2 ist in Abb. 3.2b dargestellt. Grundsätzlich ist festzustellen, dass sich dieses Szenario nur bei einer eins-zu-eins Kommunikationssituation effizient realisieren lässt, da für jeden weiteren Kommunikationsteilnehmer die notwendig Anzahl aller zu synthetisierender Ansichten exponentiell steigt.

Der Fokus dieser Arbeit liegt auf den Algorithmen der Stereoanalyse und Bildsynthese. Daher werden in den weiteren Abhandlungen nur diese Blöcke weiter betrachtet (grau hinterlegt in Abb. 3.2). Von den definierten Szenarios entspricht diese Vorgehensweise dem zweiten. Es werden Verfahren der ersten drei Blöcke Aufnahme, Vorverarbeitung und Synthese entwickelt. Durch das verlustfreie Speichern der primären Zielrepräsentation entspricht diese bereits der Zielrepräsentation und kann so beispielsweise für Tests verwendet werden. Kodierung und Übertragung sind *nicht* Gegenstand dieser Arbeit.

## 3.2. Randbedingungen und Systemspezifikation

Bereits in Abschnitt 2.4 wurden aktuelle Verfahren zur bildbasierten Blickkorrektur diskutiert. Einige Verfahren nutzen mehr als zwei Kameras, andere eine Stereo-Anordnung, die für den Anwendungsfall im Heimbereich ungünstig ist. Auch sind mitunter (semi-)professionelle Kameras im Einsatz, deren Kosten die breite Anschaffung durch Normalverbraucher auch in Zukunft nicht ermöglichen. Da die Anwendung im Heimbereich maßgeblich für diese Arbeit ist, wird festgelegt, dass die Aufnahme durch maximal zwei einfache, handelsübliche Webcams erfolgen soll. Auch deren Technologie hat sich in den letzten Jahre durchaus verbessert, dennoch wird – vor allem bedingt durch die kleinen Optiken – deren produzierte Bildqualität immer schlechter ausfallen als z. B. bei Industriekameras, Camcordern oder gar Broadcast-Kameras. Die primäre Repräsentationsform ist somit als die *Bilder der Stereoaufnahme* eines Kommunikationsteilnehmers definiert.

Diese Festlegung führt implizit zur Auswahl der Repräsentationsform für die dreidimensionale Szeneninformation. Die Entwicklung eines Verfahrens unter der Verwendung expliziter Geometrie würde die Nutzung mehrerer Kameras mit großer Basislinie notwendig machen (Kameradom). Nur so lägen z. B. für ein Visual Hull-Verfahren genug Informationen vor. Auch ein Plane-Sweep-Ansatz, wie von zwei in Abschnitt 2.4 beschriebenen Verfahren verwendet, kann erst durch eine höhere Anzahl von Kameras von Vorteil gegenüber der Stereoanalyse sein. Somit wird die sekundäre Repräsentation als Disparitäts- oder Tiefenkarte mit der dazugehörigen

| Block der Verarbeitungskette | Methode und Repräsentationsform | Begründung |
|---|---|---|
| Aufnahme (primäre Repräsentation) | Webcams (Stereosequenzen) | - Kameras für den Heimbereich<br>- geringer Installationsaufwand<br>- kostengünstig |
| Vorverarbeitung (sekundäre Repräsentation) | Stereoanalyse (Textur & Disparitätskarte) | - durch Aufnahme bedingt<br>- Repräsentation einfach speicherbar |
| Synthese (Zielrepräsentation) | 3D-Warping, trifokaler Transfer (monoskop. Sequenz) | - freie Kameratransformation<br>- physikalische Korrektheit |

**Tabelle 3.1.:** Spezifikation für die Verarbeitungskette zum Zweck der Blickkorrektur mittels bildbasiertem Rendering in der Videokommunikation.

Textur festgelegt. Hinzu kommt eine durch Vordergrundsegmentierung erzeugt Maske. Durch die Möglichkeiten der Stereoanalyse kann diese als Disparitätskarte und Textur jeweils für das linke und das rechte Bild vorliegen. Ein Vorteil dieser Repräsentationsform ist die einfache Handhabung, da sie in Standard-Bildformaten gespeichert werden kann.

Die Synthese der virtuellen Kameraansicht ist eng mit der verwendeten sekundären Repräsentationsform verbunden. Disparitätskarten repräsentieren Korrespondenzen zwischen den Abbildungen von Raumpunkten. Ebenso lassen sie sich im achsenparallelen Fall mittels Gleichung (2.27) in Tiefeninformationen transformieren. Direkt aus der Disparitätskarte ließe sich eine einfache Ansichteninterpolation durchführen. Physikalische Korrektheit ist mittels View Morphing erreichbar. Beide Verfahren beschränken die Position jedoch auf die Basislinie zwischen den Kameras. Dadurch wären sowohl die realen als auch die virtuellen Kamerapositionen eingeschränkt. Unter der Verwendung von Tiefeninformation für jeden Pixel ließe sich ebenso ein 3D-Warping (vgl. Abschnitt 2.3) umsetzen, welches beliebige virtuelle Kamerapositionen und -orientierungen ermöglicht. Schlussendlich ist aus vorhandenen Korrespondenzinformationen eine physikalisch korrekte Ansicht mit beliebiger virtueller Kameratransformation durch trifokalen Transfer denkbar. Die Einschränkung einiger Verfahren des Standes der Technik auf die Basislinie bzw. eine reine Translation der virtuellen Kamera soll in dieser Arbeit nicht existieren. Daher bleiben als Synthesemethoden 3D-Warping-Verfahren oder der trifokale Transfer als Option.

Die Anzeige wird zunächst auf eine monoskopische Darstellung beschränkt, da der Hauptzweck des Verfahrens die Blickkorrektur sein soll. Dennoch wäre die Erzeugung einer stereoskopischen Ansicht durch einen zweiten Synthesevorgang problemlos möglich.

Die zuvor gemachten Überlegungen und Festlegungen sind zusammengefasst in Tabelle 3.1 dargestellt. Als Basissystem für die Implementierung der Verarbeitungskette dient ein handelsüb-

licher PC mit einer leistungsfähigen Grafikkarte. Insbesondere letztere gewinnt im Bereich der Bildverarbeitung seit Beginn des Jahrtausends zunehmend an Bedeutung, da durch die Möglichkeit massiver paralleler Programmierung sehr hohe Beschleunigungen in der Berechnung von Algorithmen möglich sind [Owens, 2005; Owens u. a., 2007]. Auch die Parallelisierung auf der CPU ist durch immer mehr integrierte Kerne ein wichtiger Aspekt.

## 3.3. Qualitätsbeeinflussende Parameter

Nachdem eine grundlegende Auswahl von Herangehensweise und Algorithmen getroffen wurde, sollen an dieser Stelle bereits qualitätsbeeinflussende Parameter identifiziert und kategorisiert werden. Dies hilft in der weiteren Entwicklung der Algorithmen. Die Fehlertypen und verursachenden Parameter sind dabei bewusst allgemein gehalten.

Rittermann definierte in seiner Dissertation zur Entwicklung eines objektiven Maßes für 3D-Videoobjekte verschiedene Fehlertypen [Rittermann, 2007]. Die Definition erfolgte anhand von Beobachtungen und sollte möglichst viele Analyse-, Repräsentations- und Synthesetypen für 3D-Videoobjekte erfassen. Fehlerursachen wurden qualitativ erfasst und genannt. Für die in Abschnitt 3.1 definierte Verarbeitungskette können die Fehlertypen reduziert bzw. verallgemeinert werden.

In der hier qualitativ vorgenommenen Systematisierung werden grundlegende Ursachen für Bildfehler und Artefakte definiert. Dabei geschieht eine Unterteilung in prinzip- und algorithmenbedingte Ursachen. Prinzipbedingte Ursachen sind solche, die sich bei gewähltem Verfahren selbst mit einem mathematisch-physikalisch korrekten Modell[1] nicht vermeiden lassen. Als algorithmenbedingt werden Ursachen bezeichnet, wenn sie durch Algorithmen verursacht werden. Üblicherweise ist dies der Fall, wenn diese Schätzungen oder angenäherte Modelle benutzen, um das Ergebnis zu erzeugen oder fehlerbehaftete Beobachtungen verwenden. Diese Definitionen basieren auf bekannten und offensichtlichen Sachverhalten aus der Literatur und bestätigenden Beobachtungen während der Arbeit. Die Fehlerausprägungen sind mit üblichen Expertenbegriffen aus der Bild- und Videoanalyse bezeichnet. Ein allgemeingültiger Anspruch kann daraus nicht hergestellt werden, weshalb im späteren Verlauf der Arbeit auch subjektive Tests zum Finden von geeignetem Vokabular vorgestellt werden. Ebenso werden an dieser Stelle nur synthesespezifische Fehler für ein *einzelnes Bild* einer Videosequenz thematisiert. Fehlerausprägungen im temporalen Kontext sind relevant und werden im weiteren Verlauf der Arbeit ebenfalls thematisiert. Abb. 3.3 veranschaulicht in einem Diagramm beispielhaft die Fehlerausprägungen und die Zusammenhänge.

[1] Im Rahmen der durch die Digitalisierung bedingten Grenzen und Ungenauigkeiten

*Aufdeckungen* sind prinzipbedingte Ursachen für Fehler. Sie sind abhängig von der externen Transformation der virtuellen Kamera mit Bezug auf eine *Referenzkamera* des Stereosystems (z. B. der linken Kamera). Sie entstehen, wenn die virtuelle Kamera Bereiche des Objektes abbildet, die in den Stereokameras nicht sichtbar sind. Diese Bildinformationen fehlen in der virtuellen Ansicht und manifestieren sich in Form von *Löchern*. Der Effekt nimmt zu, wenn die virtuelle Kameratranslation $\mathbf{t}$ in $X$- und / oder $Y$-Richtung des Bezugssystems sowie die Rotation $\mathtt{R}$ um diese Achsen zunimmt. Im Kontext der Videokommunikation sind beispielsweise die Seiten des Kopfes (vor allem die Ohren) kritische Bereiche. Werden diese bei der Aufnahme nicht erfasst, können sie in der Ansicht einer seitlich liegenden Kamera nicht synthetisiert werden. Ebenso ist die Halsregion der aufgenommen Personen ein problematischer Bereich, da dieser vom Kinnbereich verdeckt sein kann.

Mit *Re-Sampling* wird im Diagramm das Erzeugen der virtuellen Ansicht durch das Setzen berechneter Pixelposition durch eine bestimmte Synthesemethode bezeichnet. Es ist der eigentlichen Vorgang der Synthese und als prinzipbedingte Ursache unvermeidbar. Übliche Annahme dabei ist, dass die Bildgröße[2] der Referenz- und virtuellen Kamera identisch ist. Auch hier besteht offenkundig eine Abhängigkeit von der externen virtuellen Kameratransformation, insbesondere der Tiefe in Richtung der $z$-Achse. Wird die virtuelle Kamera näher an das Objekt herangeführt als die Referenzkamera, so entstehen Löcher durch fehlende Bildinformationen. Der Vorgang ist mit einer Bildskalierung zu vergleichen, bei der ebenso Lücken entstehen, die üblicherweise durch Interpolation geschlossen werden. Bei einer 2D-Bildskalierung sind diese Löcher gleichmäßig verteilt. Beim hier gemeinten Re-Sampling ist die Verteilung von der virtuellen Kameraposition abhängig. Die durch Re-Sampling auftretenden Löcher haben einen anderen visuellen Charakter als die durch Aufdeckungen entstehenden. Erstere sind eher als streifen- oder gitterförmig zu bezeichnen, letztere als großflächiger. Bei einer Entfernung der virtuellen Kamera vom Objekt können *Aliasing-Effekte* auftreten. Die Neuzuordnung der Pixel ist nun mehrdeutig und es entstehen Überlappungen, wodurch es zu hochfrequenten Störungen in der virtuellen Ansicht kommen kann.

Die Stereoanalyse ist als algorithmenbedingte Ursache für Fehler einzustufen, da sie mathematisch ein schlecht gestelltes Problem darstellt. Wie ausführlich in Abschnitt 2.2 beschrieben, bedient sie sich vereinfachter Annahmen und Modelle um die Szenenstruktur aus einer mehrdeutigen, fehlerbehafteten Beobachtung zu bestimmen. Auch bei der Stereoanalyse treten prinzipbedingte Probleme, z. B. durch Verdeckungen auf. Dennoch sind falsch bestimmte Disparitäten durch bessere und komplexere Modelle reduzierbar, wie auch die Geschichte der Entwicklung in diesem Bereich zeigt. Die Stereoanalyse selbst ist grundsätzlich nicht von der Transformation der virtuellen Kamera abhängig. Jedoch manifestieren sich fehlerhaft bestimmte Disparitäten in Abhängigkeit der virtuellen Kameratransformation in verschiedenen Fehlerausprägungen. So

[2] Im Folgenden verwendet, um Verwechslungen mit der „Auflösung“ im Sinne der Pixeldichte zu vermeiden.

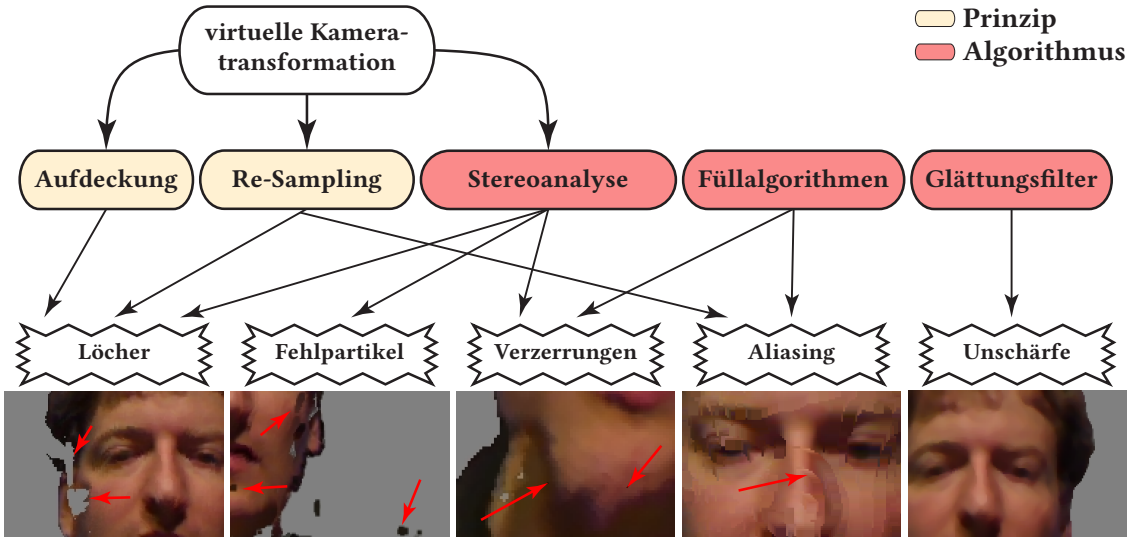



**Abbildung 3.3.:** Allgemeine Qualitätsparameter und Fehlerausprägungen der Verarbeitungskette.

entstehen Löcher, da ein Punkt aufgrund falscher Tiefeninformation auf eine falsche Pixelposition transformiert wird. Die korrekte Pixelposition bleibt ungefüllt. Gleichzeitig kann ein Fehlpartikel innerhalb oder bei extremen Abweichungen auch außerhalb des Objektes entstehen. Findet nur eine gering Fehlverschiebung durch falsche Disparitäten statt, so befinden sich die fehlerhaften Pixel noch nahe ihrer originalen Position. Dadurch entsteht der Eindruck eines Verschiebens der Objektoberfläche, was als *Verzerrung* bezeichnet wird.

*Füllalgorithmen* und *Glättungsfilter* sind Algorithmen, die verwendet werden, um die Wahrnehmung o.g. Fehler zu reduzieren oder zu kaschieren. Sie verwenden dabei ebenfalls unvollständige Annahmen und Modelle. So werden beispielsweise Nachbarschafts- und Kanteninformationen genutzt um Löcher zu füllen. Hochfrequente Strukturen werden unter Annahme einer homogenen Region der Szenenoberfläche geglättet. Dadurch könne auch diese Algorithmen selbst Fehler verursachen. So kann „ungeschicktes“ Füllen ähnlich dem Re-Sampling hochfrequente Strukturen erzeugen. Wird ein Loch über einer Objektkante falsch gefüllt, so ergeben sich Verzerrungen. Glättungsfilter wiederum vermögen Aliasingeffekte zu reduzieren, glätten aber auch abgebildete Szenenstrukturen und erzeugen so *Unschärfe*.

Durch die Aneinanderreihung einzelner synthetisierter Bilder zu einem Video können alle o. g. Fehlertypen weitere Qualitätsreduzierungen erzeugen. Diese temporalen Artefakte sind in Abb. 3.3 nicht dargestellt. Insbesondere Fehler, die starken zeitlichen Änderungen von Bild zu Bild unterworfen sind, wirken sich durch Unruhe und Flackern aus. Sie müssen ebenso wie die statischen Fehler beachtet werden.

Obwohl manche Fehlerausprägungen in ihrer Charakteristik einen Rückschluss auf die Ursache zulassen, trifft dies längst nicht immer zu. Die vielfältigen Zusammenhänge, die in diesem Abschnitt beispielhaft präsentiert wurden, machen offensichtlich, dass eine klare Identifizierung von Problemstellen anhand von Fehlerausprägungen schwierig ist. Diese können sich ähneln oder überlagern. Auch für deren Auswirkung auf die Wahrnehmung der Gesamtqualität können an dieser Stelle keine gesicherten Aussagen gemacht werden. Die Fehler sind, wenn auch bekannt, in ihrer Ausprägung neuartig und für Nicht-Experten unbekannt. Dennoch ist die hier vorgenommene Systematisierung ein wichtiger Schritt für die nachfolgenden Algorithmenentwicklung, da sie wichtige Anhaltspunkte für die Vermeidung der typischsten Qualitätsprobleme gibt.

# 4. Stereoaufnahme

An erster Stelle der im vorangegangen Kapitel definierten Verarbeitungskette steht die Aufnahme von Stereosequenzen. Bei der Auswahl der Webcams für die Verwendung innerhalb dieser Arbeit wurde primär Wert auf eine hohe Bildqualität, Bildrate und Konfigurierbarkeit gelegt. Es sollten handelsübliche Kameras sein, deren Preis im höheren Endverbraucherbereich liegt. Auch sollten baubedingte Unterschiede in der Bildaufnahme minimal sein. Im Detail sind folgende ideale technische Anforderungen zu nennen:

- Geringes Rauschen
- Geringe Artefakte durch Kodierung, Skalierung oder andere vorverarbeitende/integrierte Algorithmen
- Geringe Unterschiede in Farbe und Helligkeit der aufgenommenen Sequenzen zwischen Kameras gleichen Bautyps bei identischen Aufnahmeparametern
- Geringe geometrische Bildverzerrungen durch die Kameraoptik
- Für Videokommunikationsanwendungen notwendige Bildrate und Bildgröße

Selbst Kameras im professionellen Broadcast- oder Industriebereich können nicht all diese Anforderungen zu $100\%$ erfüllen. Umso mehr müssen bei der gewählten Webcam Kompromisse gemacht werden. Die Recherchen zu Beginn der Arbeit führten schnell zu einem Webcam-Modell der Firma Logitech. Die Kamera *QuickCam 9000 Pro* erfreut sich in der Bildverarbeitungsgemeinschaft großer Beliebtheit. Die Kamera bietet eine vergleichsweise realistische Farb- und Helligkeitswiedergabe. Die Optik, wenn auch nicht zu vergleichen mit großformatigen Objektiven, hat gewisse Verzerrungen, die jedoch nicht allzu stark sind. Rauschen ist zu beobachten, lässt sich aber prinzipbedingt bei solch lichtschwachen Objektiven kaum vermeiden, da das Signal verstärkt werden muss. Konkrete technische Daten sind im Consumer-Bereich schwer zu recherchieren. Mittels Kalibrierung und anhand von Informationen andere Nutzer lassen sich die Daten, die in Tabelle 4.1 zusammengetragen sind, herleiten.

Zukünftig werden vermehrt Webcams mit solchen Eigenschaften verfügbar sein. Im Gegensatz zum $4{:}3$-Seitenverhältnis und der niedrigen Bildrate wird sich eine am HDTV orientierte Bildgröße mit $16{:}9$ Seitenverhältnis und $25$ `Bilder/s` durchsetzt. Die Einschränkungen aufgrund kleiner Optiken und Sensoren werden jedoch auch zukünftig bestehen bleiben.

| Parameter | Wert |
|---|---|
| Brennweite | 3.7 mm, fest |
| Sensortyp | CMOS |
| Sensorgröße | 1/3 Zoll, ca. 4.536 mm × 3.416 mm |
| native Bildgröße | UXGA 1600 px × 1200 px, (2 MP) bei max. 5 Bilder/s |
| Schnittstelle | USB 2.0 HighSpeed, Datenrate max. 480 Mbit/s |

**Tabelle 4.1.:** Technische Kenngrößen der verwendeten Kamera Logitech QuickCam Pro 9000

Nachteil aller gängigen Webcams für die gedachte Anwendung ist, dass sie üblicherweise nur als monokulare Kameras verfügbar sind. Eine interessante Alternative bietet die Firma novo, die mit der Kamera *Minoru 3D* als erste eine Stereo-Webcam auf den Markt brachte. Die Kamera integriert zwei Kameras in einem Gehäuse. Die Bildqualität bleibt allerdings hinter der der Logitech Webcams zurück. Auch ist die Kamera für die stereoskopische Aufnahme und Wiedergabe vorgesehen und die Basisbreite zwischen den Stereokameras somit fest vorgegeben. Das schränkt den Spielraum zur Variation der Basisbreite bei der Stereoanalyse ein. Dennoch ist die Nutzung einer solchen Kamera für eine finale Anwendung durchaus sinnvoll. So werden z. B. durch den Nutzer bedingte Probleme bei der Einrichtung einer Stereoanordnung vermieden. Dass System kann als Gesamtlösung vertrieben und eine Kalibrierung der externen Kameraparameter schon werksseitig vorgenommen werden. Jedoch sind höhere Flexibilität und Qualität für diese Forschungsarbeit entscheidend und somit wird die Logitech-Kamera favorisiert.

Das Aufzeichnen von Sequenzen mit 25 Bilder/s bietet die Kamera nur bei einer Bildgröße von 640 px × 480 px und geringer. Ab 800 px × 600 px und aufwärts sind nur noch 15 Bilder/s bzw. 5 Bilder/s möglich. Aufnahmebedingte, negative Qualitätsmerkmale wie wahrgenommenes „Ruckeln“ sollen jedoch für subjektive Tests ausgeschlossen werden. Es wird daher eine Sequenz mit einer Bildwiederholrate genutzt, die auch den gängigen europäischen Fernsehstandards entspricht. Bei der ausgewählten Kamera beschränkt dies die Bildgröße für die aufgenommenen Videos auf 640 px × 480 px. Diese Einschränkung ist jedoch nicht der einzige Grund für die Wahl dieser Bildgröße. Ein weiterer wichtiger Faktor ist die maximal mögliche Datenrate der verwendeten Schnittstelle, in diesem Fall des USB-Busses. Für Testzwecke müssen die Sequenzen auf Festplatte aufgezeichnet werden, was ebenso ein limitierender Faktor ist. Beides spielt insbesondere bei paralleler Aufnahme mit mehreren Kameras eine nicht zu unterschätzende Rolle. Weitere Faktoren ergeben sich aus dem erhöhten Aufwand für die Algorithmen. So erhöht sich bei vielen der entwickelte Algorithmen die Berechnungszeit in einem bestimmten Verhältnis zu den Eingangsdaten, d. h. der Anzahl der Pixel. Das Testen komplexerer Algorithmen mit großen Testdaten-Sets lässt sich oft nur mit kleineren Bildern in realistischer Zeit bewerkstelligen. Eine Verbesserung der Ergebnisse der Algorithmen durch die zusätzlichen Informationen steht dabei

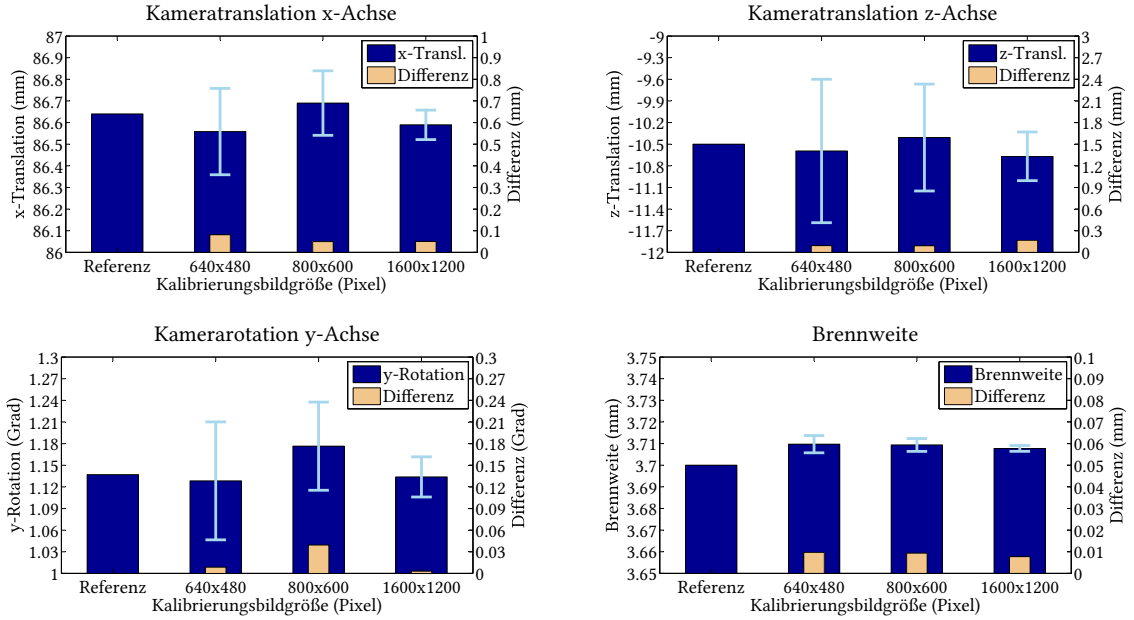
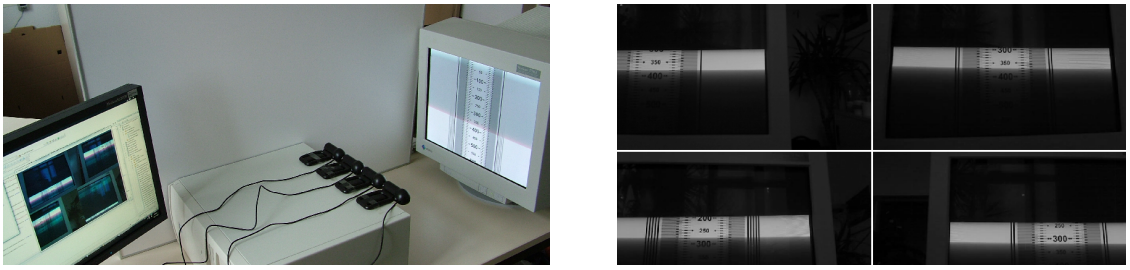



**Abbildung 4.1.:** Kalibrierungsgenauigkeit in Abhängigkeit von der Bildgröße für ausgewählte externe und interne Kameraparameter eines Stereosystems. Dargestellt sind die Ergebnisse im Vergleich zu den Referenzwerten der synthetischen Szene und die jeweilige absolute Differenz zum Referenzwert. Hellblau dargestellt sind die durch Rückprojektion bestimmten Fehlerintervalle der Schätzung.

oft nicht in angemessenem Verhältnis zum Mehraufwand. Dies wird im Folgenden am Beispiel der Kamerakalibrierung gezeigt. Verwendete Methode und Modell entsprechen den in Abschnitt 2.1.4 beschriebenen. In Abb. 4.1 sind auszugsweise Messergebnisse für eine Kamerakalibrierung unter Verwendung verschiedener Bildgrößen gezeigt. Das Experiment wurde mittels einer synthetisch erzeugten Szene durchgeführt. Die Kameras sind daher als ideale Lochkameras ohne jegliche optische Verzerrungen simuliert. Es ist erkennbar, dass der aus der Rückprojektion berechnete Messfehler mit steigender Bildgröße sinkt. Die Differenz zu den Referenzdaten der synthetischen Szene sinkt meist auch. Jedoch liegen die Abweichungen (Fehler sowie Differenz zur Referenz) in einem sehr geringen Bereich, so dass die Messung bereits bei $640\ \text{px} \times 480\ \text{px}$ als sehr genau angesehen werden kann. Reale Linsenverzerrungen können dieses Ergebnis natürlich beeinflussen. Diese Fehler sind dann jedoch unabhängig von der gewählten Bildgröße. Letztlich geht die VGA-Bildgröße auch konform mit den momentan genutzten Bildgrößen in Videokommunikationstools wie Skype. Auch in der Bildverarbeitungsgemeinschaft ist dies eine gängige Wahl für Sequenzen verarbeitende Algorithmen. So sind die Standbilder für die Evaluation von Stereoanalysealgorithmen der Middlebury Stereodatenbank von 2001 und 2003 $450\ \text{px} \times 375\ \text{px}$ groß, obschon neuere Sequenzen mit bis zu $1390\ \text{px} \times 1110\ \text{px}$ verfügbar sind [Scharstein und Szeliski, 2012].



(a) (b)

**Abbildung 4.2.:** (a) Messaufbau zur Synchronizitätmessung der verwendeten Webcams. (b) Beispielaufnahmen von vier Kameras zum Zweck der Synchronizitätmessung.

## 4.1. Synchronizität

Die Algorithmen der Stereoanalyse werden auf die einzelnen Bilder einer Videosequenz angewendet. Die dabei genutzten Modelle verwenden ausschließlich örtliche Zusammenhänge. Es wird implizit davon ausgegangen, dass die Aufnahmen zum gleichen Zeitpunkt stattfinden. Daher erfordert die Verwendung separater Kameras eine Betrachtung der Synchronizität der Aufnahmen. Hochpreisige Industrie- oder Broadcast-Kameras ermöglichen meist eine präzises, extern gesteuertes Auslösen des Belichtungsvorganges (*Triggering*). Über einen zeitlich hoch aufgelösten Taktgenerator kann sichergestellt werden, dass die Belichtung des Sensors aller gesteuerten Kameras in einem bestimmten, relativ zur Bilddauer sehr kleinen Zeitrahmen und somit synchron erfolgt. Im Gegensatz dazu bieten Webcams diese Möglichkeit nicht. Im Rahmen dieser Arbeit wurden diesbezüglich Untersuchungen angestellt, ob auch mittels Webcams eine synchrone Aufnahme möglich ist.

Zur Messung der Synchronizität bedarf es einer zeitlichen Referenz, deren Aufnahme durch die Webcams Rückschlüsse auf den Zeitpunkt der Aufnahme zulässt. Zu diesem Zweck wurde eine in [Wimmer, 2005] vorgestellte Methode verwendet. Ein Röhrenmonitor wird mit den zu messenden Kameras aufgenommen. Da dessen Zeilenfrequenz genau einstellbar und im Vergleich zur Bildrate der Kameras sehr hoch ist, kann über die Position des Elektronenstrahls in der Aufnahme bestimmt werden, ob ein zeitlicher Versatz vorhanden ist. Zu diesem Zweck wird auf dem Röhrenmonitor eine Skala mit den Zeilennummern gezeigt. Abb. 4.2 zeigt den Aufbau des Experimentes. Bei geringer Belichtungszeit der Kameras ist die Zeilenposition aus den Aufnahmen ablesbar und die zeitliche Latenz zwischen den Kameras berechenbar.

Eine Reihe von zehn Messungen ergibt, dass die eine Synchronizität nur mit der Genauigkeit der Dauer eines Bildes erreicht werden kann. Die gemessenen Verzögerungen von drei Kameras mit Bezug auf eine Referenzkamera variierten über alle Tests zufällig innerhalb eines Bereiches von 40 `ms` bei einer eingestellten Bildrate von 25 `Bilder/s`. Mitunter ist zu beobachten, dass zwei Kameras synchron laufen, jedoch in Bezug auf die anderen eine Latenzzeit aufweisen bzw.

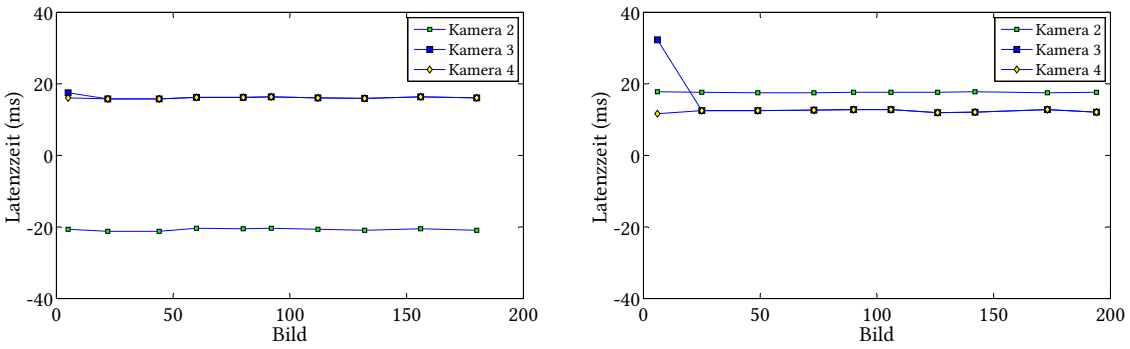



**Abbildung 4.3.:** Ergebnisdiagramme zweier Synchronizitätsmessungen. Die Messungen erfolgten mit Bezug auf eine Referenzkamera. Es ist zu erkennen, dass die Beträge der Latenzzeiten stets kleiner als $40$ ms sind. Eine genaue Synchronisierung innerhalb dieser Bilddauer ist mit Webcams jedoch nicht erreichbar. Es ist auch erkennbar, dass die Initialisierung von Kamera 3 und 4 zu einer zufälligen Synchronisierung dieser beiden führt, was jedoch bei anderen Messungen nicht der Fall und somit nicht deterministisch ist.

die Belichtung vor diesen ausführen. Abb. 4.3 zeigt zwei Messdiagramme, die den Sachverhalt verdeutlichen.

Die Ansteuerung der Webcams erfolgt mittels PC über die gängigen Software-Schnittstellen der Betriebssysteme. Verschiedene Versuche, durch Maßnahmen der Programmierung eine Synchronizität zu erzeugen, waren nicht erfolgreich. So wurde versucht, die Initialisierung der Kameras sowie den Beginn der Aufnahmen durch parallele Ausführung zeitgleich durchzuführen (*Multi-Threading*). Auch mit Mitteln der Programmierschnittstellen war nur eine bildgenaue Synchronisation zu erreichen. Es bleibt daher festzuhalten, dass über Standard-Schnittstellen angesteuerte Webcams im ungünstigsten Falle eine Latenzzeit $t_l$ von einer Bilddauer haben können, im konkreten Fall: $t_l < 40$ ms. Recherchen ergaben, dass höchsten mit einem Eingriff in die Treibersoftware der Kameras eine bessere Synchronisation erfolgen könnte, was jedoch bei proprietärer Software nicht möglich ist und selbst bei offenen System eine immensen Aufwand ohne Erfolgsgarantie bedeuten würde. Auch eine Lösung durch Modifikation der Hardware ist bei diesem Typ Kamera nicht vorgesehen und wurde daher nicht weiter verfolgt. Eine testweise Messung mit der Stereokamera Minoru 3D ergab, dass selbst deren Kameras intern nicht synchronisiert werden. Hardwareseitig liegt deren Vorteil demnach allein in der festen Anordnung im Gehäuse. So haben diese Kamera auch bzgl. dieses Parameters keinen Mehrwert gegenüber der Logitech-Kamera. Die in dieser Arbeit vorgenommenen Untersuchungen zur Aufnahmesynchronisierung wurden als Bestandteil der Arbeiten in [Weigel u. a., 2010] veröffentlicht.

## 4.2. Bustransfer und Datenspeicherung

Das in dieser Arbeit untersuchte Anwendungsszenario der Videokommunikation bedarf prinzipiell keiner Speicherung von Bilddaten, da es sich um eine Echtzeit-Kommunikation handelt. Die Entwicklung von Algorithmen erfordert jedoch die Speicherung von Videosequenzen um nachvollziehbare Analysen vorzunehmen. Neben den Daten der Stereokameras werden parallel dazu die Sequenzen zweier zusätzlicher Kameras aufgezeichnet, die zum späteren Vergleich dienen (vgl. Abschnitt 4.4).

Die Verwendung von vier Webcams an einem Standard-PC erzeugt hohe Datenraten. Unter der Annahme unkomprimierter RGB-Daten, bei denen jeder Farbkanal mit $8$ `bit` kodiert wird, erzeugt ein Video der Größe $640$ `px` $\times$ $480$ `px` bei $25$ `Bilder/s` bereits eine Datenrate von $184.32$ `Mbit/s`. Die maximale Datenrate der verwendeten USB 2.0 „High-Speed“-Schnittstelle ist mit theoretischen $480$ `Mbit/s` für alle am USB-*Controller* angeschlossenen Geräte angegeben [USB Implementers Forum, 27.04.2000]. Die Netto-Datenraten liegt im verwendeten isochronen Übertragungsmodus für Videoströme zusätzlich um $18.1\%$ niedriger und somit bei ca. $390$ `Mbit/s`. Andere Quellen und eigene Messungen ergeben eine reale Netto-Datenrate von gerade einmal $55\%$ was $264$ `Mbit/s` entspricht. Es ist offenkundig, dass vier unkomprimierte Videoströme nicht über einen USB-Controller übertragen werden können. Der Kamerahersteller nutzt daher die M-JPEG Kompression[1] in der Kamera. Dies reduziert die Datenrate, birgt aber die Gefahr der Entstehung von Kompressionsartefakten. Bei stichprobenartigen Messungen der USB-Transferrate wurde festgestellt, dass die Kamera bei der Initialisierung die verfügbare Datenrate abfragt und anschließend einmalig die Kompressionsstärke anpasst. Dadurch kann es vorkommen, dass das Bild der letzten Kamera stark komprimiert wird, da dieser eine geringere Datenrate zur Verfügung steht. Eine adaptive Anpassung der Datenrate oder eine intelligente Verteilung während der Aufnahme durch alle Kameras war nicht zu beobachten. Als Konsequenz daraus bleibt festzuhalten, dass die Kameras sinnvoll auf verschiedene USB-Controller verteilt werden müssen. Der PC muss die entsprechende Hardware beinhalten, üblicherweise sind zwei High-Speed Controller vorhanden. Eine solche Anbindung erzeugt keine sichtbaren Kodierungsartefakte. Die Kameras nutzten dann eine Datenrate von durchschnittlich $124$ `Mbit/s`. Somit steht für jedes Einzelbild mehr als die Hälfte der unkomprimierten Datenmenge zur Verfügung, was einer sehr hohen JPEG-Kompressionsqualitätsstufe entspricht. Mit der Einführung der USB 3.0 Schnittstelle und entsprechender Kameras dürfte die Beschäftigung mit dieser Problematik der Vergangenheit angehören.

Die eingehenden Videodaten werden bereits vom Treiber bzw. der Softwareschnittstelle dekomprimiert und als RGB-Daten an die aufnehmende Applikation übergeben. Für die Speicherung gilt es daher, die Daten mit den Raten des unkomprimierten Materials auf Datenträger zu speichern.

[1] Eine Folge von JPEG-komprimierten Bildern

Diese $184,32$ `Mbit/s` bzw. $23,04$ `MB/s` pro Videostrom addieren sich zu $92,16$ `MB/s`, die auf das Speichermedium geschrieben werden müssen. Da dies im Grenzbereich der verwendeten Festplatten liegt, wurden folgenden Maßnahmen getroffen um Bildausfälle durch den Speichervorgang zu vermeiden:

- Zusammenfassen zweier Ströme zu einem großen Strom durch vertikales Zusammenfügen der Bilder. Dadurch können größere Datenmengen auf einmal auf die Festplatte geschrieben werden. Dies fördert ein kontinuierliches Schreiben auf den Datenträger.
- Verteilung der Ströme auf zwei separate Datenträger. Die Datenrate ist dadurch im gängigen Bereich der verwendeten Festplatten.
- Zur Vermeidung von Verlusten durch nicht deterministische, kurzzeitige Aussetzer beim Schreibvorgang, wird eine durch den Arbeitsspeicher gepufferte Speicherung implementiert, die bis zu $100$ Bilder puffern kann.

Durch diese Maßnahmen kann ein durch die Speicherung bedingter Ausfall von Bildern vermieden werden. Dennoch kann durch andere Systemeinflüsse wie kurzzeitiger Belastung des Busses oder unvorhergesehene Effekte in den Kameras ein Bildausfall passieren. In diesem Fall wird für diese Arbeit definiert, das vorherige Bild beizubehalten, bis wieder ein aktuelles Bild verfügbar ist. Dies ist gängige Praxis in der Videokommunikation und erzeugt bei kurzzeitigen Ausfällen weniger auffällige Störungen als z. B. die Einblendung eines einfarbigen oder schwarzen Bildes.

## 4.3. Kameraanordnung

Die Anordnung der Stereokameras am Bildschirm ist ein wichtiger Aspekt des Systems. Die Lage der Kameras zueinander beeinflusst Konvergenzwinkel und Basisabstand, was hinsichtlich Rektifizierung, Disparitätsbereich und Verdeckungsproblematik beeinflussende Parameter sind. Die Lage am Bildschirm bzgl. der aufgenommenen Person sowie der virtuellen Kamera ist wichtig, da sie die Ursache für Aufdeckungen im virtuellen Bild sein kann, was zu Löchern führen würde (vgl. Abschnitt 3.3).

In der relevanten Literatur werden verschiedene Kameraanordnungen genutzt (vgl. Abschnitt 2.4). Grundsätzlich sind drei Anordnungen zu nennen, die dann im Detail variiert werden können. Sie sind in Abbildung 4.4 dargestellt. Die Anordnung in Abbildung 4.4a orientiert sich an der üblichen Anordnung bei Videokommunikation im Heimbereich. Die Kameras befinden sich nebeneinander *auf* dem Bildschirm. Der Basisabstand ist somit frei einstellbar. Die Person wird frontal bzw. leicht von oben aufgenommen. Die Anordnung bedarf keiner zusätzlichen Vorrichtungen, da ein Großteil der Webcams für diese Anbringung konzipiert sind. In Notebooks ist die (monoskopische) Kamera baubedingt zumeist auch an dieser Stelle platziert. Mit mehr mechanischem Aufwand ließe sich diese Anordnung auch unterhalb des Bildschirms realisieren,

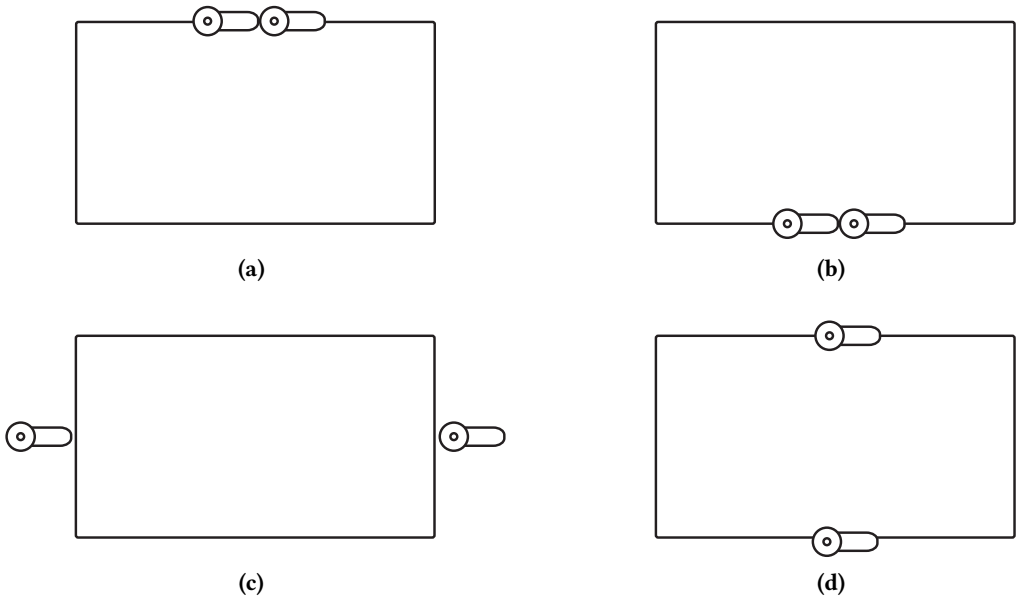



**Abbildung 4.4.:** Prinzipielle Anordnungsmöglichkeiten für Stereokameras zum Zweck der Blickkorrektur.

wie in Abb. 4.4b gezeigt. In Anordnung 4.4c sind die Kameras seitlich des Bildschirms platziert. Der Basisabstand ist somit durch die Breite des Bildschirms bestimmt und kann nicht reduziert werden. Eine vertikale Positionierung kann frontal zur Person erfolgen. Zusätzliche Halterungen zur Anbringung können vonnöten sein. Anordnung 4.4d platziert die Kamera oberhalb und unterhalb des Monitors. Es gelten die gleichen Aussagen bzgl. Basisabstand und Anbringung wie für 4.4c.

Hinsichtlich des Einflusses auf die Verarbeitungskette ist festzustellen, dass ein größerer Basisabstand zu größeren verdeckten Bereichen zwischen den Bildern führt. Dies wäre bei Anordnung 4.4c und 4.4d der Fall. Dadurch ergeben sich Probleme bei der Stereoanalyse insbesondere in den seitlichen Randbereichen des Kopfes. Es können keine Disparitätsinformationen in diesen Bereichen bestimmt werden und müssen auf andere Art und Weise gewonnen werden. Der ebenfalls erhöhte Disparitätsbereich kann durch eine konvergente Anordnung der Kameras verringern werden. Dies führte jedoch zu einer aufwändigeren und fehleranfälligeren Verarbeitung bei Rektifizierung und 3D-Warping. Abb. 4.5 veranschaulicht beide Sachverhalte an einem Bildbeispiel. Mittels der flexiblen Varianten 4.4a und 4.4b ist dieser Einfluss des Basisabstandes variierbar. Vorteil der Anordnung mit großem Basisabstand ist, dass eine einfache Interpolation oder View-Morphing ausreicht um eine Ansicht auf der Basislinie zu erzeugen. Bei horizontal bzw. vertikal zentraler Platzierung in Anordnung 4.4c und 4.4d entspricht diese oft der den Blick korrigierenden Ansicht, was von einigen bestehenden Verfahren genutzt wird.

**Abbildung 4.5.:** Ausschnitte aus Stereo-Beispielaufnahme mit großem Basisabstand. Oben wird eine achsenparallele Anordnung genutzt. Verdeckungen und großer Disparitätsbereich sind klar erkennbar. Der Disparitätsbereich kann durch eine konvergente Anordnung reduziert werden. Die Verdeckungsproblematik bleibt bestehen. Bei beiden Aufnahmen hatten die Kameras eine Abstand von ca. 34 cm und die Person saß ca. 75 cm entfernt. Dies entspricht dem Abstand bei heute gängigen 19 ″-Monitoren mit normalen Betrachtungsabstand.

Die Verdeckung in der Stereoanalyse bei den Anordnungen oberhalb oder unterhalb des Bildschirms kann durch den variablen Basisabstand minimiert werden, ist aber dennoch vorhanden. Auch der Disparitätsbereich ist selbst bei achsenparalleler Kameraanordnung kontrollierbar. Insbesondere in Hinblick auf größer werdende Monitore, aber auch Geräte, zu denen der Betrachtungsabstand der Person geringer ist, ist dies ein Vorteil. Nachteilig ist die höhere Anfälligkeit für Aufdeckungen in der Synthese. Bei der gängigsten Anordnung 4.4a liegt die virtuelle Ansicht stets unterhalb der Stereokameras. Nehmen diese die Person von oben auf, so werden bei der Synthese Bereiche im Halsbereich aufgedeckt, die vom Kinn verdeckt wurden. Für die Anordnung unter dem Bildschirm trifft dies für den Haarbereich zu.

Diese Überlegungen ergeben in Kombination mit nicht determinierbaren Faktoren wie der Position und Haltung der Person die Entscheidungsparameter für die Auswahl einer Stereokameraanordnung. Obwohl das Problem der Aufdeckung durch die Anordnungen 4.4a und 4.4b nicht ignorierbar ist, so sprechen Praktikabilität, Flexibilität und insbesondere die Vorteile bei der Stereoanalyse für diese. Da der Anspruch der Arbeit auch darin besteht, nicht nur eine Interpolation oder ein View-Morphing auf der Basislinie umzusetzen, sondern eine „frei“ bewegliche virtuelle Kamera zu synthetisieren, spricht dies umso mehr für die Varianten 4.4a und 4.4b. Die notwendige Extrapolation der virtuellen Ansicht wird als Alleinstellungsmerkmal des entwickelten Verfahrens angesehen. Daher wird sich für die Verwendung von Anordnung 4.4a entschieden. Sie bietet die flexibelsten Einstellmöglichkeiten bei einfachem Aufbau, so dass nachteilige Aspekte für die Stereoanalyse minimiert werden können. Eine Zusammenfassung der einzelnen Entscheidungskriterien ist noch einmal in Tabelle 4.2 gegeben.

| Variante | prakt. Aufbau | Positiv | Negativ |
|---|---|---|---|
| 4.4a & 4.4b | einfach bzw. mäßig einfach | variabler Basisabstand, geringe Verdeckungen, geringer Disparitätsbereich | keine Interpolation oder View-Morphing, mögliche Aufdeckungen im Haar- bzw. Halsbereich |
| 4.4c & 4.4d | schwierig | Interpolation oder View-Morphing möglich | fester Basisabstand, starke Verdeckungen, großer Disparitätsbereich oder konvergente Anordnung |

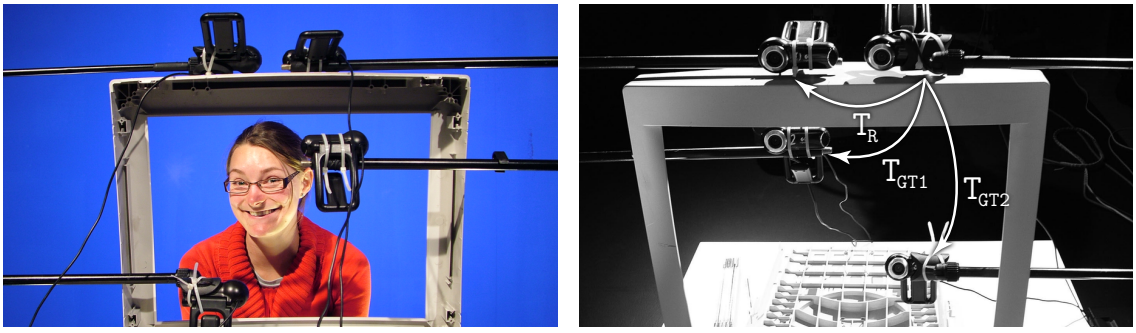
**Tabelle 4.2.:** Entscheidungsparameter bei der Stereokameraanordnung

## 4.4. Erzeugung von Testdaten

Die Entwicklung von Algorithmen verlangt nach einem umfangreichen Testdatenset. Es sollten kurze Sequenzen entsprechend der zuvor gemachten Festlegungen zu Kameraparametern, Kameraanordnung und Szeneaufbau sein. Die aufgenommenen Szenen sollten einer realen Videokommunikationssituation weitestgehend entsprechen. Da mit den Ergebnissen der Arbeit subjektive Test durchgeführt werden sollen, müssen eine Vielzahl von Szenen mit unterschiedlichen Personen erzeugt werden. Ein problematisches Thema bei der Erzeugung virtueller Kameraansichten ist eine fehlende Referenz zum Qualitätsvergleich. Üblicherweise werden Ansichten erzeugt, die von keiner realen Kamera aufgenommen wurden. Dieses Problem soll innerhalb dieser Arbeit zumindest ansatzweise reduziert werden.

Um den Darstellern in den Aufnahmen für das Testdatenset einen möglichst realen Eindruck zu vermitteln, wurde ein alter Röhrenmonitor auf einem Tisch platziert. Zur Erzeugung zweier Referenzansichten wurde die Röhre aus dem Gehäuse entfernt. Der Bildschirmrahmen dient so weiterhin zur Orientierung der Schauspieler, es lassen sich jedoch reale Kameras an der Stelle positionieren, an der im normalen Anwendungsszenario das Kommunikationsfenster sein könnte. Die Aufnahmen wurden unter reproduzierbaren Beleuchtungsbedingungen im Studio vor einem blauen Hintergrund durchgeführt. Die Blauwand dient zur vereinfachten Segmentierung der Person im nachfolgenden Verarbeitungsprozess. Gründe und Details dazu finden sich im Abschnitt 5.1.1. Die Szenen sind zwischen 20 s und 30 s lang, was bei gewählter Bildrate von 25 Bilder/s 500 und 750 Bildern entspricht. Die Schauspieler spielten in dieser Zeit eine simulierte Videokommunikationssituation mit Dialog. Dabei gab es Aufnahmen mit Blick in die Referenzkameras (dem Kommunikationsfenster) und mit wanderndem Blick auf andere Bereiche des Bildschirms.

Nach Aufbau oder Veränderung des Setups durch Umbau oder kleinere Versehen, wie Anstoßen durch Schauspieler, wurden zunächst Kalibrierungsaufnahmen durchgeführt. Neben der für




(a) (b)

**Abbildung 4.6.:** Aufnahmesetup zur Testdatenerzeugung. In Abb. (b) sind die Transformationen der Kameras mit Bezug auf die linke Kamera markiert. Index „R“ steht für rechts, „GT“ für *ground truth* und meint die Referenzkameras.

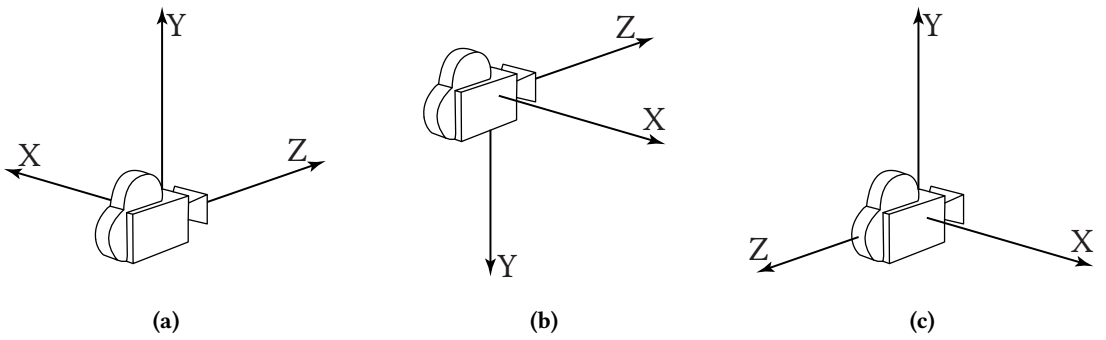
die externe Kalibrierung notwendigen Aufnahmen einer Schachbrett-Fläche wurde auch eine Weißfläche aufgenommen, um nachfolgend ggf. Farbanpassungen vorzunehmen. Neben den Videoaufnahmen wurden ebenso Audioaufnahmen durchgeführt. Wenn auch nicht innerhalb der Arbeit verwendet, so bieten sie eine Möglichkeit spätere subjektive Tests um diese Modalität zu erweitern. Die Synchronisation der Video- und Audioaufnahmen erfolgte über eine vor jeder Aufnahme benutzte Klappe. Diese dient gleichzeitig zur groben Überprüfung der Synchronizität der vier Videoströme.

Die Sequenzen liegen nach der Aufnahme als Einzelbilder im PNG-Format vor, da es eine verlustlose Kompression erlaubt. Sie wurden in einer vordefinierten Verzeichnisstruktur mit einheitlichen Namenskonventionen abgelegt. Dies macht eine Wiederverwendung sowie die automatisierte Bearbeitung durch Skripte erheblich einfacher. Da die Sequenzen der Forschungsgemeinschaft zur freien Verfügen stehen sollen, wurden sie mit kurzen Kommentaren versehen. Mittels eines automatisierten Vorganges wurde eine HTML-Übersicht erstellt, die einen schnelle Übersicht sowie einen kurzen Videoclip zur Orientierung bietet[2]. Neben den Videodaten werden ebenfalls Metadaten aus der Kalibrierung für jede Sequenz gespeichert. Somit entstanden über 50 verwendbare Testsequenzen.

## 4.5. Kamerakalibrierung

Zur Umsetzung des Vorhabens ist es vonnöten, die Parameter der Kameras entsprechend des in Abschnitt 2.1.3 gewählten Lochkameramodells zu bestimmen. Die internen Kameraparameter, namentlich Brennweite $f$, und Kamerahauptpunkt $(p_x, p_y)^\top$ und ggf. Scherungsparameter müssen bei der Verwendung immer derselben Kameras nur einmal bestimmt werden. Dem entgegen

[2]Die Übersicht ist auf der DVD der Druckversion enthalten.




(a) (b) (c)

**Abbildung 4.7.:** Lage verwendeter Kamerakoordinatensysteme. (a) Allgemeines System dieser Arbeit (auch in [Hartley und Zisserman, 2008]) (b) System der Matlab Camera Calibration Toolbox (c) System von OpenGL. Alle Koordinatensysteme sind rechtshändig.

sind externe Kameratransformationen bei jeder Änderung des Aufnahmesetups neu zu ermitteln. Im finalen Anwendungsszenario bestehen diese nur zwischen der linken und der rechten Kamera des Stereopaares. Für die Testdaten sind zusätzlich die Transformationen für die bis zu zwei Referenzkameras zu bestimmen. Alle Transformationen sind euklidische Transformationen, bestehend aus einem Translationsvektor $\mathbf{t}$ und einer Rotationsmatrix $\mathtt{R}$. Im projektiven Raum lassen sich diese in der $3 \times 4$ Transformationsmatrix $\mathtt{T}$ zusammenfassen:

$$\mathtt{T} = \left[\mathtt{R}|\mathbf{t}\right] \tag{4.1}$$

Für den Zweck der Parameterbestimmung wird die etablierte „Camera Calibration Toolbox for Matlab“ eingesetzt [Bouguet, 2010]. Mittels mehrerer Einzelaufnahmen werden auf Basis der Abbildung eines planaren Schachbrettmusters sowie dessen gegebenen Größenverhältnissen das Kameramodell und die Linsenverzerrung komplett bestimmt. Referenzen für die Details zum verwendeten Minimierungsalgorithmus sind in Abschnitt 2.1.4 aufgeführt. Die Toolbox ermittelt die Brennweite der Kamera in der Einheit von `Pixel`. Alle translatorischen Maße sind in `mm`, alle Rotationswinkel in `rad` angegeben. Ebenso wird die Genauigkeit der Berechnung mittels eines Fehlerintervalls angegeben. Diese wird bestimmt, indem die auf der Fläche liegenden bestimmten 3D-Punkte mittels des berechneten Modells auf die Kalibrierungsbilder projiziert werden und anschließend der Versatz zu den originalen 2D-Koordinaten bestimmt wird.

Da die Angabe der externen Transformation mit Bezug auf ein dreidimensionales Koordinatensystem erfolgt, ist die Kenntnis von dessen Lage in Bezug auf die Kamera notwendig. Innerhalb der Arbeit und auch im Großteil der Literatur wird ein rechtshändiges Kamerakoordinatensystem entsprechend Abbildung 4.7a verwendet. Innerhalb der Toolbox ist die Ausrichtung bezogen auf die Kameras eine andere. Sie wird in Abb. 4.7b dargestellt. Da innerhalb der Arbeit auch OpenGL zum Einsatz kommt, ist auch das dort definierte Koordinatensystem relevant, welches in Abb.

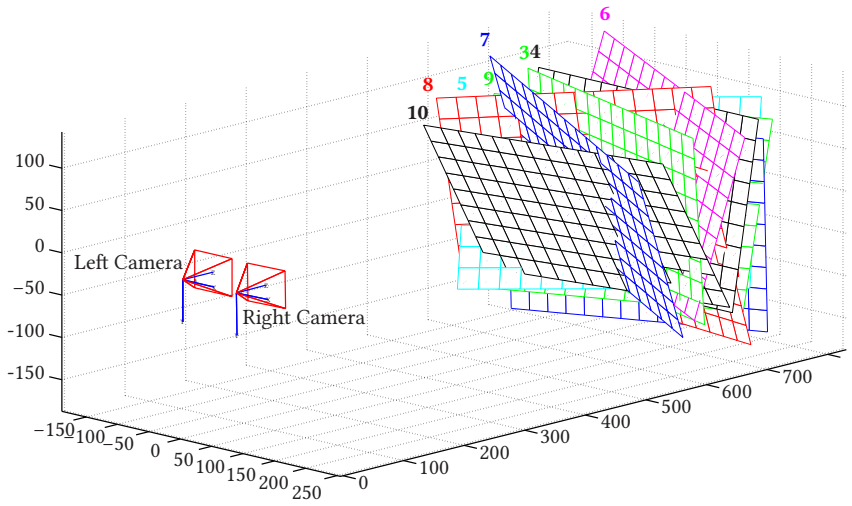



**Abbildung 4.8.:** Ergebnis einer Stereo-Kamerakalibrierung. Dargestellt ist die visuelle schematischen Ausgabe der Toolbox mit den Positionen des Stereokamerapaares und der verwendeten Schachbrettebenen.

4.7c dargestellt ist. Die Umrechnung von Koordinaten zwischen den Koordinatensystemen ist trivial, es wird jedoch stets beachtet, dass verschiedene Systeme Verwendung finden.

Die Toolbox unterstützt die externe Kalibrierung nur für Stereokamerapaare (vgl. Abb. 4.8). Nach der separaten internen Kalibrierung für jede einzelne Kameras erfolgt die Positionsbestimmung der rechten sowie ggf. vorhandener Referenzkameras paarweise. Bezugspunkt ist dabei stets das Koordinatensystem der linken Kamera des Stereopaares (vgl. Abb. 4.6b). Die Toolbox liefert als Ergebnis nicht die Transformationsmatrix für die jeweils anderen Kameras, sondern die *Koordinatentransformation* zwischen den Systemen. Dies ist die Transformation, die z. B. auf Punkte im Koordinatensystem der linken Kamera angewendet werden muss, um deren Koordinaten im System einer anderen Kamera zu erhalten. Um die Transformation dieser Kamera $\mathtt{T_{kam}}$ zu erhalten, muss diese erst aus der Koordinatentransformation $\mathtt{T_{koord}}$ berechnet werden:

$$\mathtt{T_{kam}} = \mathtt{T_{koord}}^{-1} \tag{4.2}$$

Die Ergebnisse der Kalibrierung werden sowohl zur Vorverarbeitung als auch zur Synthese herangezogen. In einem realen Anwendungsszenario ist diese Art der Kalibrierung mit einem Schachbrettmuster dem Nutzer nur schwerlich zumutbar. Während die interne Kalibrierung für die Kameras werksseitig kontrollierbar ist, ist die externe Kalibrierung von lokalen Faktoren abhängig. Neben der Lagebeziehung des Stereopaares ist auch Kenntnis über die Position des Kommunikationsfensters bzgl. der linken Kamera sowie der kommunizierenden Person notwendig. Erstere wiederum ist nur in Verbindung mit den Ausmaßen des Monitors zu bestimmen. Nur mit diesen Informationen lassen sich Position und Orientierung der virtuellen Kamera

vollständig so bestimmen, dass einerseits Blickkontakt hergestellt wird, andererseits die Person noch vollständig abgebildet ist. Im Rahmen dieser Arbeit wurden Untersuchungen angestellt, welche Informationen mindestens notwendig sind und wie diese erhoben werden können. Das Ergebnis wurde in [Korn, 2009] vorgestellt. Bei der Methode sind jedoch noch immer zu viele Messungen seitens des Nutzers notwendig. Da die Usability des Endsystems nicht primärer Fokus dieser Arbeit ist, wird auf eine weitere Betrachtung verzichtet. Mögliche Ausgangspunkte wären die Verwendung von restriktiveren Hardware-Setups, Selbstkalibrierung und die Verwendung von Gesichtserkennung zur Ausrichtung der virtuellen Kamera.

# 5. Entwicklung eines Verfahrens zur Blickkorrektur mittels Bildsynthese

Im vorherigen Kapitel wurde die Stereoaufnahme im Detail beschrieben. Die daraus erzeugte primäre Repräsentation dient nun als Ausgangspunkt für den nächsten Schritt der Verarbeitungskette, die Vorverarbeitung. Zu dieser gehören alle bildverändernden und analysierenden Prozesse, die zur sekundären Repräsentation führen. Das sind die Segmentierung des Vordergrundes, die Rektifizierung, die Stereoanalyse sowie die die Disparitätskarten verbessernden Maßnahmen. Um die Übersichtlichkeit zu wahren und aufgrund ihrer Wichtigkeit, wird die Stereoanalyse separat in Abschnitt 5.2 behandelt.

Die Erzeugung der primären Zielrepräsentation durch die Bildsynthese ist Gegenstand des Abschnitts 5.3. Er beschreibt sowohl die entwickelten Syntheseverfahren sowie nachverarbeitende Maßnahmen zur Verbesserung der primären Zielrepräsentation. Im Verlauf der Arbeit hat sich insbesondere dieser Schritt als sehr wichtig für die Qualität der erzeugten Ansicht erwiesen.

## 5.1. Vorverarbeitung

Die in diesem Abschnitte beschriebenen Maßnahmen behandeln alle vorverarbeitenden Algorithmen zur Erzeugung der sekundären Repräsentation mit Ausnahme der Stereoanalyse selbst, der ein separater Abschnitt gewidmet ist. Die zuerst durchgeführte Segmentierung dient der Separation der kommunizierenden Person vom Hintergrund. Dies entspricht der Idee von 3D-Video*objekten*. In den folgenden Verarbeitungsschritten wird mit dem Vordergrundobjekt sowie mit der assoziierten Alphamaske als Ergebnis der Segmentierung gearbeitet. Es folgt die Rektifizierung der Stereosequenzen. Primäres Ziel dieses Schrittes ist die Beschränkung der Korrespondenzsuche auf waagerechte Epipolarlinien. Eine Filterung der Eingangsbilder zur Reduzierung von granularem Rauschen ist die letzte Maßnahme vor der Stereoanalyse.

### 5.1.1. Segmentierung

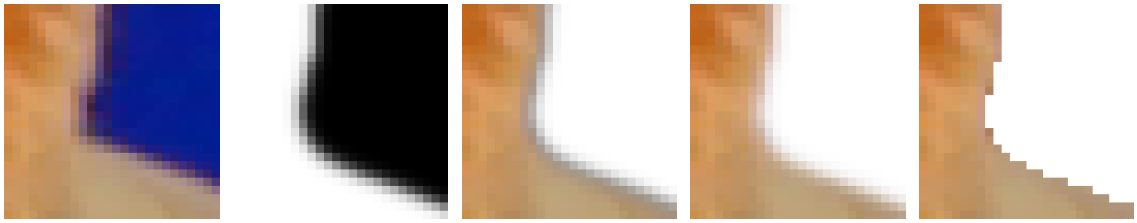
Die Trennung der Person im Vordergrund von deren Hintergrund erzeugt eine Alphamaske als zusätzliche Information. Sie definiert mittels Graustufenwerten die Transparenz des Vordergrund-

objektes, wobei der maximale Wert komplette Opazität, der minimale vollständige Transparenz bedeutet. Im Kontext der Videokommunikation hat dies folgende Vorteile:

- Tiefensprünge zwischen Vordergrund und Hintergrund können klar identifiziert werden. Dies unterstützt den Stereoanalysealgorithmus.
- Bei alleiniger Analyse und Synthese des Vordergrundes entstehen keine Aufdeckungen durch die mitunter starken Tiefenunterschiede zwischen Vorder- und Hintergrund.
- Die Beschränkung der Algorithmen auf den Vordergrund reduziert die Anzahl der Rechenoperationen und somit die Laufzeit der Algorithmen.

Ein Nachteil, der durch diese Maßnahme entstehen kann, ist der mögliche Verlust der Plausibilität der Szene. Auch können Fehler in der Alphamaske in die synthetisierte Ansicht propagiert werden und so deren Qualität mindern. Der erste Nachteil kann durch die Kombination der virtuellen Ansicht mit dem originalen Hintergrund kompensiert werden. Der Hintergrund hat in Relation zum Vordergrund zumeist einen großen Abstand zur virtuellen Kamera. Eine Synthese des Bildes der vergleichsweise gering transformierten virtuellen Kamera würde eine geringe Verschiebung des Hintergrundes bedeuten. Somit könnte der von einer Stereokamera aufgenommene Hintergrund ohne größeren Plausibilitätsverlust auch in der virtuellen Ansicht verwendet werden. Voraussetzung ist allerdings, dass dieser vollständig vorliegt, z. B. durch eine vorherige Aufnahme ohne die Person im Vordergrund. Ebenso könnte ein komplett anderer Hintergrund genutzt werden, der jedoch bezüglich Beleuchtungs- und Größenverhältnissen sowie der Perspektive der virtuellen Kamera entsprechen muss.

Propagierte Fehler der originalen Alphamaske sind nicht ohne Weiteres kompensierbar. Daher hat es Priorität, diese bereits bei der Segmentierung weitestgehend auszuschließen. Da innerhalb dieser Arbeit Fokus auf der Stereoanalyse und Bildsynthese liegt, sollen ebensolche Fehler weitestgehend ausgeschlossen werden. Daher wurde entschieden, die Segmentierung mittels des etablierten Verfahrens des farbbasierten *Keyings* durchzuführen. Die dafür notwendigen Voraussetzungen während der Aufnahme, wie ein möglichst monochromatischer Hintergrund, wurden entsprechend umgesetzt (vgl. Abschnitt 4.4). Die Segmentierung der Sequenzen erfolgte somit manuell mittels gängiger professioneller *Compositing-Software* wie *discreet combustion* oder *Adobe After Effects*. Diese Werkzeuge nutzen modifizierte und weiterentwickelte Methoden wie Chroma-Keying, Farbdifferenz-Keying oder 3D-Keying. Kern aller Methoden ist die Separation der Hintergrundfarben vom Vordergrundobjekt. Dabei kommen verschiedene Methoden zum Einsatz, die zumeist im $RGB$ oder $YC_RC_B$-Farbraum arbeiten. Eine gute Übersicht über grundlegende Verfahren bietet [Smith und Blinn, 1996]. Konkrete Details der Umsetzung sind aufgrund proprietärer Software schwer zu erhalten und sollen daher nicht Gegenstand weiterer Betrachtungen sein. Die erzeugten Alphamasken sind von hoher Qualität im Sinne der Anwendung im Broadcastbereich und mit verhältnismäßig geringem Aufwand erzeugbar. Die Alphamaske wird



(a) (b) (c) (d) (e)

**Abbildung 5.1.:** Unterschiedliche Anwendung von Alphamasken auf den Vordergrund im Vergleich. (a) Originale Aufnahme. (b) Extrahierte Maske (c) Vormultiplizieren mit schwarzem Hintergrund (übliche Vorgehensweise bei der Postproduktion) und anschließender Kombination mit weißem Hintergrund (d) Kombination mit weißem Hintergrund ohne Vormultiplizieren. (e) Maskierung mit binärer Maske.

entsprechend der Compositing-Gleichung mit einem (neuen) Hintergrund kombiniert:

$$I(x,y) = M(x,y)I_V(x,y) + (1 - M(x,y))I_H(x,y) \tag{5.1}$$

Es handelt sich um eine Addition der Pixel des Vordergrundbildes $I_V$ und des Hintergrundbildes $I_H$, gewichtet durch die auf 1 normierten Alphamaske $M$. Für die Nutzung innerhalb dieser Arbeit ist zu beachten, dass der ursprüngliche Vordergrund entsprechend dem Original erhalten bleibt. Eine sonst übliche Vormultiplikation mit der Alphamaske und einem schwarzen Hintergrund (*premultiplying*) ist zu vermeiden, da so die Farbwerte im Randbereich modifiziert werden, was zu Problemen bei der Stereoanalyse führen kann (vgl. Abb. 5.1c). Jedoch ist auch die direkte Nutzung einer abgestuften Alphamaske problematisch. So würde z. B. die Wichtung einer Disparitätskarte entsprechend Gleichung (5.1) zu verfälschten Tiefenwerten und somit zu Synthesefehlern führen. Daher wird sich in dieser Arbeit auf eine binären Maske beschränkt (vgl. Abb. 5.1e). Die dadurch entstehende Qualitätsminderung durch Aliasing im Randbereich der Person wird aufgrund der Genauigkeit und oben genannter Vorteile in Kauf genommen. In gewissem Maße können die Effekte durch die Weichzeichnung der synthetisierten Alphamaske reduziert werden.

Die im vorherigen Absatz genannte Vorgehensweise ist zur Generierung der Maske für die Erzeugung von Testdaten und deren Nutzung innerhalb dieser Arbeit ausreichend. Jedoch kann diese Vorgehensweise in einem Echtzeit-Kommunikationsszenario nicht genutzt werden. Daher wurden im Rahmen der Arbeit grundlegende Untersuchungen für alternative Methoden durchgeführt, deren Einsatz in einem Echtzeit-Szenario denkbar ist. Als gegebener Ausgangspunkt wurde definiert, dass ein Hintergrundmodell ohne den Vordergrund verfügbar ist. Dies ließe sich z. B. einfach realisieren, indem sich der Kommunikationsteilnehmer zu Beginn des Videochats kurz aus dem Kamerabild entfernt. Im einfachsten Fall ergibt sich dadurch die Möglichkeit einer Hintergrundsubtraktion. Dabei werden von den Intensitätswerten des aktuellen Bildes $I$ die des

(mittleren) Hintergrundbildes $I_H$ abgezogen[1]. In der binären Maske $M$ werden die Stellen, an denen die Differenz einen bestimmen Schwellwert $\tau$ überschreitet, als Vordergrund definiert, alle anderen als Hintergrund:

$$M(x,y) = \begin{cases} 1 & \text{wenn } \|I(x,y) - I_H(x,y)\| > \tau \\ 0 & \text{sonst} \end{cases} \tag{5.2}$$

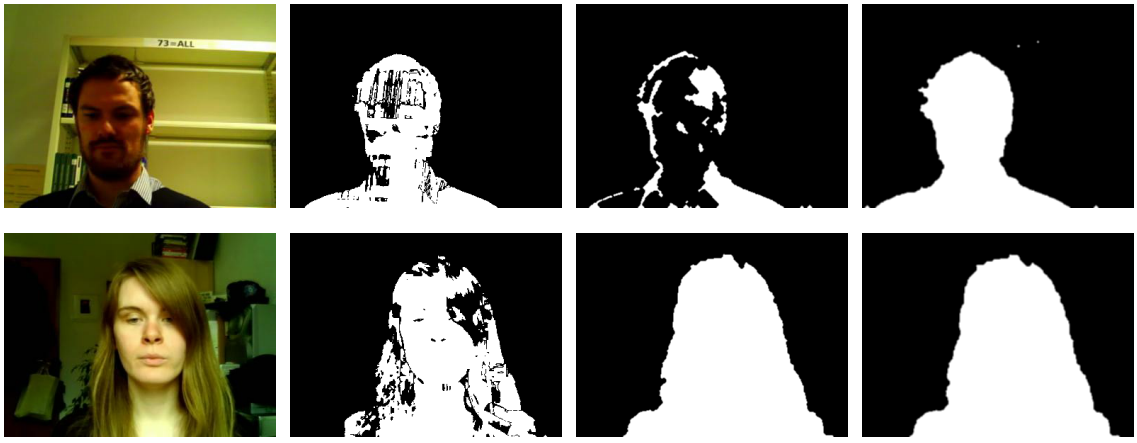
Durch Rauschen, graduelle Lichtveränderungen der Szene sowie ähnliche Farben im Vordergrund und Hintergrund ist diese Methode jedoch sehr unzuverlässig und erzeugt schlechte Masken. Daher wurden noch weitere Methoden untersucht. Ein gängiges Verfahren ist die Modellierung eines so genannten Pixelprozesses unter der Verwendung von Gaußschen Mischverteilungen (*Gaussian Mixture Models*-GMM oder *Mixture of Gaussians*-MoG) [Stauffer und Grimson, 1999]. Solch eine Mischverteilung ist die Summe verschiedener Normalverteilungen $\eta$, welche jede eine so genannte Komponente des Modells repräsentiert. Jede dieser Normalverteilungen ist die Wahrscheinlichkeitsdichtefunktion für den Zustand, dass ein zum Zeitpunkt $t$ beobachteter Pixelwert $I_t$[2] zur jeweiligen Komponente gehört, sprich einer bestimmten Intensität bzw. Farbe zugeordnet werden kann. Für jede Pixelposition wird nun ein solches Prozessmodell erstellt:

$$P(I_t) = \sum_{k=1}^{K} \omega_{k,t} \eta(I_t - \mu_{k,t}, \sigma_{i,t}) \tag{5.3}$$

Aus Gleichung (5.3) ist ersichtlich, dass die Parameter des Modells die Wichtung $\omega$ der Komponenten, der Mittelwert$\mu$ und die Standardabweichung $\sigma$ sind. Es sei an dieser Stelle darauf hingewiesen, dass diese Gleichung schon eine Vereinfachung darstellt, da bei Farbbildern an Stelle des verwendeten Skalars der Standardabweichung ursprünglich die Kovarianzmatrix $\Sigma$ der einzelnen Farbkanäle verwendet wird. Aufgabe des Verfahrens ist es nun, auf Basis der Beobachtung der Intensität/Farbe eines Pixels (a) eine Zuordnung des Pixels zu einer der Komponenten $k$ und anschließend zum Vorder- oder Hintergrund vorzunehmen und (b) die Parameter des Modells entsprechend dem aktuell beobachteten Pixelwert zu aktualisieren. Dabei existieren so viele unterschiedliche Zuordnungsmöglichkeiten, wie es Komponenten gibt. Üblicherweise werden $5$ bis $7$ Komponenten genutzt.

Die (approximierte) Lösung beschreiben Stauffer und Crimson in ihrem Paper [Stauffer und Grimson, 1999]. Unter Verwendung des Bayes-Theorems (vgl. auch Gleichung (2.57)) wird die bedingte Wahrscheinlichkeit (*likelihood*) formuliert, die angibt, dass ein Pixelwert der Komponente $k$ entspricht. Durch Maximierung dieser bedingten Wahrscheinlichkeit (vgl. Gleichung (2.58)) kann das entsprechende $k$ gefunden werden. Zur detaillierten mathematischen Formulierung für das konkrete Problem sei auf [Stauffer und Grimson, 1999; Power und Schoonees, 2002]

[1] ggf. pro Farbkanal
[2] Vereinfachte Schreibweise von $I_t(x,y)$



**(a)** Original **(b)** Differenz **(c)** GMM **(d)** GMM+Differenz

**Abbildung 5.2.:** Ergebnisse untersuchter automatischer Segmentierungsmethoden. Die einfache Differenzbildung (b) nach Gleichung (5.2) erfolgte mit einem hohen Schwellwert, so dass Fehlstellen im Hintergrund vermieden werden. (c) zeigt den zeitlichen Effekt des „Hintergrund Werdens“. Oben handelt es sich bereits um Bild 210 mit wenig bewegter Person. Das Bild unten ist Nummer 50 und die Bewegung in das Bild hält noch an. (d) Ein Kombination aus Differenzbildung und GMM vermeidet den Effekt des „Hintergrund Werdens“. Aus [Schmidt, 2010].

verwiesen. Ist nun die Komponente $k$ bestimmt, erfolgt eine Zuordnung zum Vordergrund oder Hintergrund, Diese geschieht durch die Sortierung der Komponenten entsprechend dem Wert $\omega_k/\sigma_k$. Ein hohes $\omega$ bedeutet ein häufigeres Auftreten, ein geringes $\sigma$ geringe Veränderlichkeit der Komponente, beides Merkmale für den Hintergrund. Ein definierter Schwellwert für diesen Term ordnet dann darunter liegende Komponenten dem Vordergrund, die restlichen dem Hintergrund zu. Mit dieser auch Rangordnungs-basiert genannten Zuordnung ist Schritt (a) des Algorithmus beendet.

Es folgt die Aktualisierung des Modells in Schritt (b). Hierfür wird in der Theorie der *Expectation-Maximisation* Algorithmus [Dempster u. a., 1977] genutzt. Durch die Maximierung der Likelihoodfunktion aller bisher beobachteten Werte kann unter Verwendung des erwarteten Wertes (bestimmt durch die aktuelle Beobachtung) der aktuelle Zustand des Modells bestimmt werden. Aufgrund der Komplexität der Berechnung werden die daraus entstehenden Gleichungen und Rechenschritte durch skalare Wichtungsfaktoren approximiert. Für jeden Parameter $\omega$, $\mu$, $\sigma$ entsteht so eine einfache Gleichung für dessen Aktualisierung. Interpretiert werden können diese als eine graduelle zeitliche Anpassung des Modells an die aktuelle Beobachtung. Diese ermöglicht auch völlig neuen Beobachtungen, wie z. B. einer in das Bild bewegten Hand, bei längerem (statischen) Vorhandensein Teil des Hintergrundmodells zu werden.

Ist der zuletzt genannte Umstand für viele Applikationen wünschenswert, so ist er für Videochatsituationen eher von Nachteil, da ein Kommunikationspartner sich durchaus wenig bewegen

kann und somit langsam zum „Hintergrund“ würde. Dem wurde in den Experimenten dieser Arbeit entgegengewirkt, indem zusätzlich zur Rangordnungs-basierten Klassifizierung von Vorder- und Hintergrund auch das Kriterium einer einfachen Differenzbildung zur Entscheidung beiträgt. Experimente mit verschiedenen Sequenzen, die mit den Logitech-Kameras aufgenommen wurden, zeigen, dass diese Vorgehensweise durchaus Potenzial hat. Auch existieren inzwischen effiziente, schnell echtzeitfähige Implementierungen des Ansatzes. Exemplarische Ergebnisse sind in Abb. 5.2 dargestellt. Das Verfahren wurde zur Verbesserung der Masken noch durch die Verwendung von MRFs erweitert (vgl. auch 2.2.6), worauf hier aber nicht weiter eingegangen werden soll. Eine detaillierte Beschreibung ist in der bearbeitenden Diplomarbeit von Julia Schmidt zu finden [Schmidt, 2010]. Abschließend ist festzuhalten, dass eine Segmentierung als automatisierter Vorverarbeitungsschritt unter den gegebenen Voraussetzungen möglich ist. Auch neuere Veröffentlichungen, wie in [Pei u. a., 2011] zeigen klar in diese Richtung. Aus zeitökonomischen Gründen wird innerhalb dieser Arbeit auf eine weitere Betrachtung verzichtet, da die effiziente manuelle Segmentierung für die Experimente der ideale Ausgangspunkt ist.

### 5.1.2. Rektifizierung

Die zu analysierenden Aufnahmen liegen nach dem vorherigen Vorverarbeitungsschritt als segmentierte Bildsequenzen in Form der RGB-Bildinformationen und einer zusätzlichen Maske vor. Um nun die Stereoanalyse durchzuführen, bedarf es eines weiteren Vorverarbeitungsschrittes der Daten. Wie in den Abschnitten 2.1.5 und 2.2 bereits erwähnt, setzten die meisten Stereoanalysealgorithmen Punktkorrespondenzen entlang identischer Bildzeilen des linken und rechten Bildes voraus. D. h. die Epipolarlinien sind waagerecht und liegen jeweils auf derselben Bildzeile wie der dazugehörige Punkt bzw. Pixel. Da dieser Zustand nur in perfekt parallelen Kameraanordnungen vorkommt, müssen bei realen, konvergenten Aufnahmen Maßnahmen getroffen werden, um diesen Zustand herzustellen. Diesem Zweck dient die Rektifizierung des linken und rechten Bildes. Allgemein bezeichnet Rektifizierung die lineare oder nichtlineare Transformation eines zweidimensionalen Bildes. Unter der Verwendung der Zusammenhänge der Stereogeometrie kann diese Transformation derart bestimmt werden, dass oben genannter Zustand herbeigeführt werden kann. Innerhalb dieser Arbeit wurden zwei Methoden untersucht und umgesetzt – eine lineare (projektive) und eine nichtlineare Transformation.

**Projektive Rektifizierung:** Das Prinzip der linearen Rektifizierung ist in Abb. 5.3 dargestellt. Die Abbildungen $I$ und $I'$ in den originalen Bildebenen werden jeweils durch $3 \times 3$ Homographien $\mathtt{H}$ und $\mathtt{H}'$ auf die rektifizierte Bildebene transformiert, was faktisch ein Re-Sampling der Bilder darstellt. Es ergibt sich die Fragestellung, wie diese Homographien bestimmt werden können und zwar so, dass die Verzerrung der beiden Bilder dabei so gering wie möglich ausfällt. In dieser Arbeit wurde für die Rektifizierung zunächst der unkalibrierte Fall angenommen, so

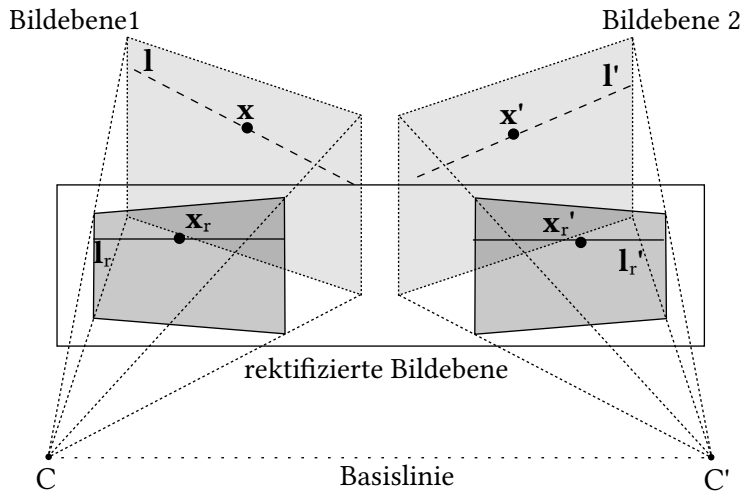



**Abbildung 5.3.:** Prinzip der projektiven Rektifizierung. Nach [Kreibich, 2005].

dass die Bestimmung allein auf Basis robuster Punktkorrespondenzen erfolgt. Dabei wird der Algorithmus von Hartley [Hartley, 1999] verwendet. Zunächst wird anhand von hinreichend vielen Punktkorrespondenzen die Fundamentalmatrix $\mathtt{F}$ zwischen der linken und der rechten Abbildung bestimmt. Da üblicherweise mehr als die notwendigen 7 Punktkorrespondenzen vorliegen, erfolgt eine robuste Bestimmung mittels RANSAC-Algorithmus (vgl. Abschnitt 2.1.5, Gleichung (2.30)). Zum weiteren Vorgehen sind nun die Epipole notwendig. Aus Gleichung (2.29) und dem Umstand, dass die Epipole alle Epipolarlinien schneiden, lässt sich folgende Bedingung ableiten:

$$\mathtt{F}\mathbf{e} = 0 \text{ und } \mathtt{F}^\top \mathbf{e}' = 0 \tag{5.4}$$

Somit können durch Lösen des rechtsseitigen Nullraumes von $\mathtt{F}$ die Epipole aus der Fundamentalmatrix berechnet werden. Zur Bestimmung der rektifizierenden Homographiematrizen $\mathtt{H}$ und $\mathtt{H}'$ sind weitere Schritte erforderlich, die hier kurz dargelegt werden. Zunächst wird $\mathtt{H}'$ bestimmt. Sie setzt sich aus drei Bildtransformationen zusammen. Zuerst erfolgt die Transformation des Bildkoordinatensystems in den Bildmittelpunkt:

$$\mathtt{T} = \begin{bmatrix} 1 & 0 & -u \\ 0 & 1 & -v \\ 0 & 0 & 1 \end{bmatrix} \tag{5.5}$$

Diese Transformation wird auch auf den Epipol $\mathbf{e}'$ angewendet, wodurch sich dessen transformierte Koordinaten $\mathbf{e}'^{+}$ ergeben. Anschließend erfolgt eine Rotation mittels einer Matrix $\mathtt{R}$ derart, dass der Epipol $\mathbf{e}'^{+} = (e_x'^{+}, e_y'^{+}, 1)^\top$ auf die x-Achse und somit auf Position $\mathbf{e}'^{*} = (e_x'^{*}, 0, 1)^\top$

gedreht wird:

$$\mathtt{R} = \begin{bmatrix} \cos r & \sin r & 0 \\ -\sin r & \cos r & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ mit } r = \arctan \frac{e'^{+}_{y}}{e'^{+}_{x}} \tag{5.6}$$

Die bisherigen Transformationen sind euklidische Transformationen, so dass die Bildverzerrungen gering ausfallen. Die Projektion des so transformierten Epipols in das (anschaulich gesprochene) Unendliche – die *Line at Infinity* – erfolgt abschließend mittels der Transformation

$$\mathtt{G} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -1/e'^{*}_{x} & 0 & 1 \end{bmatrix} \tag{5.7}$$

Die Homographie ergibt sich somit zu

$$\mathtt{H}' = \mathtt{GRT} \tag{5.8}$$

Durch diese Transformation liegen alle Epipolarlinien waagerecht im Bild $I'$. Nun gilt es, die Homographie $\mathtt{H}$ für das andere Bild zu bestimmen. Eine direkte Lösung anhand der auf den Epipolarlinien liegenden Korrespondenzen ist nicht möglich, da diese als Schätzungen fehlerbehaftet sind. Daher wird zur Bestimmung der mittlere quadratische Abstand der transformierten Bildpunktpositionen korrespondierender Punkte $\mathbf{x}_{i,r} \leftrightarrow \mathbf{x}'_{i,r}$ minimiert (Subindex $i$ steht für den jeweiligen Korrespondenzpunkt, $r$ für die Rektifizierung/Transformation durch $\mathtt{H}$ und $\mathtt{H}'$):

$$\sum_{i} d(\mathtt{H}\mathbf{x}_{i,r}, \mathtt{H}'\mathbf{x}'_{i,r})^2 \tag{5.9}$$

Hartley beweist in [Hartley, 1999], dass, wenn eine faktorisierte Fundamentalmatrix der Form

$$\mathtt{F} = \mathbf{e}' \times \mathtt{M} = [\mathbf{e}']_{\times}\mathtt{M} \tag{5.10}$$

vorliegt ($\mathtt{M}$ stellt hierbei eine beliebige nicht-singuläre $3 \times 3$-Matrix dar), eine projektive Transformation $\mathtt{H}$ nur dann zu $\mathtt{H}'$ passt, wenn

$$\mathtt{H} = (\mathtt{I} + \mathtt{H}'\mathbf{e}'\mathbf{a}^T)\mathtt{H}'\mathtt{M} \tag{5.11}$$

gilt [Kreibich, 2005]. $\mathbf{a}^T$ ist hierbei ein beliebiger Vektor. Wie oben beschrieben, transformiert $\mathtt{H}'$ den Epipol $\mathbf{e}'$ ins Unendliche z. B. auf $(1, 0, 0)^\top$. Somit kann der Term $\mathtt{I} + \mathtt{H}'\mathbf{e}'\mathbf{a}^T$ als $\mathtt{I} + (1, 0, 0)^T\mathbf{a}^T$ geschrieben werden. Daraus lässt sich die affine Transformation in Gleichung (5.12) ableiten.

$$\mathtt{H}_A = \begin{bmatrix} a & b & c \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5.12}$$

Rektifizierte Korrespondenzen $\mathbf{x}_{i,r} = (u_{i,r}, v_{i,r}, 1)^\top$ und $\mathbf{x'}_{i,r} = (u'_{i,r}, v'_{i,r}, 1)^\top$ ergeben sich durch eine lineare Transformation nach Gleichung (5.13).

$$\mathbf{x}_{i,r} = \mathtt{H'M}\mathbf{x}_i \qquad \mathbf{x'}_{i,r} = \mathtt{H'}\mathbf{x'}_i \tag{5.13}$$

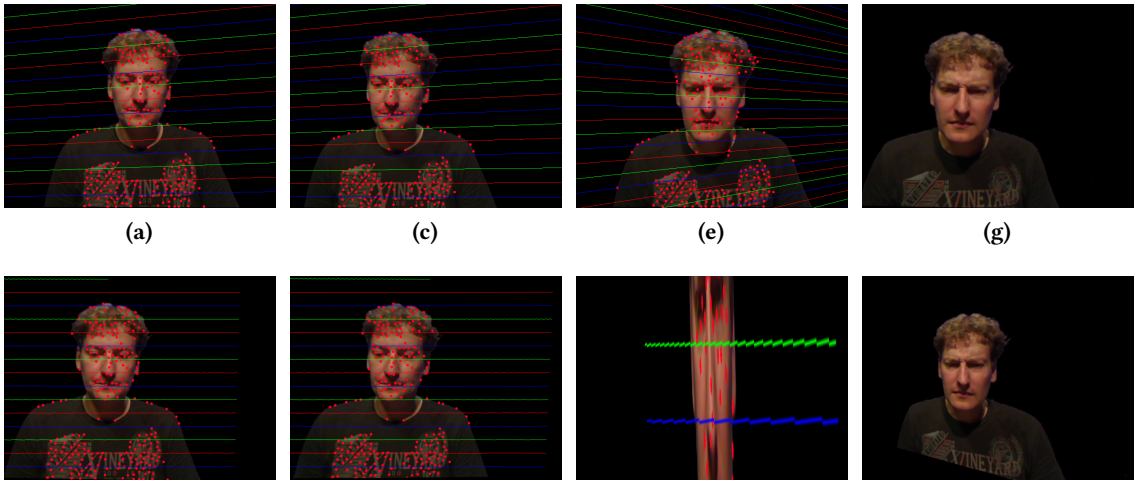
Das Optimierungsproblem aus Gleichung (5.9) wird nun so umformuliert, dass am Ende nur noch das lineare Minimierungsproblem aus Gleichung (5.14) verbleibt. Dieses lässt unter Verwendung der transformierten Korrespondenzen durch eine lineares Gleichungssystems z. B. mittels Singulärwertzerlegung lösen.

$$\sum_i d(au_{i,r} + bv_{i,r} + c - u'_{i,r})^2 \tag{5.14}$$

Nach Ermittlung der Koeffizienten $a, b, c$ kann die gesuchte Homographie schließlich durch die Beziehung $\mathtt{H} = \mathtt{H}_A\mathtt{H'M}$ berechnet werden. Damit liegen zwei Homographien vor, mit deren Hilfe die Referenzansichten in rektifizierte Ansichten überführt werden können. Ursprüngliche und tiefer gehende Erläuterungen sowie die Beweise für die obigen Gleichungen sind in [Kreibich, 2005] und [Hartley, 1999] zu finden.

Da die durch die Transformation berechneten neuen Punktposition nicht immer auf dem ganzzahligen Pixelraster liegen, müssen diese Werte interpoliert werden. Hartley empfiehlt in [Hartley, 1999] eine lineare Interpolation, die auch in dieser Arbeit verwendet wird.

Die Rektifizierung mittels linearer Transformation scheint eine effektive und zugleich einfache Lösung zu sein. Die Homographien könne direkt aus Bildmerkmalen gewonnen werden (Selbstkalibrierung). Die Nutzung einer einzelnen Transformationsmatrix für das gesamte Bild ist effizient. Beispielsweise kann diese Vorwärtstransformation ohne Probleme invertiert werden. Als Rückwärtstransformation ist sie dann ideal geeignet, um als paralleler Prozess z. B. auf der GPU implementiert zu werden. Eine entsprechende Umsetzung wurde auch in dieser Arbeit realisiert. Grundsätzlich ist die Bestimmung der Fundamentalmatrix durch die Verwendung von RANSAC robust. Genauere Untersuchungen zeigen jedoch, dass sie sich von Bild zu Bild einer Sequenz ändern kann. Die Implikationen sind folgende: Die Epipolarlinien können für jedes Bildpaar eine unterschiedliche Steigung aufweisen. Da der oben beschriebene Algorithmus sich an diesen orientiert, werden in den rektifizierten Bildern korrespondierende Abbildungen auch auf denselben Bildzeilen liegen. Jedoch ist der Grad der Verzerrung für jedes Bild unterschiedlich stark. Durch die Anwendung der Homographie kann das Bild so stark verzerrt werden, dass kaum noch Bildinformationen enthalten sind, womit die anschließend gewonnene Disparitätskarte unbrauchbar wird. Nun könnte die Fundamentalmatrix aus einem Bildpaar bestimmt werden und anschließend auf alle weiteren angewendete werden. Bei dieser Methode bleibt jedoch die Frage nach der optimalen Fundamentalmatrix für die gesamte Sequenz offen. Abb. 5.4 veranschaulicht den Sachverhalt an einem Beispiel. Ein weiteres Problem ist, dass nichtlineare Linsenverzerrun-



(a) (c) (e) (g)

(b) (d) (f) (h)

**Abbildung 5.4.:** Beispiele für projektive Rektifizierung. (a) Linkes Bild 101 einer Sequenz mit einzelnen, eingezeichneten Epipolarlinien und Markierung der verwendeten Merkmale zur F-Matrix-Bestimmung. (b) Rektifizierung von 101 links mit F-Matrix aus 101. (c) Bild 101 rechts. (d) Rektifizierung von Bild 101 rechts. (e) Bild 120 links. (f) Misslungene Rektifizierung von Bild 120 links. (g) Rektifizierung von Bild 120 links mit F-Matrix aus 101. (h) Rektifizierung des Linken Bildes 120 mit F-Matrix aus Bild 136.

gen von dieser Methode nur unzureichend ausgeglichen werden können und somit im manchen Bereichen die korrespondierenden Abbildungen nicht mehr auf denselben Epipolarlinien liegen. Dies führt zu der Entscheidung, in dieser Arbeit eine andere, bessere Art der Rektifizierung zu nutzen, die im Folgenden beschrieben wird.

**Nichlineare Rektifizierung:** Diese hier untersuchte Art der Rektifizierung unterscheidet sich in zwei wesentlichen Merkmalen von der projektiven Rektifizierung. Sie nutzt vorher ermittelte Daten der externen Kalibrierung und sie verwendet eine nichtlineare Transformationsvorschrift unter Verwendung eines Linsenverzerrungsmodells. Der Vorgang ist als Rückwärtstransformation formuliert. Das nun vorgestellte Verfahren basiert auf der Implementierung in der Matlab Camera Calibration Toolbox [Bouguet, 2010].

Ein dreidimensionaler Punkt $\mathbf{X} = (X, Y, Z)$ im Koordinatensystem der linken Kamera wird durch die Rotation $\mathtt{R}$ und die Translation $\mathbf{t}$ in das Koordinatensystem der rechten Kamera überführt. $\mathtt{R}$ und $\mathbf{t}$ sind das Ergebnis der Kamerakalibrierung (vgl. 4.5). Für die Rektifizierung wird nun eine euklidische Transformation bestimmt, die beide Kameras so transformiert, dass sie eine parallele Kameraanordnung bilden. Dabei wird beachtet, die Kameras so gering wie möglich zu transformieren um die resultierenden Bildverzerrungen zu minimieren. Zunächst werden beide Koordinatensysteme so rotiert, dass deren z-Achsen dieselbe Orientierung aufweisen (vgl. Abb. 5.5b). Die Rotation wird dabei auf beide Kameras gleichmäßig „verteilt“, indem die notwendigen

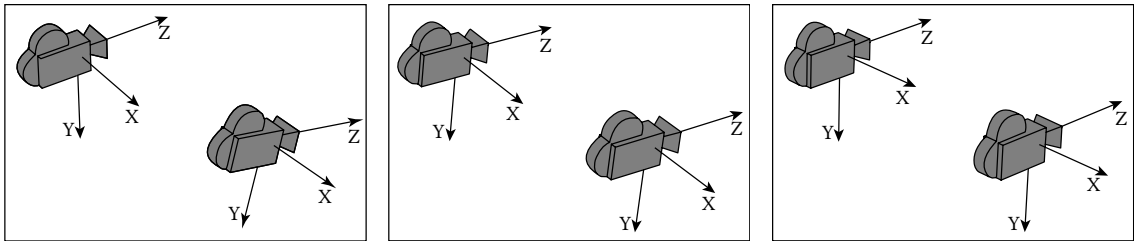



**Abbildung 5.5.:** Externe Transformation bei nichtlinearer Rektifizierung

Rotationswinkel der Rotation $\mathtt{R}$ von der linken in die rechte Kamera halbiert werden[3]. Die daraus resultierenden Rotationsmatrizen seien als $\mathtt{R}_{0,\mathtt{l}}$ für die linke und $\mathtt{R}_{0,\mathtt{r}}$ für die rechte Kamera bezeichnet. Anschließend wird der durch die Drehung entstehende Translationsvektor $\mathbf{t}^* = \mathtt{R}_{0,\mathtt{r}}\mathbf{t}$ auf die X-Achse des linken Kamerakoordinatensystems gedreht. Die dafür notwendige Rotation $\mathtt{R}_\mathtt{X}$ wird in beiden Koordinatensystem angewendet, so dass eine parallele Ausrichtung entsteht (vgl. Abb. 5.5c). Die Gesamtrotationen zur Überführung in eine parallele Kameraanordnung ergeben sich somit zu

$$\mathtt{R}_\mathtt{l} = \mathtt{R}_\mathtt{X}\mathtt{R}_{0,\mathtt{l}} \quad \text{und} \quad \mathtt{R}_\mathtt{r} = \mathtt{R}_\mathtt{X}\mathtt{R}_{0,\mathtt{l}} \tag{5.15}$$

Die internen Parameter des parallelen Kamerapaares werden entsprechend idealen Lochkameras gesetzt. Horizontale und vertikale Brennweiten und der Kamerahauptpunkt werden so bestimmt, dass der maximale Bildinhalt erhalten bleibt. Hieraus ergeben sich die internen Kameramatrizen der idealen Kameras $\mathtt{K}_{\mathtt{n},\mathtt{l}}$ und $\mathtt{K}_{\mathtt{n},\mathtt{r}}$.

Unter der Kenntnis der jeweiligen Transformationen in eine parallele Anordnung lässt sich nun die Rektifizierung durchführen, was am Beispiel der linken Kamera kurz erläutert wird. Zunächst werden die Punkt- bzw. Pixelpositionen der *idealen, rektifizierten* Kamera in den Raum projiziert, was einer Umkehrung der Kameraprojektion in Gleichung (2.19) entspricht, wobei die Projektionsmatrix nur aus der internen Kameramatrix besteht:

$$\mathbf{X} = \mathtt{K}_{\mathtt{n},\mathtt{l}}^{-\mathtt{1}} \quad (x, y, 1)^\top \tag{5.16}$$

Der dreidimensionale Punkt $\mathbf{X}$ in homogenen Koordinaten des (rektifizierten) Kamerakoordinatensystems ist bis auf einen Skalierungsfaktor definiert. Da dieser durch die unbekannte Tiefe des Punktes determiniert ist, kann er nicht bestimmt werden. Für die Rektifizierung ist die Tiefe jedoch auch nicht notwendig. Die Anwendung der inversen Transformation von der ursprünglichen in die rektifizierende Anordnung transformiert die Raumpunkte (besser vorstellbar als Strahlen) entsprechend:

$$\mathbf{X}' = \mathtt{R}_\mathtt{l}^{-1}\mathbf{X} \tag{5.17}$$

[3] Die Bestimmung dieser Winkel aus $\mathtt{R}$ kann durch die Rodrigues-Formel erfolgen

Anschließend werden das nichtlineare Verzerrungsmodell sowie die interne Kameratransformation der realen Kameras – beide bestimmt durch die Kamerakalibrierung – entsprechend Gleichungen (2.21) bis (2.24) angewendet. Für jeden ursprünglichen Pixel des rektifizierten Bildes wird so der Ausgangspunkt im originalen Kamerabild bestimmt. Da dieser Punkt durch die Projektion auch zwischen verschiedenen Pixelpositionen liegen kann, wird aus benachbarten Punkten eine bilineare Interpolation durchgeführt. Aufgrund des nichtlinearen Charakters dieser Transformationen müssen für jeden rektifizierten Bildpixel die Ursprungspixelposition sowie vier Wichtungsfaktoren für die Interpolation mit den benachbarten Pixeln gespeichert werden. Pixel, die außerhalb der Stereo-Bilder liegen, werden verworfen. Die Datenstruktur muss so gespeichert werden, dass sie sich für eine schnelle Berechnung rektifizierter Videos eignet. Da auf jedes Bild des Videos dieselbe nichtlineare Abbildung angewendet wird, ist auch dieser Vorgang zügig durchführbar, jedoch rechenintensiver als die projektive Rektifizierung.

Diese Form der nichtlinearen Rektifizierung unter Verwendung des Verzerrungsmodells ist wesentlich akkurater als eine lineare Rektifizierung. Insbesondere bei der Verwendung von Webcams mit vergleichsweise schlechten Optiken ist sie das Mittel der Wahl innerhalb dieser Arbeit. Jedoch erfordert sie eine genaue Kamerakalibrierung, vorzugsweise unter der Verwendung eines definierten Kalibrierungskörpers wie dem Schachbrett. Die Verwendung einer Selbstkalibrierung anhand von korrespondierenden robusten Merkmalen würde ohne weitere stabilisierende Maßnahmen auch bei der nichtlinearen Rektifizierung zu Problemen führen.

Die nichtlineare Derektifizierung erfolgt nach demselben Prinzip. Ausgehend vom Zielbild (verzerrt, derektifizierte Kamera) wird die Pixelposition im Quellbild (ideal, rektifizierte Kamera) errechnet. Problematisch dabei erweist sich, dass nun die originale Kamera Ausgangspunkt der Berechnung ist. Es müssen demnach die durch das nichtlineare Verzerrungsmodell der Kamera abgebildeten Punkte als Strahlen in den Raum projiziert werden (vgl. Gl. (5.16)). Da hierfür keine algebraische Lösung existiert, kommt eine numerische Lösung zum Einsatz, die in der Camera Calibration Toolbox umgesetzt ist [Bouguet, 2010]. Anschließend wird wie bei der Rektifizierung vorgegangen. Die rückprojizierten Strahlen werden transformiert (Gl. (5.17)) und in die ideale rektifizierte Kamera projiziert um die Quellpixel zu bestimmen.

## 5.2. Stereoanalyseverfahren

Die Algorithmen für die Synthese virtueller Ansichten zur Blickkorrektur benötigen Korrespondenzen zwischen den Abbildungen dreidimensionaler Punkte. Die Abbildung dieser Punkte erfolgt auf einen Sensor mit entsprechendem Pixelraster (vgl. Abschnitt 2.2.1). Aufgrund dieser Eingangsdaten ist, wie im Grundlagenkapitel im Abschnitt 2.2 bereits ausgeführt, die Stereoanalyse als ein Problem des Findens korrespondierender Pixel formuliert - der Korrespondenzsuche (*stereo matching*). Die Formulierung unterliegt der Annahme, dass diese örtliche Abtastung kein

Aliasing verursacht und der örtlichen bzw. der Tiefenauflösung des menschlichen Auges nahe kommt. Ein Pixel der Abbildung wird damit implizit der Abbildung eines Raumpunktes gleich gesetzt. Ob diese Annahme für den Anwendungsfall ausreichend ist, wird im folgenden Abschnitt 5.2.1 kurz untersucht.

Das Ziel des hier entwickelten Verfahrens ist ein ebenso „dichtes“ Bild des Objektes wie in den originalen Ansichten unter ähnlichen Betrachtungsbedingungen. Daher müssen möglichst alle verfügbaren Bildinformationen als Eingangsdaten genutzt werden. Dies impliziert, dass nach Möglichkeit für jeden Bildpunkt auch eine Korrespondenz gefunden werden muss. Diese Korrespondenzen werden durch eine *dichte Disparitätskarte* repräsentiert. Die in dieser Arbeit entwickelten und untersuchten Verfahren zur Erzeugung solcher dichten Disparitätskarten werden in den folgenden Abschnitten beschrieben. Ausgehend von einer grundlegenden Auswahl des verwendeten Prinzips werden ggf. Weiterentwicklungen und Maßnahmen zur Verbesserung vorgestellt.

### 5.2.1. Disparitätsrepräsentation

Zur Auswahl einer geeigneten Datenrepräsentation der Disparitätskarten soll diese hier kurz thematisiert werden. Wie eingangs beschrieben, wird aufgrund der ganzzahligen Eingangsdaten der Stereoanalyse die Disparität zumeist ebenso ganzzahlig repräsentiert. Üblicherweise finden dabei $8$ `bit` Verwendung, die $256$ diskrete Werte ermöglichen. Ob diese Repräsentation ausreicht, kann durch die folgende Analyse des Abbildungsvorganges ermittelt werden.

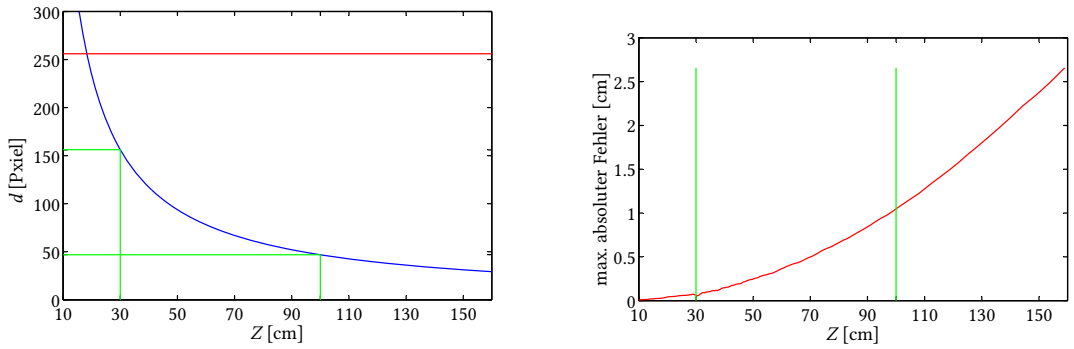
Bei der Abbildung eines dreidimensionalen Bildpunktes auf die Sensoren zweier achsenparalleler Kameras berechnet sich die Disparität durch Umformung von Gleichung (2.27) aus

$$d = bf \cdot \frac{1}{Z} \tag{5.18}$$

Die Einheit der Disparität entspricht in dieser Gleichung noch der Einheit von Basisabstand $B$ und Brennweite $f$ (z. B. `mm`) und ist somit unabhängig vom verwendeten Sensor. Da in der Stereoanalyse jedoch nur diskrete Pixel als Eingangsdaten verfügbar sind, ist eine Angabe im Maß des jeweils verwendeten Sensors sinnvoll. Durch den Term $\delta_x$ wird die Disparität in einen Wert der Einheit `Pixel` umgerechnet. $\delta_x$ entspricht der Breite eines Sensorelementes und ist von der Größe des Sensors sowie der Anzahl aktiver Sensorelemente abhängig.

$$d = \frac{bf}{\delta_x} \cdot \frac{1}{Z} \tag{5.19}$$

Die Disparität in Gleichung (5.19) hat noch immer einen kontinuierlichen Wertebereich. Die Eignung einer bestimmten diskreten Repräsentation hängt nun letztlich von folgenden Faktoren ab: der Nutzung ganzzahliger oder reeller Disparitäten, der verwendeten Anzahl kodierender




**(a)** **(b)**

**Abbildung 5.6.:** (a) Disparitätsbereich für gewähltes Anwendungsszenario. Es ist erkennbar, dass der Wertbereich für den in Frage kommenden Tiefenbereich ausreicht. (b) Darstellung der maximalen absoluten Fehler durch Verwendung ganzzahliger Disparitäten. Die Parameter der Aufnahme für beide Abb.: $B = 9$ cm, $f = 3.7$ mm, Sensorbreite $4.546$ mm womit sich bei einer Bildbreite von $640$ Pixeln $\delta_x = 7.1031$ $\mu$m ergibt.

Bits, der Größe eines Sensorelementes, sowie vom abzubildenden Tiefenbereich der Szene. Die Verwendung ganzzahliger Disparitäten kommt einer Rundung gleich und erzeugt einen Tiefenfehler im Vergleich zum realen Wert $d$ in Gleichung (5.19). Dieser nimmt durch die Nichtlinearität von Gleichung (5.19) mit steigender Tiefe zu. Ebenso muss der durch die gewählte Repräsentation verfügbare Wertebereich den sich aus der Tiefe ergebenen Wertebereich von $d$ fassen können.

Aus diesen Feststellungen ergibt sich die Frage, ob eine ganzzahlige $8$ bit-Repräsentation im konkreten Anwendungsfall der Videokommunikation ausreicht. Die minimale und maximale Disparität für einen bestimmten Tiefenbereich $\{Z_{min} \dots Z_{max}\}$ berechnen sich aus:

$$d_{min} = \frac{bf}{\delta_x} \cdot \frac{1}{Z_{max}} \quad \text{und} \quad d_{max} = \frac{bf}{\delta_x} \cdot \frac{1}{Z_{min}} \tag{5.20}$$

Unter Verwendung der in Kapitel 4 festgelegten Kamera- und Aufnahmeparameter, eines Basisabstandes von $b = 9$ cm und der Annahme, die aufgenommene Person bewege sich innerhalb eines Tiefenbereichs von $30 \leq Z \leq 120$ cm ergibt sich $d_{min} = 47$, $d_{max} = 156$ und der maximale absolute Tiefenfehler bei $Z = 100$ cm zu $\approx 1$ cm. Abb. 5.6 veranschaulicht den Sachverhalt. Der $8$ bit-Wertebereich reicht demnach für ein direkte Abbildung der Disparität aus und lässt noch Spielraum, sollte eine Person näher oder weiter entfernt sein. Kritischer scheint der Tiefenfehler zu sein. Dieser ließe sich nur durch Repräsentation der Disparitäten als diskrete reelle Zahl beheben. Da jedoch viele Stereoanalysealgorithmen prinzipbedingt ganzzahlige Disparitäten bestimmen, ist diese Repräsentationsform seltener. Auch sind viele Kostenmaße grundsätzlich auf eine ganzzahlige Analyse ausgelegt. Diese Problem wurde z. B. in [Birchfield und Tomasi, 1998] durch ein Kostenmaß auf Subpixel-Ebene adressiert. Dies führt zwar zur Verbesserung der

Stereoanalyse, jedoch werden die Disparitäten auch dort nur ganzzahlig berechnet.

Anhand dieser kurzen Betrachtung wird trotz des Tiefenfehlers entschieden, ebenfalls eine ganzzahlige $8$ `bit` Repräsentation zu verwenden. Die Eingangsdaten, die Eigenschaft gängiger Kostenmaße auf ganzzahliger Pixelbasis zu arbeiten, eingeschränkte Repräsentationsformate sowie aufwändigere Verarbeitung sprechen gegen eine Repräsentation mittels reeller Zahlen. Der Wertebereich reicht aus und es erfolgt eine direkte Abbildung. Es sind keine Skalierung oder ein Offset notwendig[4]. Ein Blick auf den Fehler von ca. $0.25$ `cm` bei dem weitaus häufigeren Abstand der Person von $50$ `cm` festigt schlussendlich diese Entscheidung. Der Basisabstand wird für alle Testaufnahmen auf $\approx 9$ `cm` eingestellt.

### 5.2.2. Methodenauswahl

In den folgenden Abschnitten wird die Entscheidung für das zu verwendende Prinzip für die Stereoanalyse vorgestellt. Ausgewählte Ansätze und ggf. deren Modifikation werden beschrieben. Bildbasierte Stereoanalyse ist in den letzten 15 Jahren zu einem sehr aktiven und wichtigen Forschungszweig geworden. Dementsprechend existiert eine sehr große Anzahl von Verfahren, um aus einem Bild- oder Videopaar einer Szene dreidimensionale Informationen zu extrahieren. Zur Systematisierung und auch zur Evaluation der Algorithmen untereinander stellten Scharstein und Szeliski zu Beginn des Jahrtausends ihre Taxonomie für derartige Algorithmen vor [Scharstein und Szeliski, 2002]. Parallel dazu machten sie Stereobilder samt Referenzdisparitätskarte (*Ground Truth*) sowie einen Softwarebaukasten frei verfügbar. Eine Evaluationsmethodik sowie eine offene Datenbank ermöglichen seither jedem Forscher, sich am Stand der Technik zu messen [Scharstein und Szeliski, 2012].

Bietet dies einerseits einen sehr guten Überblick, welche Algorithmen derzeit die besten Ergebnisse mit dieser Evaluationsmethode erreichen, so bleiben Fragen hinsichtlich der Verwendung dieser Algorithmen im Kontext bestimmter Anwendungen unbeantwortet. Beispielsweise wurden viele Algorithmen zur Verwendung bei der Basislinieninterpolation entwickelt und sind dazu auch sehr gut geeignet. Jedoch sind sie oft auf diesen speziellen Anwendungsfall optimiert. Die Fehlermaße der Evaluationsmethodik setzen sich aus den Disparitätsfehlern (*bad pixels*) in verdeckten, halb verdeckten und nicht verdeckten Regionen für verschiedene Bildpaare zusammen. Sie lassen zunächst nur Aussagen über die Qualität des Algorithmus hinsichtlich der Referenzdisparitätskarten zu. Als Quintessenz bleibt an dieser Stelle, dass die Datenbank eine sehr gute Grundlage für eine erste Auswahl bietet, jedoch noch wesentlich mehr Parameter im konkreten Anwendungsfall bedacht werden müssen. Die technische Evaluation aller Verfahren für den Anwendungsfall der Videokommunikation ist unmöglich. Die Datenbank enthält bereits über

[4] Dies gilt für alle Berechnungen. Für die Visualisierung innerhalb dieser Arbeit kommt mitunter eine Skalierung und Falschfärbung zum Einsatz

100 verschiedene Vorschläge mit entsprechenden Referenzen, jedoch meist ohne die konkrete Implementierung. Viele Methoden teilen jedoch die grundlegenden Prinzipien.

### 5.2.3. Globale Korrespondenzsuche

Zum Zeitpunkt der Analyse innerhalb dieser Arbeit waren unter den am besten platzierten Verfahren globale Optimierungsansätze (vgl. Abschnitt 2.2.6) zu finden. Insbesondere Verfahren, die eine vorherige farbbasierte Segmentierung durchführen, zählen dazu. Ein offensichtlicher Vorteil der Verwendung von Segmenten anstelle von Pixeln ist die Reduzierung des Lösungsraumes der Optimierung (vgl. Abschnitt 2.2.6). Zwei grundlegende Annahmen liegen den Verfahren zugrunde. Die einfachere nimmt an, dass jedem Segment eine Disparität zugehörig ist. Eine Weiterentwicklung dieses Ansatzes ist, dass sich die Disparitäten innerhalb eines Segmentes nur gering ändern und diese sich im Disparitätsraum jeweils durch nicht überlappende Ebenen repräsentieren lassen. Die allgemeine Normalenform dieser Ebenengleichung im dreidimensionalen Disparitätsraum $(x, y, d)$ für jeden Pixel $(x, y)$ ist $d = \pi_1 x + \pi_2 y + \pi_3$. Ziel der Algorithmen ist es nun, die Parameter $\pi_1$, $\pi_2$ und $\pi_3$ dieser Ebenengleichungen zu bestimmen, wobei Nachbarschaftsbeziehungen über eine Glattheitsbedingung formuliert werden. Am Beispiel des in der Middlebury-Datenbank mehrere Jahre führenden Algorithmus von Klaus [Klaus u. a., 2006] soll das Prinzip stellvertretend erläutert werden. Der Ablauf des Algorithmus ist in Abb. 5.7 dargestellt.

Die Segmentierung der Eingangsbilder in Flächen gleicher Farbe erfolgt mittels des bewährten Mean-Shift-Algorithmus [Comaniciu und Meer, 2002]. Um eine erste Zuweisung von Disparitäten zu jedem Segment vorzunehmen, werden robuste Disparitäten durch eine lokale Korrespondenzsuche bestimmt. Dies erfolgt unter Verwendung eines Kostenmaßes basierend auf der gewichteten Summe absoluter Differenzen (SAD) und eines gradientenbasierten Maßes zwischen linkem und rechtem Bild. Die Wichtung der Maße wird so gewählt, dass sich ein Maximum an konsistenten Korrespondenzen entsprechend einer Konsistenzprüfung (*Cross-Check*) zwischen den Disparitäten des linken und des rechten Bildes ergibt. Die Autoren bezeichnen dieses Maß als selbst-adaptiv. Anhand dieser Disparitäten wird nun für jedes Segment eine repräsentative Ebene im Disparitätsraum bestimmt. Grundsätzlich ließe sich dies durch Lösen des überbestimmten Gleichungssystems aus den Ebenengleichungen unter Verwendung der lokal bestimmten Disparitäten und deren $(x, y)$-Position realisieren. Die Autoren wollen diesen Vorgang jedoch robuster machen. Sie bestimmen horizontale und vertikale Neigung $\pi_1$ und $\pi_2$ der Ebene separat anhand der sortierten und mittels Gauß-Filter geglätteten Gradienten der Disparitäten entlang der Zeilen bzw. Spalten des Segments. Anschließend wird Parameter $\pi_3$ über die Berechnung der zentralen Disparität mittels $\pi_1$ und $\pi_2$ und der robusten Disparitäten bestimmt. Alle so bestimmten Dispariätsebenen werden nun als Ebenen der gesamten Szene gespeichert. Es erfolgt eine Verfeinerung dieser Menge von Disparitätsebenen, indem eine Überprüfung durch

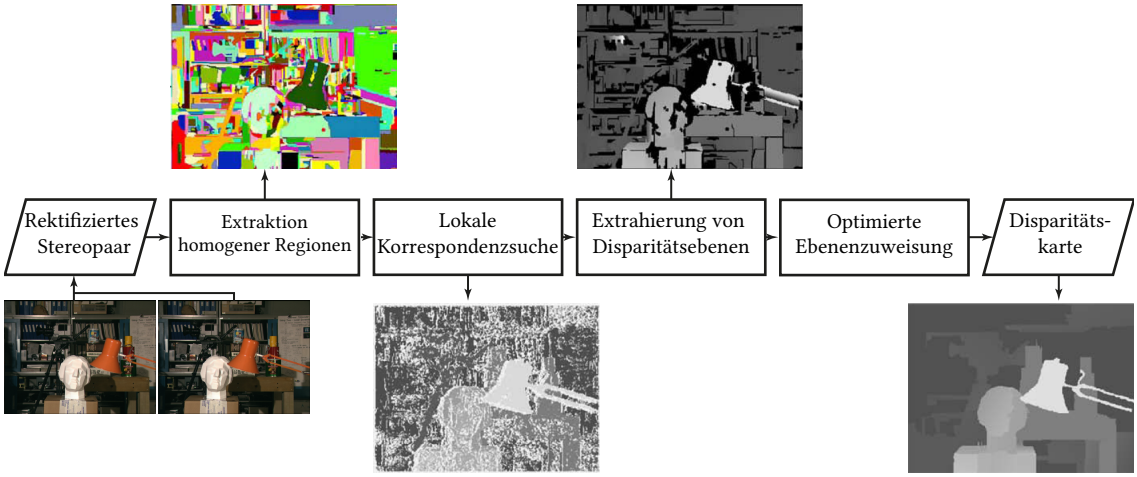



**Abbildung 5.7.:** Prinzip der segmentbasierten, globalen Stereo-Korrespondenzsuche. Nach [Klaus u. a., 2006].

ein einfaches Kostenmaß für jede Segment-Ebenenzuweisung stattfindet. Die Ebene mit den geringsten Kosten wird letztlich dem Segment zugewiesen und benachbarte Segmente mit identischer Ebenenzuweisung gruppiert. Für die gruppierten Segmente wird die Ebenenbestimmung noch einmal durchgeführt. Ziel des Schrittes ist weniger die exakte Ebenenzuweisung als eine kleinere und exaktere Menge von möglichen Disparitätsebenen der Szene. Im letzten Schritt wird nun die endgültige Zuweisung durch eine globale Optimierung im Sinne von Abschnitt 2.2.6 durchgeführt. Der Datenterm setzt sich aus den Zuweisungskosten (*matching costs*) einer Ebenen zum Segment sowie einem Glattheitsterm zusammen (vgl. Gleichung (2.65)). In diesem Fall wird die gemeinsame Kantenlängen sowie die mittlere Farbähnlichkeit benachbarter Segmente genutzt. Zum Finden des Optimums kommt die Methode der *Loopy Belief-Propagation* zum Einsatz. Dieses Verfahren basiert auf ähnlichen Algorithmen wie z. B. in [Tao u. a., 2001; Hong und Chen, 2004; Bleyer und Gelautz, 2005b,a; Deng u. a., 2005] vorgestellt. Während der grundlegende Ablauf stets ähnlich ist, unterscheiden sich die Methoden in der Art der initialen Ebenenzuweisung, den verwendeten Kostenmaßen sowie den Optimierungsverfahren. Allen gemein ist eine sehr gute Leistung im Sinne der Middlebury-Evaluation.

Globale Optimierungsverfahren, sei es auf Pixelebenen oder segmentbasiert, sind sehr rechenaufwändig. Viele dieser Verfahren benötigen eine lokale Stereo-Korrespondenzsuche, um initiale Disparitäten zu bestimmen. Die weiteren Bestandteile der Algorithmen sind zumeist noch aufwändiger. Sie beinhalten, wie eben beispielhaft beschrieben, nochmals mehrere Berechnungen lokaler Kostenmaße. Außerdem benötigen die Verfahren zur Segmentierung sowie zum Finden des Energieminimums sehr viele Recheniterationen. Fazit ist, dass zum Zeitpunkt der Recherchen solche Algorithmen zwar eine gute Qualität der Disparitätskarte liefern, jedoch wegen langer Berechnungszeiten nicht für den in dieser Arbeit geschilderten Anwendungsfall geeignet sind.

Selbst ohne die Anforderungen an eine Echtzeit-Darstellung[5] ist allein die Anwendung auf Sequenzen zur Erzeugung von Testvideos kaum in einem vernünftigen Zeitrahmen durchführbar. Eine Beschleunigung durch Parallelisierung der Algorithmen ist nur bedingt erreichbar, da insbesondere die Nutzung von Nachbarschaftsbeziehungen bei der Energieminimierung dem entgegen steht.

Durch die Motivation, dennoch einen globalen Algorithmus in die Betrachtungen mit einzubeziehen, führten weitere Recherchen zu einem von Andreas Geiger vorgestellten Verfahren für eine schnelle globale Stereo-Korrespondenzsuche [Geiger u. a., 2010]. Im Verfahren wird ebenso der in Abschnitt 2.2.6 vorgestellte bayessche Ansatz der Energieminimierung zur Disparitätsbestimmung genutzt. A-priori-Wahrscheinlichkeit und bedingte Wahrscheinlichkeit (Likelihood) (vgl. Gl. (2.60) und (2.61)) werden dabei geschickt gewählt und eingegrenzt. So wird der Lösungsraum der Optimierung eingeschränkt und die folgende MAP-Schätzung zur Energieminimierung beschleunigt. Ausgehend von robusten Merkmalen der Filterantwort eines Sobel-Operators werden robuste Korrespondenzen bestimmt. Auf diese Unterstützungspunkte (*support points*) wird eine Delaunay-Triangulierung durchgeführt. Die stückweise, lineare Repräsentation dient im Modell der A-priori-Wahrscheinlichkeit als Mittelwert einer gaußschen Verteilung, die mit einer Gleichverteilung addiert wird. Die Disparitäten in Bereichen ohne Unterstützungspunkte werden demnach aus den durch die Unterstützungspunkte aufgespannten Ebenen interpoliert. Diese Modell ähnelt der Repräsentation durch Ebenen bei den segmentbasierten Verfahren, nutzt jedoch keine farbbasierte Segmentierung. Als bedingte Wahrscheinlichkeit wird eine durch die Epipolarbedingung beschränkte Laplace-Verteilung genutzt, ebenfalls unter der Verwendung von Sobel-Merkmalen zwischen linkem und rechtem Bild. Für die konkrete Herleitung der zu minimierenden Energiefunktion sei auf [Geiger u. a., 2010] verwiesen. Sie ergibt sich für die linke Disparitätskarte zu:

$$E(d) = \beta \left\| \mathbf{f}^{(l)} - \mathbf{f}^{(r)}(d) \right\|_1 - \log \left[ \gamma + \exp \left( -\frac{[d - \mu(\mathbf{S}, \mathbf{o}^{(l)}]^2}{2\sigma^2} \right) \right] \tag{5.21}$$

Die $L_1$-Norm der Merkmalsvektoren $\mathbf{f}$ repräsentiert die Likelihood-Energie bzw. im Sinne von Gl. (2.65) den Datenterm. Die A-priori-Energie und somit der Glattheitsterm wird durch die stückweise lineare Funktion $\mu(\mathbf{S}, \mathbf{o}^{(l)})$ determiniert, wobei $\mathbf{S}$ die Menge der Unterstützungspunkte und $o^{(l)}$ eine Beobachtung, bestehend aus der Pixelposition und dem Sobel-Merkmal, darstellen. Aufgrund der berichteten Berechnungsgeschwindigkeit, des für die Anwendung vielversprechenden Ansatzes sowie einer verfügbaren und somit verifizierbaren Implementierung wurde das Verfahren für diese Arbeit unverändert angewendet.

Ebenso wurde die Dynamische Programmierung, wie im Grundlagenkapitel im Abschnitt 2.2.6 vorgestellt, in Betracht gezogen. Da selbst die Variante mit nur drei Vorgängern bei der Akku-

[5] Echtzeit im Sinne der erforderlichen Bildwiederholrate

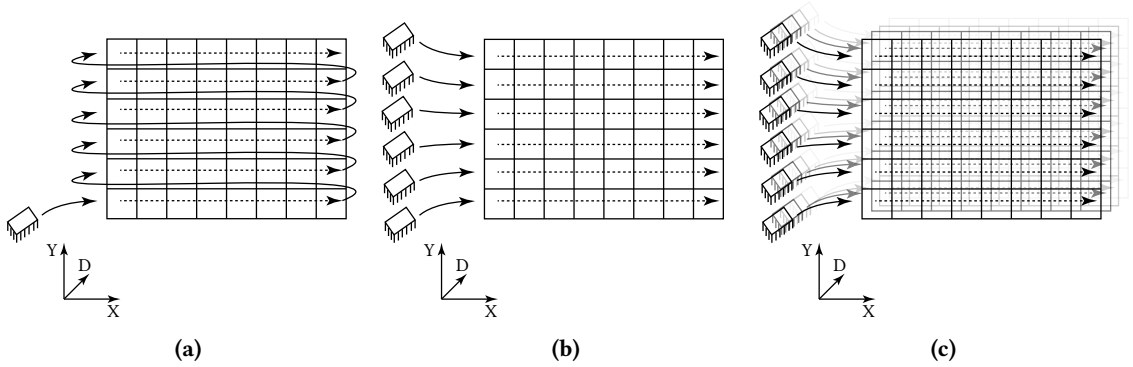
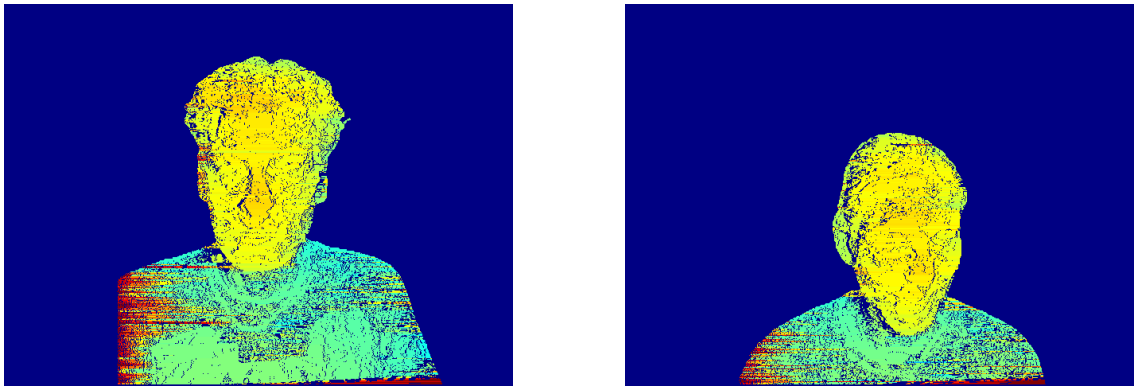



**Abbildung 5.8.:** Schematische Darstellung der Abarbeitung des Disparitätsraumbildes bei der dynamischen Programmierung. (a) Konventionell, zeilenweise Abarbeitung mit einer Berechnungseinheit. (b) Parallelisierung mit Aufteilung auf Zeilen. (c) Entwickelte Parallelisierung mit Aufteilung auf Zeilen und Disparitätsebenen.

mulation (vgl. Gl. (2.69)) einen hohen Berechnungsaufwand erfordert, wurde diese effizient auf der GPU implementiert. Das Prinzip ist in Abb. 5.8 schematisch dargestellt. Eine Parallelisierung des Vorgangs auf Pixelbasis ist nicht möglich, da aktuell zu berechnende Werte von vorherigen akkumulierten Werten abhängen. Der triviale Ansatz zur Lösung diese Problems ist, eine Parallelisierung auf Zeilenbasis vorzunehmen (Abb. 5.8b). Ergibt diese Maßnahme zwar schon eine Beschleunigung, lastet sie jedoch noch immer nicht die Masse der verfügbaren Recheneinheiten einer GPU aus. Um den Vorgang weiter zu beschleunigen, wird die Parallelisierung weiter erhöht, indem auf einer Zeile mehrere Recheneinheiten gleichzeitig arbeiten (Abb. 5.8c). Die gemeinsam genutzten Datenmengen, die sie dabei benötigen, sind gering genug, um unter Verwendung des lokalen schnellen Speichers der GPU zwischengespeichert zu werden. Diese Maßnahme ermöglicht eine sehr effiziente Implementierung des DP-Ansatzes. Eine genaue Beschreibung der Umsetzung ist in [Treutner, 2010] nachzulesen und wurde vom Autor in [Weigel und Treutner, 2011] veröffentlicht. Stichprobenartige Tests zeigten jedoch schnell, dass der Ansatz mit dem verwendeten Ausgangsmaterial die typischen Artefakte erzeugt (vgl. Abb. 5.9). Insbesondere an Disparitätssprüngen wie bei Objektkanten erfolgte oft eine Fortpflanzung der Disparitätswerte. Auch neigt der Algorithmus durch die Annahme geringer Disparitätsveränderungen zu einer Favorisierung frontalparalleler Ebenen (vgl. auch [Birchfield und Tomasi, 1999]). Dies ist insbesondere im Gesichtsbereich als kritisch einzustufen. Eine Erweiterung auf ein Modell mit mehr Ebenen in der Akkumulation erhöht wiederum den Berechnungsaufwand und kann nur die Disparitätsfortpflanzung verringern, weniger die Annahmen frontalparalleler Ebenen. Auch ist dessen Beschleunigung durch Parallelisierung nicht mit o. g. Schema durchführbar. Unter diesen Gesichtspunkten und wegen des geringen Mehrwertes, der durch einen vielfach erhöhten Rechenaufwand erzeugt wird, wurde eine Nutzung dieses Ansatzes für die weitere Arbeit ausgeschlossen.



(a) (b)

**Abbildung 5.9.:** Artefakte unter Verwendung der einfachen Dynamischen Programmierung zur Disparitätsbestimmung. In beiden Varianten wurde bereits ein robustes Kostenmaß (NCC) sowie eine Kostenraumaggregation verwendet. Der Vordergrund (die Person) wurde aus den Disparitätskarten des gesamten Bildes durch die Alphamasken freigestellt. Dunkelblaue Bereiche innerhalb der Person sind verdeckt markierte Schritte des DP-Algorithmus und haben keine Disparitätszuweisung.

### 5.2.4. Lokale Korrespondenzsuche

Die grundlegenden Methoden der lokalen Korrespondenzsuche und der dabei verwendeten Kostenmaße wurden bereits im Abschnitt 2.2.5 vorgestellt. Das Hauptproblem dieser Methoden ist die Anfälligkeit für Rauschen bei der Verwendung kleiner Filtermasken (Fenster). Die Vergrößerung des Fensters führt zu robusteren Ergebnissen, bringt jedoch auch besagte Vordergrunderweiterungen (*foreground fattening*) und Kantenverwischung (*border bleeding*) mit sich. Die zunächst innovative, jedoch schwierig umzusetzende Verbesserung durch die Nutzung adaptiver Fenster wurde 2005 durch die die Nutzung adaptiver Filtergewichte (ASW) abgelöst. Diese Methode stellt den aktuellen Stand der Technik bei der lokalen Korrespondenzsuche dar. Ähnlich wie die normalisierte Kreuzkorrelation (NCC) stellt sie eine Kombination aus reinem Kostenmaß und Aggregationsschritt dar. Wenn auch einfacher in der Handhabung als adaptive Fenster, so stellt der Ansatz hohe Anforderungen an die Rechenleistung, da für jeden betrachteten Pixel ein eigener Filterkernel berechnet werden muss. Ansätze, diesen Vorgang zu beschleunigen gibt es, sie sind jedoch noch immer recht aufwändig in der Umsetzung und müssen Kompromisse eingehen [Yu u. a., 2010].

Hinsichtlich der konkreten Anwendung in dieser Arbeit muss überprüft werden, ob ein solcher Aufwand gerechtfertigt ist. Seine Stärken spielt ASW insbesondere in Szenen mit vielen Tiefensprüngen aus. Dies ist bei vielen gestaffelten, kleineren Objekten vor dem Hintergrund, wie bspw. in vielen Szenen der Middlebury-Testdaten, der Fall. Unter der Nutzung segmentierter Eingangsbilder ist im Videokommunikationsszenario allein die kommunizierende Person von Interesse.

Abgesehen vom Übergang Kopfes zum Körper weisen Gesichts- und Körperbereich keine Tiefensprünge auf. Auch die farbliche Ähnlichkeit der Regionen, welche die Filterkoeffizienten des ASW determiniert, ist eher konstant im Gesichtsbereich. Im Körperbereich könnten, z. B. durch eine stark texturierte Kleidung, sogar nicht vorhandene Tiefensprünge vom Algorithmus als solche „interpretiert“ werden. Die aufwändige Verwendung eines ASW-Ansatzes steht in diesen Szenen demnach in keinem Verhältnis zum Nutzen. Zudem wird bei ASW üblicherweise ein Kostenmaß wie die die absolute Differenz verwendet, die im Vergleich zu anderen Maßen empfindlich gegen Rauschen und Helligkeits- und Farbunterschiede zwischen linkem und rechtem Bild ist. Jedoch sind gerade diese Unterschiede bei den verwendeten Webcams vorhanden.

Die Frage nach einer geeigneten Kombination aus Kostenmaß, Korrespondenzalgorithmus und Vor- sowie Nachbearbeitungsschritten bleibt somit bestehen. In [Hirschmüller und Scharstein, 2009] werden verschiedene dieser Kombinationen hinsichtlich ihrer Eignung in Szenen mit radiometrischen Unterschieden untersucht. Als Fazit der Evaluation schneiden Verfahren mit bilateraler Filterung [Tomasi und Manduchi, 1998] und Hintergrundsubtraktion am besten ab [Ansar u. a., 2004]. Das Verfahren kann radiometrische Unterschiede reduzieren, ohne die Tiefenkanten zu verwischen. Die Nutzung der adaptiven Filterkoeffizienten kommt hier nicht bei der Aggregation, sondern bei der Vorverarbeitung der Stereobilder zum Einsatz. Der Aufwand der Berechnung der Filterkoeffizienten bleibt jedoch. Auch die ZNCC (*zero-mean cross correlation*) erzielt gute Ergebnisse. Sie erzeugt Verwischungen an Tiefenkanten, was jedoch für die in dieser Arbeit betrachtete Anwendung aus o. g. Gründen weniger kritisch ist. Letztlich wird in [Hirschmüller und Scharstein, 2009] auch noch die Census-Transformation [Zabih und Woodfill, 1994] als positiv hervor gehoben, jedoch wird dem Kostenmaß Versagen bei starkem Rauschen attestiert. Damit kommt es für die Anwendung auf Videos von Webcams nicht in Frage.

All diese Betrachtungen decken sich mit exemplarischen Untersuchungen mit dem Testmaterial. Sowohl Rauschen als auch konstante Helligkeitsunterschiede können durch die ZNCC am besten kompensiert werden. Die Nachteile sind anwendungsbezogen als weniger kritisch zu bewerten, obschon sie natürlich qualitätsbeeinflussend sind. Der Berechnungsaufwand ist höher als bei anderen Kostenmaßen, wird jedoch wegen der sehr guten Leistung akzeptiert. Dieser Kompromiss wird als am akzeptabelsten bewertet. Formal wird demnach folgendes Kostenmaß unter Verwendung eines lokalen Korrespondenzanalysealgorithmus verwendet:

$$C_{ZNCC}(\mathbf{p},\hat{\mathbf{p}}) = \frac{\sum_{\mathbf{q}\in N_\mathbf{p}}((I_L(\mathbf{q})-\bar{I}_L(\mathbf{p}))(I_R(\hat{\mathbf{q}})-\bar{I}_R(\hat{\mathbf{p}})))}{\sqrt{\sum_{\mathbf{q}\in N_\mathbf{p}}(I_L(\mathbf{q})-\bar{I}_L(\mathbf{p}))^2 \sum_{\mathbf{q}\in N_\mathbf{p}}(I_R(\hat{\mathbf{q}})-\bar{I}_R(\hat{\mathbf{p}}))^2}} \tag{5.22}$$

Die Notation entspricht Tabelle 2.2 und ist für die Suche vom linken in das rechte Bild formuliert. Die Gleichung gilt für Intensitätsbilder (Farbinformation verworfen) und entsprechend auch für die Suche von rechts nach links. Entgegen der einfachen NCC werden bei der ZNCC zusätzlich die Mittelwerte $\bar{I}_L$ und $\bar{I}_R$ der jeweiligen Umgebung subtrahiert, was das Verfahren robust gegen

gleichmäßige Helligkeitsänderungen macht. Dies ist z. B. bei den verwendeten Webcams in Form eines „Pumpens“ des Bildes vorhanden, welches trotz fester Einstellung aller Kameraparameter nicht zu beseitigen war. Da Gleichung (5.22) Kostenwerte im Bereich von $-1 \leq C_{ZNCC} \leq 1$ erzeugt, werden diese mittels

$$C_{ZNCC,norm} = \frac{1}{2}(1 - C_{ZNCC}); \tag{5.23}$$

auf einen Wertebereich von $0 \leq C_{ZNCC,norm} \leq 1$ normiert, womit sie auch der gängigen Annahme, dass niedrige Kosten die beste Korrespondenz darstellen, entsprechen. Eine Implementierung des Kostenmaßes kann einfach parallelisiert werden. Zusätzlich kann durch die vorherige Berechnung der Mittelwerte und die Wiederverwendung der Differenzen eine weitere Beschleunigung erzielt werden. Es zeigte sich, dass Berechnungszeiten einer GPU-Implementierung bei kleinen Fenstergrößen bereits nahe denen einer Echtzeitanforderung kommen können.

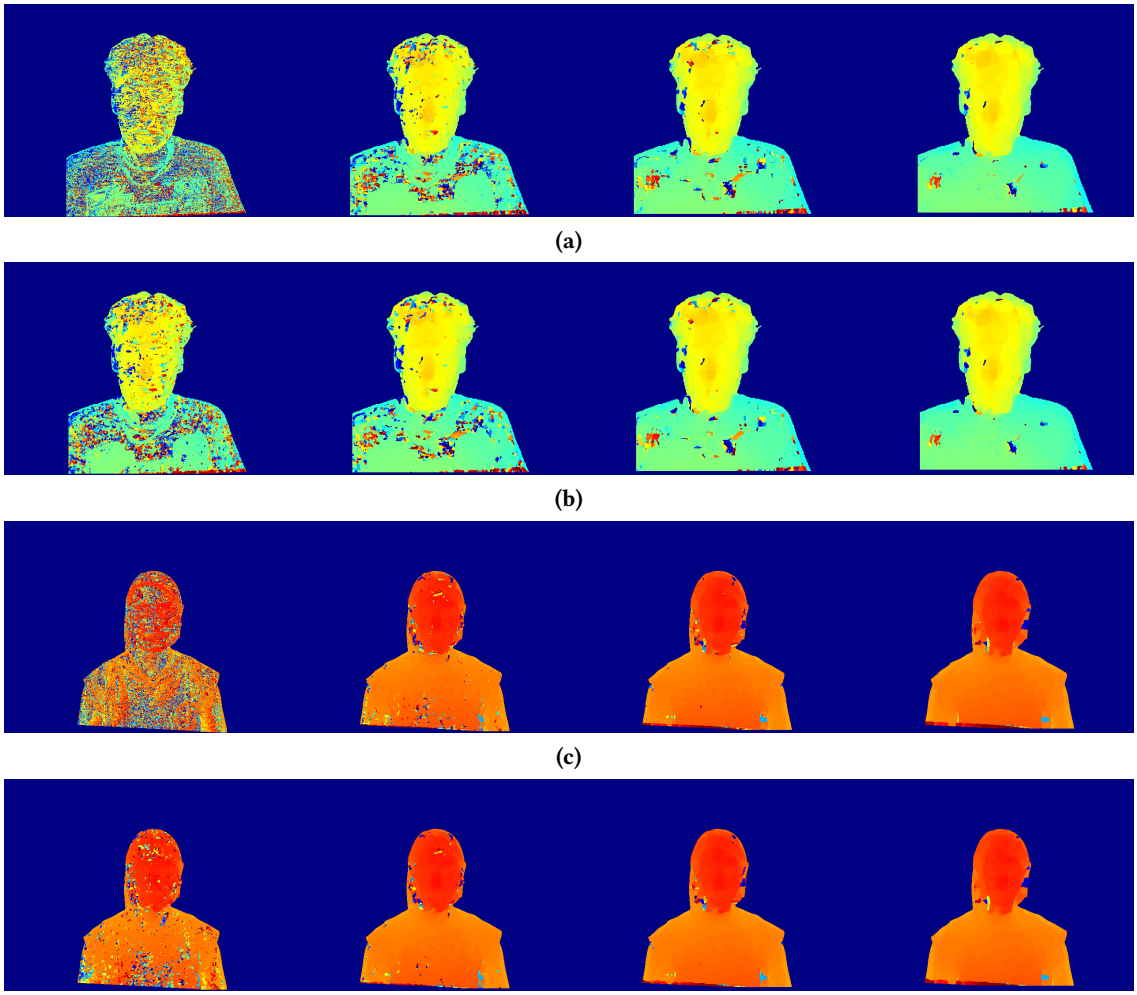
Die Wahl der Fenstergröße und somit für $N_{\mathbf{p}}$ ist entscheidend für die Leistung des Kostenmaßes. Der Umstand wurde bereits im Grundlagenkapitel erläutert. Eine allgemeingültige Auswahl ist schwer zu treffen, da die Genauigkeit der Disparitätsbestimmung in starkem Maße von der Szene abhängt. So ist die Disparitätskarte einer stark mit nicht wiederholenden Mustern texturierten Szene (z. B. eine Person mit Sommersprossen und einem Hawaii-Hemd) wesentlich besser zu bestimmen als eine wenig texturierte. Neben den homogenen Regionen würde etwa bei einem schwarzen, einfarbigen T-Shirt auch noch das Rauschen stärker ins Gewicht fallen.

Versuche mit 10 möglichst heterogenen Szenen aus der Testdatenmenge bestätigen diese Aussage. Führen bei einigen Szenen bereits kleine Fenster zu subjektiv sauber wirkenden Disparitätskarten, so erzeugen sie bei anderen Szenen noch extrem unruhige und zufällige Disparitätskarten. Diese Artefakte treten primär aufgrund von verrauschten homogen Flächen auf. Sie lassen sich nur mittels größerer Fensterausmaße reduzieren. Da insbesonders starke Sprünge zwischen benachbarten Disparitäten zu starken Deplatzierungen in der Synthese führen (vgl. Abschnitt 3.3), wurde die Fenstergröße für die Korrespondenzsuche auf eine Größe von $21 \times 21$ Pixeln um den Aufpunkt gewählt. Abb. 5.10a und 5.10c veranschaulichen den Sachverhalt für zwei unterschiedliche Testsequenzen.

Die Nutzung von Farbinformationen bei der ZNCC ergibt kaum Verbesserungen [Hirschmüller und Scharstein, 2009], verdreifacht jedoch den Aufwand für die Berechnung. Daher werden die Eingangsbilder $\hat{I}_L$ und $\hat{I}_R$ für die weitere Verarbeitung zu einem Intensitätsbild (8 `bit` natürliche Zahlen) transformiert. Für linkes und rechtes Bild gilt:

$$\dot{I}(\mathbf{p}) = 0.2989\hat{I}_r(\mathbf{p}) + 0.5870\hat{I}_g(\mathbf{p}) + 0.1140\hat{I}_b(\mathbf{p}) \tag{5.24}$$

Ergänzend zu der Auswahl des Kostenmaßes wird eine Vorfilterung der Stereo-Bilder durchgeführt. Der Grund dafür ist die bereits mehrfach erwähnte Neigung des Kamerasystems zu



**(a)** **(b)** **(c)** **(d)**

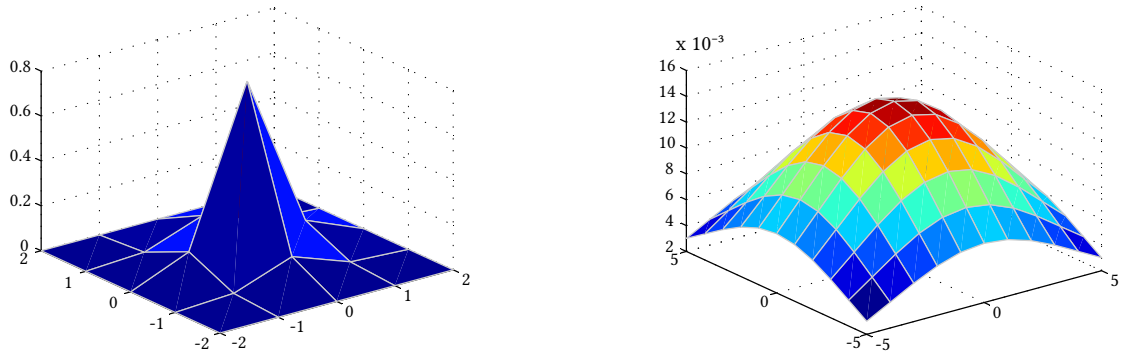
**Abbildung 5.10.:** Fenstergröße und Kostenaggregation bei lokaler Korrespondenzsuche. Abb. (a) und (c): Disparitätskarten bestimmt durch lokale Korrespondenzsuche mit ZNCC. Verwendete Fenstergrößen ( px) von links nach rechts: $3 \times 3$, $9 \times 9$, $15 \times 15$ und $21 \times 21$. Mit steigender Fenstergröße wird die Korrespondenzsuche akkurater, jedoch werden die Objektkanten, vor allem im Kinnbereich, zunehmend verwischt. Die Szene in Abb. (a) beinhaltet dunkle verrauschte Bereiche und bedarf größerer Fenster als die Szene in Abb. (c). Abb. (b) und (d): Nutzung derselben Fenstergröße und desselben Kostenmaßes, jedoch mit gaußscher Aggregation des Kostenraumes.

Rauschen. Um dieses Problem zu reduzieren, wird *vor* der Korrespondenzsuche ein Gauß-Filter auf linkes und rechtes Intensitätsbild $\dot{I}_{L,R}$ angewendet:

$$I_L(\mathbf{p}) = \dot{I}_L(\mathbf{p}) * G(u,v) \qquad \text{und} \qquad I_R(\mathbf{p}) = \dot{I}_R(\mathbf{p}) * G(u,v) \tag{5.25}$$

mit

$$G(u,v) = \frac{\exp(\frac{-(u^2+v^2)}{2\sigma^2})}{\sum_u \sum_v \exp(\frac{-(u^2+v^2)}{2\sigma^2})}\text{, mit } u,v \in \mathcal{H} \tag{5.26}$$




**(a)** **(b)**

**Abbildung 5.11.:** (a) Gaußscher Filterkernel zur Vorverarbeitung. (b) Gaußscher Filterkernel bei der Kostenglättung.

Die Größe des Filterfensters wird durch Setzen von $\mathcal{H} = \{-2 \dots 2\}$ auf eine Umgebung von $5$ px $\times$ $5$ px Pixeln festgelegt, und dessen Wirkung mit $\sigma = 0.5$ eher konservativ gesetzt. So fällt der „Verwischungseffekt“ durch die Weichzeichnung und somit eine Beeinflussung der Kostenberechnung nicht zu stark aus. Der Filterkernel ist in Abb. 5.11a dargestellt.

Die durch die Gleichung (5.22) und (5.23) berechneten Kosten erzeugen das Disparitätsraumbild (vgl. Abschnitt 2.2.3). Die in der Taxonomie nun folgende Kostenaggregation ist beim gewählten Kostenmaß bereits implizit durch die Einbeziehung der Nachbarschaft $N_{\mathbf{p}}$ gegeben. Dennoch ist eine nochmalige Glättung der Kosten sinnvoll, wie z. B. auch in [Criminisi u. a., 2003] berichtet. Insbesondere Ausreißer in homogenen Regionen können so wirkungsvoll vermieden werden. So ist der Effekt besonders bei kleineren Fenstergrößen zu beobachten, wie Abb. 5.10b und 5.10d veranschaulichen. Der positive Effekt ist bei kleinen Fenstern $N_{\mathbf{p}}$ der ZNCC enorm. Er kann jedoch selbst bei einem großen Fenster von $21 \times 21$ noch Ausreißer beseitigen. Die Filterung der Kosten erfolgt für jede $d$-Ebene des Disparitätsraumbildes ebenfalls mittels eines Gauß-Filters. Unter Verwendung der Gleichung (5.26) ergibt sie sich zu:

$$\hat{C}(\mathbf{p}, d) = C(\mathbf{p}, d) * G(u, v) \;, \quad \forall d \in \{d_{min}, \dots, d_{max}\} \tag{5.27}$$

In der Gleichung wird $d$ im Gegensatz zur bisherigen Notation explizit aufgeführt. Die Größe des Kernels wird für den Zweck der Kostenglättung erhöht. Dabei ist wichtig, dass er nicht größer als das Fenster $N_{\mathbf{p}}$ der ZNCC ist, um einen noch stärke Verwischung an Objektkanten mit Tiefensprüngen zu vermeiden. Für die Experimente in dieser Arbeit wurde ein $11$ px $\times$ $11$ px-Fenster ($\mathcal{N} = \{-5 \dots 5\}$) mit $\sigma = 4$ gewählt. Letzteres führt zu einem flacheren Anstieg der Filterfunktion (vgl. Abb. 5.11b).

Die Disparitätskarten für linkes und rechtes Bild $D_{L,R}$ lassen sich nun aus dem jeweiligen DSI

mittels WTA ( *winner-takes-all*) ermitteln:

$$D(x,y), D(\mathbf{p}) = \underset{d \in \mathcal{D}}{\operatorname{argmin}} \hat{C}(\mathbf{p}, d), \text{ mit } \mathcal{D} = \{d_{min}, \ldots, d_{max}\} \tag{5.28}$$

Algorithmus und Parameter der untersuchten und entwickelten Stereoanalyseverfahren sind somit definiert. Tabelle 5.1 fasst sie abschließend zusammen. In den folgenden Abschnitten werden Maßnahmen zur Verbesserung der Disparitätskarte durch Algorithmen der Nachverarbeitung vorgeschlagen. Zur Erinnerung sei erwähnt, dass diese Nachverarbeitung der Disparitätskarten im Kontext der Systematik aus Abschnitt 3.1 noch immer zum Block Vorverarbeitung gehört, der die sekundären Repräsentation erzeugt.

### 5.2.5. Konsistenzprüfung und Füllen

Wie das erste subjektive Experiment ergab, welches im Abschnitt 6.3 vorgestellt wird, sind die kritischsten Artefakte im synthetisierten Bild Löcher oder Fehlplatzierungen von Pixeln oder Pixelbereichen. Diese können, wie in Abs. 3.3 erläutert, aufgrund falscher Disparitäten entstehen. Um solche Disparitäten zu eliminieren, wird der eigentlichen Korrespondenzanalyse eine Konsistenzprüfung zwischen der linken und rechten Disparitätskarte nachgeschaltet. Das Prinzip basiert auf einem einfachen Schwellwertvergleich. Fehlerhafte Disparitäten am Rand würden zu einem offenen Loch in der Silhouette führen. Der anschließende Füllalgorithmus kann dann nicht mehr korrekt arbeiten. Daher wird zunächst eine „Sicherheitszone“ definiert, in der keine Konsistenzprüfung stattfindet. Dies geschieht durch Erosion der originalen binären Masken $M_{L,R}$. Beispielhaft für das linke Bild demonstriert ergibt dies:

$$\hat{M}_L(\mathbf{p}) = (M_L(\mathbf{p}) \ominus S) \tag{5.29}$$

wobei $S$ ein kreisförmiges strukturierendes Element mit einem Radius von $5$ ist. Nun lässt sich eine binäre Ausschlusskarte erstellen, die ungültige Disparitäten markiert. Im Beispiel des linken Bildes ergibt sie sich zu

$$R_L(x,y) = \begin{cases} 1, \text{ wenn } |D_L(x,y) - D_R(x - D_L(x,y), y))| > \xi \ \wedge \hat{M}_L(x,y) = 1 \\ 0, \text{ sonst} \end{cases} \tag{5.30}$$

Äquivalent kann $R_R$ ermittelt werden. Aufgrund der aus Gründen der Übersichtlichkeit gemischt verwendeten Notation von Funktionsargumenten als Vektor bzw. als Vektorelemente sei an dieser Stelle noch einmal darauf hingewiesen, dass $\mathbf{p} = (x,y)^T$ gilt.

Anschließend müssen die fehlenden Disparitäten möglichst originalgetreu gefüllt werden. Eine übliche Methode ist die Fortführung (*propagation*) der Randdisparitäten in den leeren Bereich

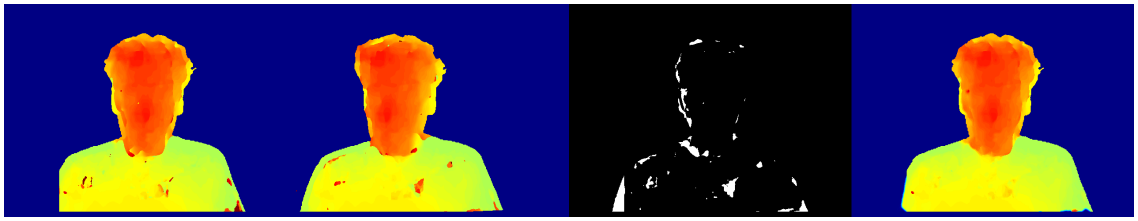


**Abbildung 5.12.:** Füllung der Disparitätskarte auf Basis der links-rechts Konsistenzprüfung. V. l. n. r.: linke Disparitätskarte, rechte Disparitätskarte, binäre Ausschlusskarte $R_L$, gefüllte linke Disparitätskarte.

entlang der Bildzeilen. Die kann jedoch zu ähnlichen Streifenartefakten wie bei der dynamischen Programmierung führen. Auch können Objektgrenzen, wie z. B. zwischen Kinn und Hals- bzw. Körperregion verwischt werden, wenn das Loch gerade in einen solchen Bereich führt. Daher kommt in dieser Arbeit ein besserer und dennoch nicht allzu rechenintensiver Algorithmus zum Einsatz, der 2004 von Telea vorgestellt wurde [Telea, 2004]. Grundprinzip des Verfahrens ist das Fortführen der Pixelwerte in die ungefüllte Region. Der Wert des zu füllenden Pixels $\mathbf{p}$ wird dabei aus den gewichteten Werten bekannter Pixel $\mathbf{r}$ in einer bekannten Region $B_\epsilon$ berechnet:

$$I(\mathbf{p}) = \frac{\sum_{\mathbf{r}\in B_\epsilon} w(\mathbf{p},\mathbf{r})\left[I(\mathbf{r}) + \nabla I(\mathbf{r})(\mathbf{p}-\mathbf{r})\right)]}{\sum_{\mathbf{r}\in B_\epsilon} w(\mathbf{p},\mathbf{r})} \tag{5.31}$$

Die Wichtungsfunktion $w$ ist entscheidend für die Qualität der Füllung. Sie ist das Produkt dreier Wichtungsterme, welche aus den Pixeln der Umgebung $B_\epsilon$ bestimmt werden. Dabei handelt es sich um eine Richtungskomponente, eine geometrische Entfernungskomponente und eine Kontur-Entfernungskomponente. Die Füllung erfolgt nun derart, dass der Randbereich gleichmäßig in das Loch propagiert wird. Dabei werden die unbekannten Pixel in der Reihenfolge entsprechend ihrem Abstand zum Rand gefüllt. Diese Methode wird als *Fast Marching Method* bezeichnet. Für Details sei auf [Telea, 2004] verwiesen. Abbildung 5.12 veranschaulicht den Vorgang für ein Disparitätspaar.

### 5.2.6. Zeitliche Glättung der Disparitätskarte

Die bisher angewendeten Algorithmen betrachten ausschließlich einzelne Bilder der Videosequenz. Zusammenhänge zwischen zeitlich aufeinanderfolgenden Bildern wurden nicht genutzt. Unter der Annahme einer fehlerfrei bestimmten Disparitätskarte ist die Nutzung solcher Zusammenhänge auch nicht notwendig. Die aus Einzelbildern extrahierte sekundäre Repräsentation wäre Bild für Bild korrekt. Fehler, die sich aus deren Erzeugung ergeben, wären nicht vorhanden und würden somit keine Artefakte in der Zielrepräsentation verursachen. Eine zeitliche Betrachtung wäre, wenn überhaupt, nur für prinzipbedingte Fehlerursachen sinnvoll.

Relevant wird die zeitliche Betrachtung jedoch, wenn davon ausgegangen werden muss, dass

algorithmenbezogene Fehlerursachen existieren. Da insbesondere die Stereoanalyse ein noch ungelöstes Problem der Bildanalyse darstellt, ist dies auch in dieser Arbeit notwendig. Die Fragestellung bei dieser Betrachtung ist, ob der Charakter von Fehlern in Einzelbildern durch deren schnelle zeitliche Abfolge zu zusätzlichen Bildartefakten führt. Die Fehlerursachen für falsch detektierte Disparitäten wurden bereits öfter genannt. In dieser Arbeit sind vor allem durch Rauschen und homogene Flächen Fehlschätzungen möglich. Da das thermische Rauschen des Sensors ein Zufallsprozess ist, haben dadurch verursachte Artefakte ebenfalls einen zufälligen Charakter. Wird dies auf die Auswirkungen von Fehlschätzungen in der Stereoanalyse übertragen (vgl. Abschnitt 3.3), so besteht die Möglichkeit sich über die Zeit zufällig verändernder Löcher, Fehlpartikel oder Verzerrungen. Während erstere in der Sequenz Flimmern oder Flackern erzeugen, verursacht zuletzt genanntes eher ein Zittern. Die in Abschnitt 6.3.3 vorgestellten Ergebnisse des ersten Experiments bestätigen die Relevanz dieser Fehler durch die häufige Nennung solcher Artefakte.

Primäres Ziel ist selbstverständlich die weitgehende Vermeidung von Fehlschätzungen durch die in den vorherigen Abschnitten entwickelten Maßnahmen. Es wird dennoch eine Methode entwickelt um auch über die Zeit vorkommende Fehler zu reduzieren. Dies muss derart erfolgen, dass die Auswirkungen der durch sie verursachten Artefakte auf die wahrgenommene Qualität der Zielrepräsentation abgeschwächt werden. Eine denkbare Vorgehensweise ist die zeitliche Glättung der Disparitätskarten über ein bestimmtes Zeitfenster.

Ein erster Ansatz innerhalb dieser Arbeit war die Verwendung eines einfachen Mittelwertfilters. Basierend auf einem oberen und unteren Schwellwert wird entschieden, ob zeitlich aufeinander folgende Pixel überblendet werden.

$$\dot{D}_t(\mathbf{p}) = \begin{cases} D_{t-1}(\mathbf{p}) & \text{, wenn } |D_t(\mathbf{p}) - D_{t-1}(\mathbf{p})| < \tau_l \\ \frac{D_{t-1}(\mathbf{p}) + D_t(\mathbf{p})}{2} & \text{, wenn } \tau_l \leq |D_t(\mathbf{p}) - D_{t-1}(\mathbf{p})| < \tau_u \\ D_t(\mathbf{p}) & \text{, wenn } \tau_u \leq |D_t(\mathbf{p}) - D_{t-1}(\mathbf{p})| \end{cases} \tag{5.32}$$

Die Übernahme des vorherigen Pixels bei Schwellwerten kleiner als $\tau_l$ wird fest auf maximal 30 aufeinander folgende Bilder begrenzt. Anschließend erfolgt eine Überblendung. Ist die Differenz größer als $\tau_u$ so wird dies als eine Bewegung der auf dieser Pixelposition abgebildeten Objekte interpretiert und eine Überblendung vermieden.

Dieser einfache Ansatz kann bei Szenen mit wenig Bewegung eine zeitliche Glättung erzeugen. Bei einer Szene mit starker Bewegung der Person würde bei niedrig gewähltem $\tau_u$ jedoch keine Glättung erfolgen. Bei Erhöhung des oberen Schwellwertes entstehen wiederum Weichzeichnungseffekte durch Überblendung nicht zusammengehöriger Objekte, die als falsche Disparitäten neue Artefakte erzeugen würden.Um diesen Effekt zu vermeiden, wird vor der Glättung eine Bewe-

gungskompensation durchgeführt. Die Bewegungsschätzung erfolgt auf Basis einer einfachen blockbasierten Bewegungsschätzung in der rektifizierten Originalsequenz.

Nach einer Umwandlung des aktuellen RGB-Bildes in ein Intensitätsbild entsprechend Gleichung (5.24) erfolgt eine Unterteilung des Bildes in Blöcke der Größe $w \times h$. Anschließend erfolgt eine Blocksuche für jeden Block des aktuellen Bildes $I_t$ innerhalb des vorherigen Bildes $I_{t-1}$ mittels NCC. Dabei werden nur Blöcke betrachtet, die sich mit der Alphamaske $M$ überschneiden. Die Gleichung der einfachen NCC lautet

$$R_{NCC}(x,y) = \frac{\sum_{u,v} I_t(u,v) I_{t-1}(x+u, y+v)}{\sqrt{\sum_{u,v} I_t(u,v)^2 \sum_{u,v} I_{t-1}(x+u, y+v)^2}} \quad \text{mit} \quad \begin{array}{ll} u = \{0, \ldots, w-1\} & \text{und} \\ v = \{0, \ldots, h-1\} & \end{array} \tag{5.33}$$

Die Blockgröße wird dabei auf $16$ px $\times$ $16$ px festgelegt, wobei der Referenzpunkt in der linken oberen Ecke liegt. Die Suche innerhalb des gesamten vorherigen Bildes würde einen hohen Rechenaufwand bedeuten. Daher wird der Suchbereich $S$ in $I_{t-1}$ unter der Annahme nicht allzu schneller Bewegungen und für die genutzte Bildgröße auf eine Umgebung von $5 \times 5$ Blöcken um den aktuell betrachteten Block mit dem Referenzpunkt $(x_b, y_b)^\top$ gesetzt, was einem Bereich von $80$ px $\times$ $80$ px entspricht. Der Aufpunkt $(\hat{x}, \hat{y})$ des ähnlichsten Blocks wird anschließend über die maximale Korrelation bestimmt

$$(\hat{x}, \hat{y})^\top = \underset{x,y \in S}{\operatorname{argmax}} R_{NCC}(x,y) \tag{5.34}$$

Somit lässt sich der Bewegungsvektor $\mathbf{b} = (x_b, y_b)^T - (\hat{x}, \hat{y})^\top$ für jeden betrachteten Block bestimmen. Ähnlich der Stereo-Analyse sind auch hier Rauschen und homogen texturierte Regionen Ursache für Fehlbestimmungen von Bewegungsvektoren. Zur Reduzierung solcher Fehler werden die Bewegungsvektoren mittels einer normierten Gaußfunktion geglättet. Ausreißer, für die $|\mathbf{b}| > 2\bar{\nu}$ mit $\bar{\nu}$ als durchschnittliche Länge aller betrachteten Bewegungsvektoren gilt, leisten keinen Beitrag zur Filterung. Die Nicht-Einbeziehung dieser Ausreißer geschieht unter der Annahme, dass sich nur die Person im Bild bewegt und sich somit der Betrag der Bewegungsvektoren nicht zu stark unterscheiden sollte.

$$\dot{\mathbf{b}}(i,j) = \sum_{k=-m}^{m} \sum_{l=-n}^{n} g(k,l) \mathbf{b}(i-k, j-l) \qquad \text{mit} \quad \mathbf{b}(i,j) = 0 \ : |\mathbf{b}(i,j)| > 2\bar{\nu} \tag{5.35}$$

Das Fenster der Faltung ist im Fall von Gleichung (5.35) keine Pixel-, sondern eine *Blocknachbarschaft* und wird auf eine Umgebung von $5 \times 5$ Blöcken festgelegt. $g$ ist die normierte, diskrete Gaußfunktion mit $\sigma = 1.5$. Abschließend kann die bewegungskompensierte zeitliche Glättung der Disparitätskarte $D$ durchgeführt werden:

$$\dot{D}_t(\mathbf{p}) = (1-\kappa) \dot{D}_{t-1}(\mathbf{p} - \mathbf{b}_\mathbf{p}) + \kappa D_t(\mathbf{p}) \tag{5.36}$$

Der Wichtungsfaktor $\kappa$ wird auf $2/3$ gesetzt. Der Bewegungsvektor $\mathbf{b_p}$ ist derjenige des Bewegungsblockes, unter dem sich der aktuelle Pixel $\mathbf{p}$ gerade befindet. Es ist ersichtlich, dass die Glättung rekursiv erfolgt, d. h., dass als vorherige Disparitätskarte $t-1$ die bereits geglättete verwendet wird.

### 5.2.7. Abschließende Filterung und binäre Segmentierung

Zur Beseitigung noch immer vorhandener Ausreißer kann abschließend ein Medianfilter angewendet werden. Ebenso kann eine binäre Maskierung durchgeführt werden. Der Median-Filter ermöglicht eine finale Kanten erhaltende örtliche Glättung der Disparitätskarten. Er wird mit einem Filterfenster der Größe $11 \times 11$ angewendet. Die Maskierung mit der binären Alphamaske entfernt Randartefakte, die möglicherweise durch die vorhergehenden Schritte erzeugt wurden und verhindert so ein „Ausfransen“ der Syntheseergebnisse aufgrund falscher Randdisparitäten.

### 5.2.8. Zusammenfassende Übersicht

Aus den in den vorherigen Abschnitten vorgestellten Algorithmen für Vorverarbeitung der Bilder, Berechnung und Nachverarbeitung der Disparitätskarten ergibt sich eine Vielzahl von möglichen Kombinationen. Die Wichtigsten davon sind in Tabelle 5.1 zusammengefasst.

## 5.3. Bildsyntheseverfahren

Im vorherigen Abschnitt 5.2 wurden etablierte, modifizierte und neue Verfahren zur Stereoanalyse vorgestellt. Die sekundäre Repräsentation der Verarbeitungskette liegt nun in Form rektifizierter Stereoaufnahme (Textur) und dazugehörigen rektifizierten Disparitätskarten vor (vgl. Abb. 3.2b und Tabelle 3.1). Dieser Abschnitt beschreibt die in dieser Arbeit entwickelten Verfahren zur Synthese derjenigen virtuellen Ansicht, die den Augenkontakt in der Videokommunikation wiederherstellen. Wie bereits in Kapitel 3 begründet, kommen als Basismethoden trifokaler Transfer und 3D-Warping zum Einsatz. Beide Vorgehensweisen werden aufbauend auf dem Grundlagenkapitel vorgestellt. Anschließend erfolgt die Beschreibung von in dieser Arbeit entwickelten Methoden zur Reduzierung prinzip- und algorithmenbedingter Fehler im Syntheseprozess.

| Var. | Vorv. | Algorithmus | | Kosten | | Aggr. | Nachverarbeitung | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | GF | LWTA | GMAP | ZNCC | Sobel | GF | KBF | ZG | BKZG | MF | BM |
| 05 | • | • | | • | | • | | | | | • |
| 05-01 | • | • | | • | | • | | | | • | • |
| 05-02 | • | • | | • | | • | | • | | • | • |
| 05-03 | • | • | | • | | • | • | | | | • |
| 05-04 | • | • | | • | | • | • | | | • | • |
| 05-05 | • | • | | • | | • | • | • | | • | • |
| 05-06a | • | • | | • | | • | • | | $8 \times 8$ | | • |
| 05-06b | • | • | | • | | • | • | | $16 \times 16$ | | • |
| 05-07a | • | • | | • | | • | • | | $8 \times 8$ | • | • |
| 05-07b | • | • | | • | | • | • | | $16 \times 16$ | • | • |
| 07 | • | | • | | • | | | | | | • |
| 07-01 | • | | • | | • | | • | | | | • |
| 07-02a | • | | • | | • | | • | | $8 \times 8$ | | • |
| 07-02b | • | | • | | • | | • | | $16 \times 16$ | | • |

GF - Gaußscher Filter (Vorfilter: $5 \times 5$ @ $\sigma = 0.5$, Aggregationsfilter $11 \times 11$ @ $\sigma = 4$
LWTA - lokale Korrespondenzsuche WTA
GMAP - globaler MAP Schätzer
ZNCC - Zero-mean Normalisierte Kreuzkorrelation
KBF - Konsistenz-basiertes Füllen mit $\xi = 5$
ZG - einfache zeitliche Glättung($\tau_l = 4$, $\tau_u = 10$)
BKZG - bewegungskompensiertes zeitliches Glätten
MF - Medianfilter $11 \times 11$
BM - binäre Maskierung

**Tabelle 5.1.:** Übersicht über verschiedene relevante Algorithmenkombinationen für die Bestimmung von Disparitätskarten.

### 5.3.1. Trifokaler Transfer

Im Grundlagenkapitel wurden bereits die Voraussetzungen für eine Bildsynthese mittels Trifokalem Transfer erläutert. Der trilineare Tensor zwischen drei Ansichten kann genutzt werden, um mittels der Basistrilinearitäten und korrespondierenden Pixelpositionen zweier Ansichten die jeweilige Position in der dritten Ansicht zu bestimmen (vgl. Gleichungen (2.39) und (2.40)). Die Korrespondenz wird durch die Disparitätskarte repräsentiert, deren Bestimmung im vorherigen Kapitel vorgestellt wurde. Es bleibt die Ermittlung des Tensors. Da ein kalibriertes Kamerasystem zum Einsatz kommt, sind externe und interne Kameraparameter bekannt. Der Tensor kann aus der kanonischen Form der Kameramatrizen bestimmt werden (siehe Gl. (2.38) (2.39)):

$$\mathtt{P} = [\mathtt{I}|\mathbf{0}] \qquad \mathtt{P}' = [a^i_j] \qquad \mathtt{P}'' = [b^i_j] \tag{5.37}$$

$$\mathcal{T}^{jk}_i = a^j_i b^k_4 - a^j_4 b^k_i \qquad \text{mit } i,j,k = 1,2,3 \tag{5.38}$$

Die Transformation der virtuellen Kamera $C''$ sei durch Translationsvektor $\mathbf{t}''$ und Rotationsmatrix $\mathtt{R}''$ definiert. Sie muss entsprechend der Position des Kommunikationsfensters sowie des Kommunikationspartners bestimmt werden. $\mathbf{t}'$ und Rotationsmatrix $\mathtt{R}'$ beschreiben entsprechend die Transformation von Kamera $C$ in Kamera $C'$, die das reale Stereopaar sind (vgl. Abschnitt 4.5). Unter der Kenntnis der internen Kameraparmeter lassen sich die Epipole aus der Projektion der jeweiligen Translationsvektoren bestimmen

$$\mathbf{e}' = \mathtt{K}'\mathbf{t}' \qquad \text{und} \qquad \mathbf{e}'' = \mathtt{K}''\mathbf{t}'' \tag{5.39}$$

Die o. g. kanonischen Formen der Projektionsmatrizen werden nun unter Verwendung der „unendlichen“ Homographien geschrieben (Gl. (2.34)), die sich aus der allgemeinen Disparitätsgleichung (Gl. (2.33)) herleiten:

$$\mathtt{H}'_\infty = \mathtt{K}'\mathtt{R}'\mathtt{K}^{-1} \qquad \text{und} \qquad \mathtt{H}''_\infty = \mathtt{K}''\mathtt{R}''\mathtt{K}^{-1} \tag{5.40}$$

Die Projektionsmatrizen der zweiten und der neuen Ansicht ergeben sich zu:

$$\mathtt{P}' = [\mathtt{H}'_\infty \mid \mathbf{e}'] \qquad \mathtt{P}'' = [\mathtt{H}''_\infty \mid \mathbf{e}''] \,. \tag{5.41}$$

Werden diese nun in Gleichung (5.38) eingesetzt, berechnet sich der Tensor nach

$$\mathcal{T}_i^{jk} = e'^j h_i''^k - e''^k h_i'^j \qquad \text{mit } i,j,k = 1,2,3 \,. \tag{5.42}$$

Unter der Nutzung der Basistrilinearitäten können aus Tensor und rektifizierten Disparitätskarten die neuen Pixelpositionen in der virtuellen Kamera berechnet werden. Im Weiteren wird dies ohne Einschränkung der Allgemeinheit für die linke Disparitätskarte $D_L$ demonstriert. Im Verlauf der Arbeit kam zunächst die in Abschnitt 5.1.2 vorgestellte Methode der projektiven Rektifizierung zum Einsatz. Die dabei verwendete lineare Transformation kann leicht invertiert und somit zur Derektifizierung genutzt werden. Im rektifizierten Fall sind die Intensitätswerte der Disparitätskarte ein direktes Maß für die horizontale Differenz zweier korrespondierender Bildpunkte. Durch die Derektifizierung erhält die Disparität nun zusätzlich eine vertikale Komponente. Die korrespondierenden Abbildungen eines Punktes $\mathbf{P}$ seien $\mathbf{p}$ in der linken und $\mathbf{p}'$ in der rechten Kamera. Sie ergeben sich durch Multiplikation der inversen rektifizierenden Homographiematrizen $\mathtt{H}^{-1}$ und $\mathtt{H}'^{-1}$ mit den korrespondierenden Bildpunktpositionen der rektifizierten Bilder $(x_{rekt}, y_{rekt}, 1)^\top$ und $(x_{rekt}, y_{rekt}, 1)^\top$:

$$\begin{pmatrix} \hat{x} \\ \hat{y} \\ \hat{w} \end{pmatrix} = \mathtt{H}^{-1} \begin{pmatrix} x_{rekt} \\ y_{rekt} \\ 1 \end{pmatrix} \,, \; \mathbf{p} = \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \hat{x}/\hat{w} \\ \hat{y}/\hat{w} \end{pmatrix} \text{ und} \tag{5.43}$$

$$\begin{pmatrix} \hat{x}' \\ \hat{y}' \\ \hat{w}' \end{pmatrix} = \mathtt{H}'^{-1} \begin{pmatrix} x_{\text{rekt}} - D_L(x_{\text{rekt}}, y_{\text{rekt}}) \\ y_{\text{rekt}} \\ 1 \end{pmatrix} \quad , \; \mathbf{p}' = \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \hat{x}'/\hat{w}' \\ \hat{y}'/\hat{w}' \end{pmatrix} \tag{5.44}$$

jeweils für alle Bildpunkte.

Zur Berechnung einer Bildpunktposition $\mathbf{p}'' = [x'', y'', 1]^T$ der dritten Ansicht, wird aus den vier linear unabhängigen Basistrilinearitäten (vgl. Gl. (2.41)) ein überbestimmtes lineares Gleichungssystem der Form $\mathbf{Ax} = \mathbf{b}$ aufgestellt. Dabei ist die Summenkonvention über $i = 1 \ldots 3$ zu beachten. Die Skalare $x''$ und $y''$ lassen sich aus den Gleichungen separieren und es ergibt sich das Gleichungssystem

$$\begin{pmatrix} \mathcal{T}_i^{13} p^i - x' \mathcal{T}_i^{33} p^i & 0 \\ 0 & \mathcal{T}_i^{13} p^i - x' \mathcal{T}_i^{33} p^i \\ \mathcal{T}_i^{23} p^i - y' \mathcal{T}_i^{33} p^i & 0 \\ 0 & \mathcal{T}_i^{23} p^i - y' \mathcal{T}_i^{33} p^i \end{pmatrix} \cdot \begin{pmatrix} x'' \\ y'' \end{pmatrix} = \begin{pmatrix} \mathcal{T}_i^{11} p^i - x' \mathcal{T}_i^{31} p^i \\ \mathcal{T}_i^{12} p^i - x' \mathcal{T}_i^{32} p^i \\ \mathcal{T}_i^{21} p^i - y' \mathcal{T}_i^{31} p^i \\ \mathcal{T}_i^{22} p^i - y' \mathcal{T}_i^{32} p^i \end{pmatrix} \tag{5.45}$$

Dieses kann anschließend z. B. mittels Methoden der Singulärwertzerlegung oder LR-Zerlegung gelöst werden. Die Pixelpositionen werden während der Berechnung durch reelle Zahlen repräsentiert[6]. Zum Setzen des Farbwertes der ganzzahligen Pixelposition $\mathbf{p}''$ in der virtuellen Ansicht werden sie nach dem trifokalen Transfer gerundet.

Der Einsatz der linearen Rektifizierung birgt Probleme, wie in Abschnitt 5.1.2 beschrieben und lieferte unbrauchbare Disparitätskarten. Daher wurde während der Arbeit entschieden, sie durch die nichtlineare Rektifizierung zu ersetzen. Die Bestimmung der Disparitätskarten aus nichtlinear rektifizierten Originalbildern erbrachte wesentlich bessere Ergebnisse. Jedoch ergibt sich aus dieser Vorgehensweise eine neue Problematik. Die Gleichungen (5.43), (5.44) der Derektifizierung müssen ersetzt werden, um die für den trifokalen Transfer notwendigen Korrespondenzen zwischen den originalen linken und rechten Ansichten zu bestimmen. Bei der Erstellung des ersten Testdatensatzes wurde hierbei bewusst eine systematische Ungenauigkeit akzeptiert, da die Implementierung des Trifokalen Transfers zu diesem Zeitpunkt nur die lineare Derektifizierung unterstützte. Um dennoch Korrespondenzen zwischen den originalen Ansichten zu erhalten, wurden die Disparitätskarten mittels der Matlab Camera Calibration Toolbox [Bouguet, 2010] nichtlinear derektifiziert. Die Vorgehensweise der nichtlinearen Derektifizierung wurde in Abschnitt 5.1.2 kurz erläutert. Die so erhaltenen derektifizierten Disparitätskarten kodieren nach wie vor nur eine horizontale Korrespondenz, obwohl die korrekte Disparität nun sowohl eine horizontale als auch eine vertikale Komponente enthält. Sie beinhaltet somit einen Fehler. Dieser Fehler ist *nicht* durch die Disparitätsanalyse sondern ausschließlich durch die beschriebene Vorgehensweise determiniert. Die Bestimmung der korrekten Korrespondenzen ist nur als

[6] In der Implementierung mittels *double*-Werten

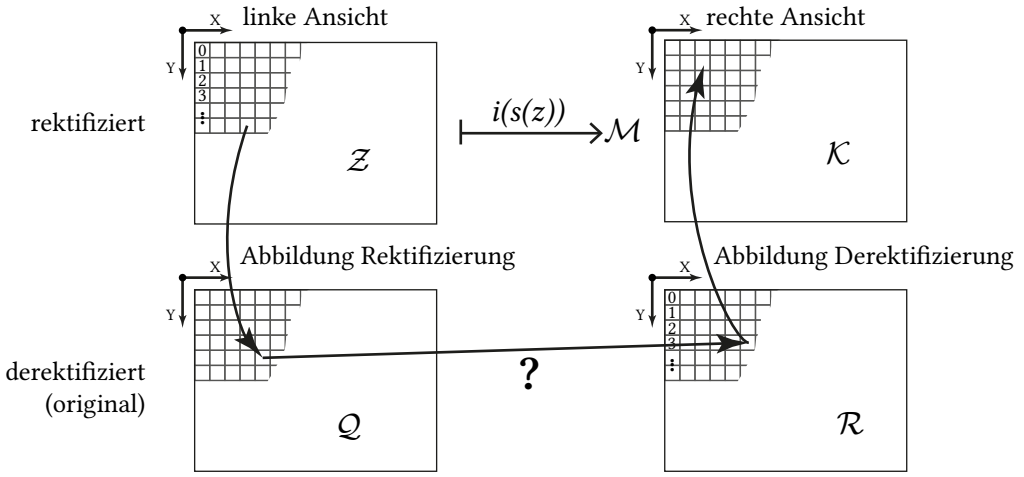



**Abbildung 5.13.:** Bestimmung von Korrespondenzen bei nichtlinearer Rektifizierung. Gesucht ist die Abbildung $\mathcal{Q} \longmapsto \mathcal{R}$ determiniert durch die Familie $(r_q)_{q \in \mathcal{Q}}$ mit $r_q \in R$. Da keine analytische Lösung existiert, muss diese durch eine Schnittmengenbildung bestimmt werden. Details siehe Text.

aufwändiger Suchprozess durch eine Schnittmengenbildung möglich und soll an dieser Stelle kurz erläutert werden.

**Exkurs nichtlineare Rektifizierung:** Die nichtlineare Rektifizierung für die linke Ansicht kann formal als Abbildung der Menge $\mathcal{Z}$ auf die Menge $\mathcal{Q}$ ausgedrückt werden:

$$\mathcal{Z} \longmapsto \mathcal{Q} \tag{5.46}$$

$\mathcal{Z}$ ist die Menge geordneter, gültiger Pixelindizes im rektifizierten Zielbild im geordneten Raster und hat die maximale[7] Mächtigkeit der Anzahl der Bildpixel $|\mathcal{Z}| \leq (M \times N)$. $\mathcal{Q}$ ist die Menge der Pixelindizes im Quellbild. Die Abbildung wird über die Familie $(q_z)_{z \in \mathcal{Z}}$ mit $q_z \in \mathcal{Q}$ definiert. $\mathcal{Z}$ stellt somit die Indexmenge der Familie $(q_z)$ dar. Ebenso lässt sich die Derektifizierung des rechten Bildes schreiben als:

$$\mathcal{R} \longmapsto \mathcal{K} \tag{5.47}$$

oder als Familie ausgedrückt: $(k_r)_{r \in \mathcal{R}}$ mit $k \in \mathcal{K}$ mit $\mathcal{R}$ als Indexmenge.

Die Umrechnung eines Pixels $\mathbf{p} = (x, y)$ in einen Pixelindex erfolgt allgemein mittels der Funktion:

$$j = i(\mathbf{p}) = xN + y, \quad \text{mit} \quad N \quad \text{als Anzahl der Bildzeilen} \tag{5.48}$$

und umgekehrt:

$$\mathbf{p} = s(j) = \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} j \div N \\ j \bmod N \end{pmatrix} \tag{5.49}$$

[7] Indizes, deren Abbildung Pixelpositionen außerhalb des Quellbildes sind, werden entfernt.

Gesucht ist nun eine Familie $(r_q)_{q\in\mathcal{Q}}$, die jedem Pixelindex $q \in \mathcal{Q}$ des linken originalen Bildes den korrespondierenden Pixelindex $r_q \in \mathcal{R}$ im rechten originalen Bild zuordnet. Dazu muss für jedes Element $z \in \mathcal{Z}$ die entsprechende Pixelposition mittels Gleichung (5.49) bestimmt werden:

$$\mathbf{p}_z = \begin{pmatrix} x_z \\ y_z \end{pmatrix} = s(z), \forall z \in \mathcal{Z} \tag{5.50}$$

Anschließend wird dessen korrespondierende Position im rechten, derektifizierten Bild aus der Disparitätskarte $D$ berechnet:

$$\hat{\mathbf{p}}_z = \begin{pmatrix} \hat{x}_z \\ \hat{y}_z \end{pmatrix} = \begin{pmatrix} x_z - D(x_z, y_z) \\ y_z \end{pmatrix}, \forall z \in \mathcal{Z} \tag{5.51}$$

Unter Verwendung von Gleichung (5.48) werden die korrespondierenden Pixel wieder in die Mitglieder der Familie $(m_z)_{z\in\mathcal{Z}}$ umgerechnet, deren Indexmenge nach wie vor $\mathcal{Z}$ ist:

$$m_z = i(\hat{\mathbf{p}}_z), \text{ mit } m_z \in \mathcal{M}, \ \forall z \in \mathcal{Z} \tag{5.52}$$

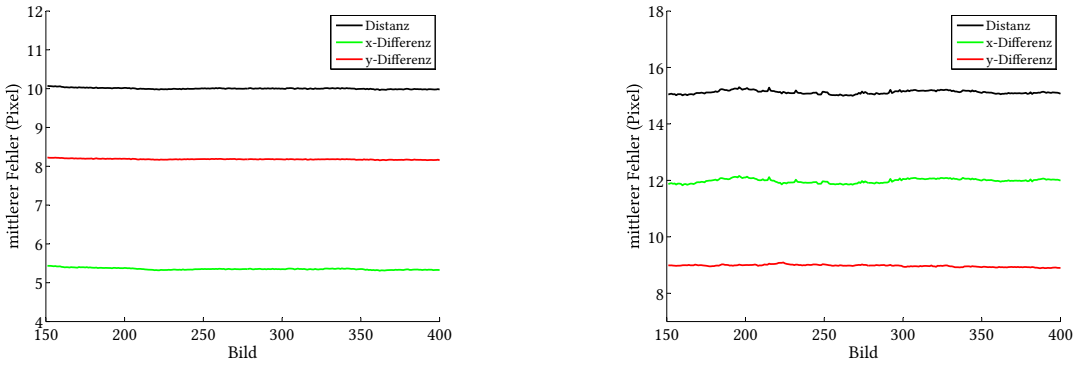
Die Mitglieder der gesuchten Familie $(r_q)$ können nun bestimmt werden mittels:

$$r_{q_z} = r \quad \text{wenn gilt} \quad m_z = k_r, \quad \forall z \in \mathcal{Z} \text{ und } \forall r \in \mathcal{R} \tag{5.53}$$

Dieser Ausdruck entspricht letztlich der Bildung einer Schnittmenge zwischen Mengen der Mitglieder der Familien $(m_z)$ und $(k_r)$. Mittels der geordneten Indexmengen $\mathcal{Z}$ und $\mathcal{R}$ lässt sich die jeweilige Beziehung zu den Koordinaten der Originalbilder herstellen. Abbildung 5.13 veranschaulicht den Sachverhalt. Die Bildung der Schnittmengen dieser verhältnismäßig großen Mengen ist mit erhöhtem Rechenaufwand verbunden.

Mittels dieser Berechnung wurde der systematische Fehler für jede der genutzten Sequenzen bestimmt. Beispielhaft ist dieser in Abb. 5.14 für zwei Sequenzen über eine Dauer von $10$ s dargestellt. Eine Visualisierung für zufällig ausgewählte Punkte zeigt Abb. 5.15.

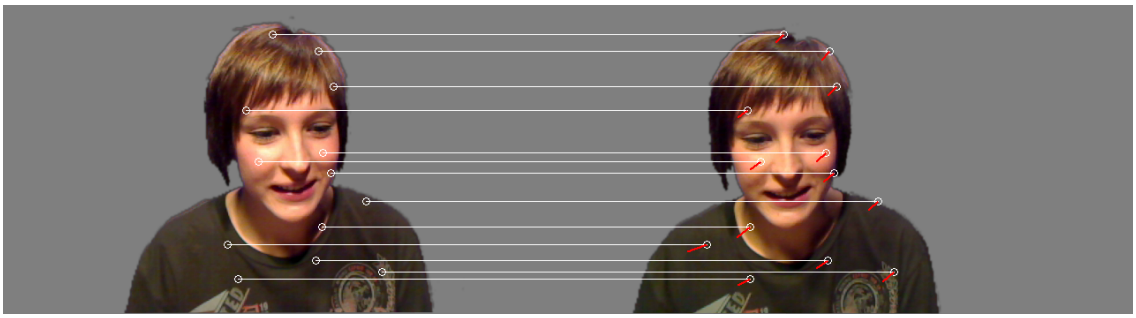
Der Distanz- oder Verschiebungsfehler liegt bei den untersuchten Sequenzen im Bereich von $9 \leq \epsilon \leq 16$ Pixel. Innerhalb eines Bildes ist die Varianz des Fehlers wesentlich geringer. Die Auswirkungen auf die Synthese bestehen *nicht* in Farbfehlern, wie durch die verschobenen Korrespondenzen evtl. vermutet werden könnte, da für die Synthese die Farbwerte der Pixel nur einer Originalkamera verwendet werden. Vielmehr besteht er in einer Verschiebung der Perspektive der virtuellen Ansicht, da die externe Transformation der Kameras nicht mehr mit deren abgebildeten Korrespondenzen überein stimmt. Da die Verschiebung innerhalb eines Bildes jedoch eine geringe Varianz aufweist, ist diese eher homogen über das gesamte Objekt und wird daher weniger stark wahrgenommen.




(a) (b)

**Abbildung 5.14.:** Die Graphen zeigen den mittleren Fehler, der durch die Derektifizierung der Disparitätskarten und anschließender Anwendung der derektifizierten Disparitäten zur Korrespondenzbestimmung entstanden ist. Angegeben sind euklidischer Abstand, sowie $x$- und $y$-Fehler im Vergleich zu den korrekten Werten in Pixeln. Distanz bzw. Richtungsfehler sind umso größer, je näher die Person am Kamerapaar war (in a weiter entfernt als in b), da so Basislinie und Linsenverzerrung eine größere Verschiebung bei der nichtlinearen (De-)rektifizierung erzeugen.

Nach diesem kurzen Exkurs zur nichtlinearen Rektifizierung soll nun wieder das weitere Vorgehen in der Synthese betrachtet werden. Durch die virtuelle Kameratransformation kann es vorkommen, dass zwei oder mehrere Pixel der Ausgangsansicht auf dieselbe Position in der virtuellen Ansicht transformiert werden (Verdeckung in der virtuellen Ansicht). Die Abarbeitung der Positionen erfolgt üblicherweise zeilenweise entsprechend des Bildkoordinatensystems. Daher ist nicht gewährleistet, dass im o. g. Fall die Farbe des Pixels gesetzt wird, dessen 3D-Punktentsprechung der virtuellen Kamera am nächsten ist (Verdeckungskompatibilität). Es kommt daher ein Tiefenpuffer zum Einsatz, der eine Überprüfung ermöglicht. Unter der Annahme, dass die virtuelle Kameratransformation nicht zu stark von der originalen abweicht, wird die



(a)

**Abbildung 5.15.:** Visualisierung des Fehlers bei der nichtlinearen Derektifizierung. Die weißen Linien zeigen zufällig ausgewählte verwendete Korrespondenzen. Die roten Vektoren sind die Verschiebungsvektoren zum korrekten Wert.

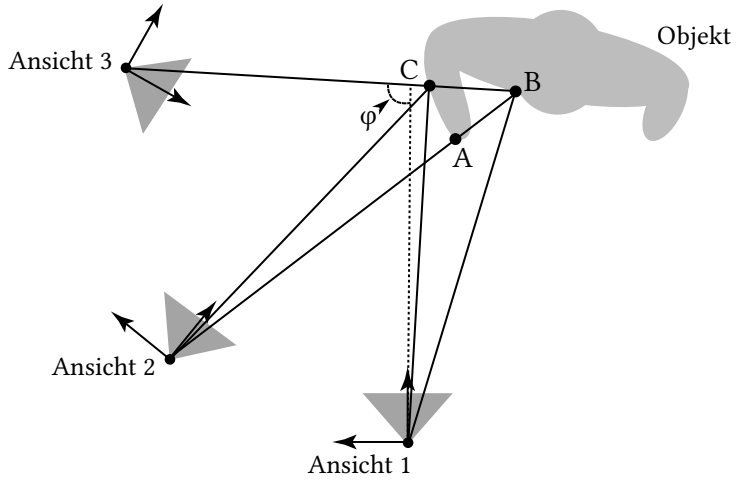



**Abbildung 5.16.:** Verdeckungsproblematik der Bildsynthese. Die Disparitäten in Ansicht 1 sind folgendermaßen geordnet: $d(\mathbf{A}) > d(\mathbf{B}) > d(\mathbf{C})$. In Ansicht 2 überschreibt die Abbildung von Punkt $\mathbf{A}$ die von Punkt $\mathbf{B}$ durch den Tiefenpuffer. in Ansicht 3 würde jedoch fälschlicherweise die Abbildung von Punkt $\mathbf{B}$ anstelle von Punkt $\mathbf{C}$ dargestellt. Nach Kreibich [Kreibich, 2005]

Disparität des Pixels im Ausgangsbild als Maß für die virtuelle Kamera genutzt. Bei Rotationen der Kamera um das Objekt, bei der der Winkel zwischen Sichtstrahl und optischer Achse der Referenzkamera $90°$ überschreitet, kann diese Annahme zu einzelnen Fehlsortierungen führen (vgl. Abb. 5.16). Für den Anwendungsfall der Videokommunikation ist eine solche Rotation jedoch ausschließbar. Der trifokale Transfer erfolgt mittels Vorwärts-Transformation. Da heißt, es werden ausgehend von der Referenzansicht die Pixelpositionen im Zielbild berechnet und gesetzt. Eine Möglichkeit der Rückwärtstransformation wurde von Avidan in [Avidan und Shashua, 1998] vorgestellt, soll jedoch an dieser Stelle nicht näher erläutert werden. Die hier dargestellte Vorgehensweise sowie ein Ansatz zur effizienten Implementierung auf der GPU wurden vom Autor in [Weigel und Kreibich, 2006] und [Weigel und Schübel, 2007] veröffentlicht.

Durch die Vorwärtstransformation kann es je nach Position der virtuellen Kamera zu synthesebedingten Löchern in der erzeugten Ansicht kommen (vgl. Abschnitt 3.3) Daher kommt zum Füllen dieser kleinen Löcher ein einfacher Interpolationsalgorithmus zum Einsatz. Abhängig von einer konfigurierbaren Pixelweite $w$ wird zunächst entschieden, ob ein vorliegender Pixel als Loch eingestuft wird. Hierzu wird getestet, ob der Pixel schwarz (Bildausgangsfarbe) ist und ob die darauf folgenden horizontalen Pixel im Bereich $w$ einen Wert ungleich Null annehmen. Ist dies der Fall, wird der Pixel als Loch angenommen und durch Mittelwertbildung aus dem darüber und links daneben liegenden Pixel gefüllt (bilineare Interpolation). Derselbe einfache Test erfolgt für die vertikalen Nachbarn. Durch diese Maßnahme können kleine Löcher bis zur Ausdehnung von $w$ Pixeln behoben werden. Üblicherweise sind dies Löcher, die durch geringe Annäherung bzw. Verschiebung der virtuellen Kamera und Rundungsfehler auf ganzzahlige Pixelpositionen

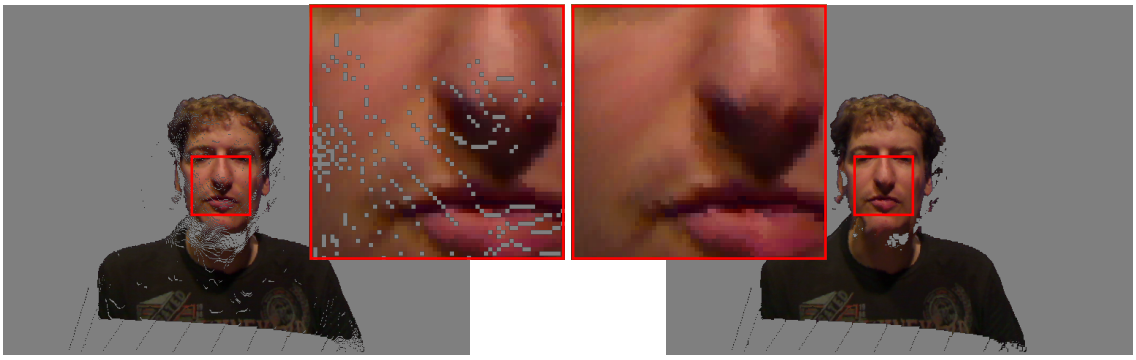


**Abbildung 5.17.:** Kleine, durch die Vorwärtstransformation nicht gesetzte Pixelbereiche werden nach dem trifokalen Transfer durch bilineare Filterung gefüllt. Große durch Aufdeckung verursachte Löcher (zu sehen z. B. im Halsbereich) bleiben bestehen.

entstanden sind. Große Löcher durch Aufdeckungen können mittels dieser Maßnahme nicht gefüllt werden (vgl. auch Abb. 5.17).

### 5.3.2. Vorschlag für ein punktbasiertes Verfahren

Die Auswahl des trifokalen Transfers für die Synthese schien im Verlauf der Arbeit zunächst ideal. Die virtuelle Ansicht kann direkt auf Pixelbasis erzeugt werden, singuläre Kameraanordnungen stellen kein Problem dar. Die Arbeitsschritte, die in den vorherigen Abschnitten beschrieben wurden, zeigen jedoch, dass das Verfahren auch Nachteile hat. So ist der Transfer durch die Auflösung der Gleichungssysteme für jeden Pixel rechnerisch aufwändig. Eine Parallelisierung ist nur teilweise realisierbar. Die Beschränkung auf das Setzen von stets genau einem Pixel bietet wenig Spielraum, durch Löcher verursachte Artefakte zu beseitigen. Daher wurde noch eine zweite Synthesemethode entwickelt, die auf dem Prinzip des 3D-Warpings beruht (vergleiche Abschnitt 2.3). In den folgenden Abschnitten wird das Verfahren kurz vorgestellt. Anschließen wird eine neuer in diesem Zusammenhang ebenfalls entwickelter Algorithmus präsentiert, der große, durch Verdeckungen erzeugte Löcher füllt.

Die nahezu parallele Anordnung der Kameras im vorliegenden Anwendungsszenario führte schnell zu der Idee, für die Synthese einen 3D-Warping-Ansatz zu wählen. Die abgebildeten Pixel einer der Referenzkameras werden dabei entsprechend ihrer Disparität in den 3D-Raum zurück projiziert. Es wird angenommen, dass ein Pixel die Abbildung $\mathbf{p} = (x, y)^\top$ eines 3D-Punktes $\mathbf{P} = (X, Y, Z)^\top$ repräsentiert.

Im achsenparallelen Fall kann dessen Tiefe $Z$ einfach durch die in Abschnitt 2.1.5 eingeführte Gleichung (2.27) berechnet werden:

$$Z = \frac{bf}{\delta_x} \cdot \frac{1}{d} \tag{5.54}$$

Die beiden Koordinaten $X$ und $Y$ des Punktes lassen sich anschließend durch die Umkehrung der intrinsischen Kameraabbildung berechnen, wobei von einer idealen Lochkamera mit Brennweite $f$ und Kamerahauptpunkt $(p_x, p_y)^\top$ ausgegangen wird (vgl. auch Gl. (2.12)):

$$X = \frac{fx + p_x}{Z} \text{ and } Y = \frac{fy + p_y}{Z} \tag{5.55}$$

Die Annahme einer idealen Lochkamera trifft nicht auf die real verwendeten Kameras zu. Auch kann nie vollkommene Parallelität in der Anordnung gewährleistet sein. Sie ist jedoch Voraussetzung für das o. g. einfache Vorgehen. Hingegen erzeugt die eingesetzte Methode der nichtlinearen Rektifizierung Abbildungen, die im Rahmen der Modell- und Kalibrierungsungenauigkeiten einer achsenparallelen Anordnung idealer Lochkameras entsprechen. Dabei wird auf minimale Verzerrung der Bilder und somit der abgebildeten Person geachtet (vgl. Abschnitt 5.1.2). Es wurde deshalb entschieden, die rektifizierten Bilder als Ausgangsbilder zu definieren. Diese werden zusammen mit der aus ihnen bestimmten Disparitätskarte direkt für das 3D-Warping benutzt.

Es wird demnach auch hier eine Vereinfachung vorgenommen, welche die Realität nicht mehr korrekt abbildet. Im Vergleich zur Vereinfachung beim trifokalen Transfer hat diese jedoch keine Implikation für die korrekte perspektivische Abbildung bei der Bildsynthese. Dies gilt unter der Bedingung, dass die rektifizierten Bilder als Aufnahme eines idealen Stereokamerapaares interpretierbar sind.

### 5.3.3. 3D-Warping

Die im vorherigen Abschnitt festgelegten Bedingungen lassen eine effiziente Implementierung der Gleichung (2.27) und (5.55) zu. Die aus den Pixeln der originalen (rektifizierten) Ansicht und der dazugehörigen Disparitätskarte berechneten 3D-Punkte können mittels einfacher perspektivischer Projektion entsprechend Gleichung (2.14) in die virtuelle Ansicht projiziert werden. Diese Projektion hat wiederum reelle 2D-Punktkoordinaten als Ergebnis, die anschließen in Pixel überführt werden müssen.

Der oben beschriebene Vorgang des 3D-Warpings ist dem Bereich der Computergrafik – konkret, dem Rendering – zuzuordnen. Für eine schnelle Berechnung solcher Abbildungen haben sich 3D-Softwareschnittstellen wie DirectX® von Microsoft® oder OpenGL® etabliert. Für die konkrete Umsetzung innerhalb dieser Arbeit wurde OpenGL® gewählt. Ohne auf spezifische Implementierungsdetails einzugehen, wird das Vorgehen im Folgenden beschreiben. Eine identische Umsetzung ließe sich auch mit anderen Schnittstellen realisieren. Es wurden zwei Varianten realisiert. Die erste, im Folgenden „CPU-Variante“ genannt, verwendet die klassische Grafik-Pipeline. Das heißt, es wurde auf Programmierung der GPU-Einheiten verzichtet. Die zweite Variante (im Folgenden: GPU-Variante) setzt die identische Funktionalität mittels GPU-Shader-Programmierung um. Diese Maßnahme erhöht nochmals die Geschwindigkeit der Bildsynthese.

Zudem gestaltet sie den Algorithmus flexibler hinsichtlich qualitätsverbessernder Maßnahmen beim Rendering.

Zunächst gilt es, die als Vertices bezeichnetet 3D-Punkte zu berechnen. In beiden Varianten geschieht dies nach Gleichung (2.27) und (5.55). Die CPU-Variante führt die Berechnung jedoch sequentiell durch. D. h. für jeden der $M \times N$ Pixel des Bildes wird die Berechnung nacheinander ausgeführt. Die GPU-Variante nutzt den Vertex-Shader. Dadurch kann eine massive Parallelisierung erreicht werden. Da der Vorgang für die Videoechtzeit $25$ mal pro Sekunde durchgeführt werden muss, ist der Geschwindigkeitsvorteil enorm. Liegen die Vertices auf der Grafikkarte vor, werden sie anschließend direkt als Punkte gerendert. D. h. im Gegensatz zum sonst üblichen Vorgehen in der Computergrafik wird *kein* Drahtgittermodell (*Mesh*) erstellt (vgl. auch Verfahren mit expliziter Geometrie in Abschnitt 2.3). Dies hat zwei Vorteile. Zum einen ist die Erzeugung eines Meshes aus einer Punktwolke nicht trivial. Es existieren diverse Algorithmen zu diesem Zweck, die unterschiedlich aufwändig sind und unterschiedliche Ergebnisse liefern. Oft entstehen Probleme an Tiefensprüngen, die ohne zusätzliche Kamerainformationen zu degenerierten Polygonen im Mesh führen können. Daher ist diese Methode eher für Multi-Kamerasysteme geeignet. Weiterhin hat das Echtzeitrendering von Meshes den Nachteil, dass die endgültige Farbe der Pixel der abgebildeten Oberflächen mittels einfacher Algorithmen (z. B. Gouraud-Shading) berechnet wird. Auch wenn inzwischen bessere Algorithmen existieren, so wirken solche Flächen dennoch oft unnatürlich oder künstlich. Nach Meinung des Autors ist dies besonders bei der Darstellung von Menschen problematisch. Im Anwendungsfall der Videokommunikation wird die virtuelle Kamera nicht näher als die Referenzkameras am Objekt platziert. Daher ist auch kein Nachteil durch fehlenden Bildinformationen durch das Punktrendering zu erwarten (vgl. Abschnitt 3.3).

Die Erzeugung des Bildes der virtuellen Kamera wird durch die (programmierbare) Grafikpipeline vorgenommen. Zunächst wird die Position des 2D-Punktes in der Bildebene durch die einfache projektive Abbildungsgleichung ((2.14)) bestimmt. Welche Pixel im Bild anschließend entsprechend gefärbt werden, kann dabei durch verschiedene Methoden und Parameter beeinflusst werden:

**Tiefenabhängige Punktgröße:** OpenGL® definiert eine Gleichung, mit der die Größe des gerasterten Punktes (und damit die Anzahl der Pixel, denen die entsprechend Farbe gesetzt wird) berechnet wird:

$$s_{punkt} = s\sqrt{\frac{1}{a + bZ + cZ^2}} \tag{5.56}$$

Die modifizierbaren Parameter $a$, $b$ und $c$ sind dabei Abschwächungsparameter, $s$ die verwendete Ausgangspunktgröße. Variable $Z$ ist die orthogonale Distanz der virtuellen Kamera zum aktuell betrachteten 3D-Punkt. Die Gleichung beschreibt eine nichtlineare Vergrößerung/Verkleinerung der Punkte mit abnehmender/zunehmender Distanz zur Kamera. Abbildung 5.18 zeigt die Punktgröße für zwei in der Arbeit verwendete Ausgangsgrößen $s$. Welche Pixel schlussendlich gesetzt

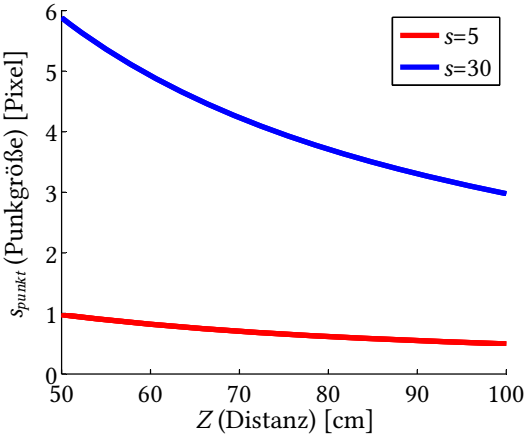



**Abbildung 5.18.:** Distanzabhängigkeit der Punktgröße bei 3D-Warping für anwendungsrelevante Kameradistanzen ($a = 0.5, b = 0.01, c = 0.01$).

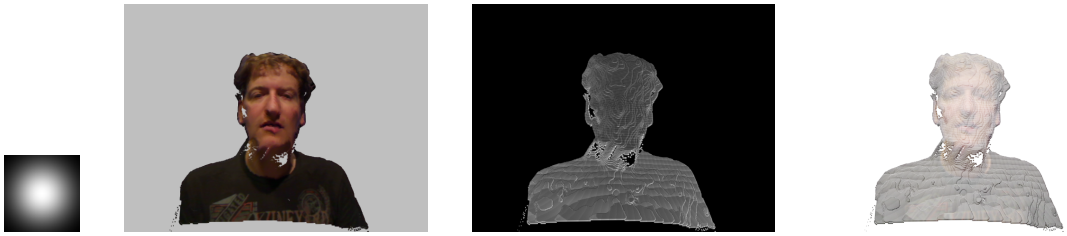
werden, hängt von der gewählten Punktform und den folgenden Verarbeitungsschritten ab.

Die Punkte werden von der Grafik-Pipeline in der einfachsten Form als Quadrate dargestellt. Die Pixel werden entsprechend der Größe des nach Gleichung (5.56) berechneten Quadrates gesetzt. Fällt die Quadratbegrenzung zwischen zwei Pixel, wird auf die nächste ganzzahlige Pixelposition gerundet. Wie bereits in Abschnitt 3.3 dargelegt, können durch die Verwendung von Quadraten Kanten im Bild entstehen. Je größer die Punkte sind, desto problematischer kann ein Intensitäts- oder Farbunterschied zwischen den Kanten werden. Hohe örtliche Frequenzen und somit als störend empfundene Artefakte sind möglich. Andererseits eignen sich größere Punkte besser dazu, ungewollte Löcher – z. B. durch Fehlplatzierungen aufgrund falscher Disparitätswerte – zu füllen. Es muss somit ein Kompromiss gefunden bzw. versucht werden, den Aliasing-Effekt zu minimieren und dennoch kleine Löcher zu vermeiden.

**Geglättete Punkte:** Einen einfachen Ansatz bietet die Grafik-Pipeline selbst. Mittels eines Glättungsparameters können anstelle von Quadraten Kreise gezeichnet werden. Die Realisierung der abgerundeten Kanten erfolgt dabei mittels so genannten Alpha-Blendings – einer gewichteten Interpolation sich überlagernder Bereiche. Ein exemplarischer Test des Verfahrens ergab, dass die gewünschte Glättung bei starker Annäherung an das Objekt durchaus hilfreich sein kann. Bei einer Kameradistanz ähnlich der Distanz der Referenzkameras erzeugt die Verwendung kreisförmiger Punkte jedoch stärkere Artefakte. Abbildung 5.19 zeigt dies an einem Beispiel. Eine Ursache sieht der Autor in der geringen Größe der Punkte bei dieser Distanz. Ein Kreis deckt dabei nur einen geringen Bereich des Bildes ab und es werden nur wenige Pixel gesetzt. Es entsteht eher eine unregelmäßige rechteckige Struktur. Diese Unregelmäßigkeit führt zu stärkeren Unruhen als die Darstellung von regelmäßigen Quadraten. Eine andere Methode ergibt sich durch die Verwendung von texturierten Punkten *point sprites*. Hier kann auf das

(a) (b) (c)

**Abbildung 5.19.:** Ungeeignete Glättung der Punkte beim 3D-Warping. (a,b) Nutzung von Kreisen als Primitive. (c) Nutzung von Quadraten als Primitive. Zwecks der alleinigen Veranschaulichung des Effekts entspricht die virtuelle Kameraposition exakt der Position der linken Kamera.



(a) (b) (c) (d)

**Abbildung 5.20.:** Glättung durch texturierte Punkte beim 3D-Warping. (a) Vergrößerte, zur Glättung verwendende Textur (Gaußfilter). (b) RGB-Kanal einer synthetisierten Ansicht. (c) Fehlerhafte Alphamaske. (d) Ansicht mittels fehlerhafter Alphamaske mit weißem Hintergrund überblendet.

Quadrat eine beliebige Alpha-Textur aufgebracht werden (vergleiche Alphamaske in Abschnitt 5.1.1). Somit lässt sich die Überblendung zwischen überlappenden Punkten flexibel steuern. Auch diese Methode wurde implementiert und wirkte zunächst vielversprechend. Durch die Verwendung einer Textur auf Basis einer Gaußfunktion (vgl. Abb. 5.20a) lassen sich Artefakte vermeiden. Jedoch ergibt sich hier erneut ein Effekt, der im Zusammenhang mit den folgenden Füllalgorithmen zu Problemen führt. Die Alphamaske des synthetisierten Bildes wird durch die Textur ebenfalls überblendet und enthält somit Alpha-Werte, die nicht benutzbar sind. Abbildung 5.20 veranschaulicht die Problematik. Die Möglichkeit einer Korrektur der Alphamaske mittels Schwellwertbildung wurde während der Arbeit in Betracht gezogen. Jedoch wurde dieses Verfahren aufgrund der mangelnder Zeit zum ausführlichen Testen und unklaren Auswirkungen auf die folgenden Verarbeitungsschritte nicht in den Testdatensatz mit aufgenommen. Der Autor verspricht sich dennoch einen klaren Gewinn in der Fehlerbeseitigung durch diese Methode, weshalb sie hier genannt wird und in weiterführenden Arbeiten bedacht werden sollte.

(a) (b)

**Abbildung 5.21.:** (a) Korrekte Tiefensortierung durch Aktivierung des Tiefenpuffers der Grafikkarte. (b) Bei deaktiviertem Tiefenpuffer können je nach Position der virtuellen Kamera Artefakte wie im Bild entstehen.

**Einfluss der Tiefensortierung beim Rendering:** Die Abbildungen der 3D-Punkte, die von der Grafikkarte erzeugt werden, können sich je nach Position der virtuellen Kamera überlappen (vgl. Verdeckungskompatibilität in Abschnitt 5.3.1). Für die korrekte Darstellung muss daher ein Tiefentest erfolgen. Bei der Darstellung mittels Grafikkarte ist dieser Tiefentest Bestandteil der Grafikpipeline und muss einfach aktiviert werden. Geschieht dies nicht, werden die Punkte sequenziell gerendert und es kann zu fehlerhaften Verdeckungen kommen, wie Abb. 5.21 zeigt. Hinsichtlich der Tiefenpufferung ergaben sich während der Arbeit zwei Problemstellungen.

Es wurde beobachtet, dass die Aliasing-Artefakte – insbesondere bei größeren Punktgrößen – durch das Tiefenrendering verstärkt werden. Die subjektiven Unterschiede im Vergleich zum sequentiellen Rendering sind bei fast allen Sequenzen deutlich wahrnehmbar (vgl. Abb. 5.22). Eine analytische Beschreibung des Effektes konnte während der Arbeit nicht gefunden werden. Jedoch lässt sich vermuten, dass das Setzen der Pixel anhand der Tiefe ein pseudozufälliges Muster erzeugt. Dessen Artefakte sind weitaus stärker wahrnehmbar als beim nacheinander Setzen ursprünglich benachbarter Pixel. Hinsichtlich der Anwendung ist abzuwägen, ob die Tiefenpufferung notwendig ist. Mit zunehmender Rotation der virtuellen Kamera um das Objekt ist die Auswirkung gravierender. Die Entscheidung ist demnach abhängig von der Lage des Stereopaares sowie der Blickrichtung der Person. Da Versuche mit einigen der Testsequenzen wenige bis keine Artefakte durch falsche Tiefensortierung zeigten, wurde auch diese Variante in den Testdatensatz aufgenommen.

Die zuvor beschriebenen Maßnahmen wie die Verwendung geglätteter Punkte oder texturierter Punkte können den negativen Effekt durch die Tiefenpufferung wie beschrieben verringern, führen jedoch zu der zweiten Problematik eines Tiefenpuffers. Diese ergibt sich im Zusammenhang mit dem notwendigen Alpha-Blending. Der Tiefenpuffer basiert auf dem Prinzip der Ersetzung von bereits gesetzten durch davor liegende Pixel. Die ersetzten Werte sind somit verloren. Wenn jedoch davor liegende Punkte eine Transparenz aufweisen, werden zum korrekten Überblenden die Informationen der hinteren Pixel noch benötigt. Insbesondere wenn jeder Punkt Alpha-Blending verwendet, ergibt das unbenutzbare Bilder. Ein Ausweg ist, die

(a) (b) (c) (d) (e)

**Abbildung 5.22.:** (b) Nahansicht des Bereiches in (a) mit eingeschalteter Tiefensortierung und Verwendung von Quadraten als Primitiven. (c) Verwendung von Quadraten bei abgeschalteter Tiefensortierung. (d,e) Wie (b) und (c), jedoch mittels texturierter Punkte geglättet. Zwecks der alleinigen Veranschaulichung des Effekts entspricht die virtuelle Kameraposition exakt der Position der linken Kamera.

Punkte entsprechend ihrer Tiefe vorab auf der CPU zu sortieren und die Tiefenpufferung der Grafikkarten zu deaktivieren. Somit wird wieder Verdeckungskompatibilität erreicht. Jedoch bringt dies starke Performanceeinbußen hinsichtlich der Berechnungsgeschwindigkeit mit sich. Die Problematik der stärkeren Artefakte durch korrektes Tiefenrendering ist natürlich auch bei dieser Vorgehensweise vorhanden.

**Anti-Aliasing:** Ein für das Bild ganzheitlicher Ansatz der Artefaktvermeidung wird von Grafikkarten durch verschiedene Varianten des Vollbild-Anti-Aliasing *Full Screen Anti-Aliasing – FSAA* realisiert. Die Verfahren arbeiten nicht mehr für jeden abgebildeten Punkt, sondern auf dem kompletten Bild. Grundsätzliche Verfahren lassen sich dabei in *Super-Sampled Anti-Aliasing – SSAA*, *Multi-Sampled Anti-Aliasing – MSAA* und *Coverage Sampled Anti-Aliasing – CSAA* unterscheiden. Die Grundidee ist allen gemein. Für jeden final gesetzten Pixel werden zusätzliche $n$ Pixel (Samples) durch das Rendering eines oder mehrerer größerer Bilder erzeugt. Anschließend werden diese zum Zielpixel mittels eines Filters herunter interpoliert. Welches Verfahren welche Grafikkarte letztendlich nutzt, ist Modell- und treiberspezifisch und über OpenGL® schwer kontrollierbar bzw. nicht vorgegeben (vgl. [Segal und Akeley, 2010]). Da die Wirkung wichtig ist, die Unterschiede hinsichtlich der Anwendung jedoch nicht gravierend sind, wird hier ausschließlich die Anzahl $n$ der Samples als Parameter betrachtet. Moderne Grafikkarten implementieren bis zu $16\times$ FSAA, was auch beim 3D-Warping zur Anwendung kommt.

### 5.3.4. Konturbasiertes Füllen

Die im vorhergehen Abschnitt entwickelten und ausgewählten Algorithmen und deren Kombination zielen auf primär zwei Fehlertypen ab: dem Füllen kleiner Löcher und der Behebung von Aliasing, welches u. a. durch diese Fülltechniken entstehen kann. Größere Löcher, die durch

Aufdeckungen entstehen, werden nicht behandelt. Nach Fertigstellung des ersten Testdatensatzes und der Durchführung des ersten Experiments mit den erzeugten Testdaten wurden von vielen Probanden eben jene Fehlertypen als Ursache für eine geringe Qualitätseinschätzung genannt (vgl. 6.3.3).

Daher wurde nach der Konzeption und Implementierung der grundlegenden Synthesealgorithmen Hauptaugenmerk auf das Füllen eben dieser Löcher in einem Nachverarbeitungsschritt gelegt. Zum Füllen von großen Löchern muss bei den meisten Algorithmen die zu füllende Region zunächst spezifiziert werden. Dies gestaltet sich besonders bei „offenen“ Löchern im Randbereich schwierig. Um dennoch für große Bereiche eine klare Abgrenzung zu erhalten, wurde eine Methode entwickelt, die die Kontur der Person zu Hilfe nimmt.

Dabei wird zunächst die Kontur in einer der originalen (rektifizierten) Ansichten $I_L$ oder $I_R$ aus der binären Alphamaske ermittelt. Die folgenden Ausführungen sind ohne Einschränkung der Allgemeinheit für beide Ansichten durchführbar. Die Punkte des sie repräsentierenden Polygons sind die Pixelpositionen des Randes der Alphamaske. Dieses dichte Polygon wird nun mittels des Douglas-Peucker Algorithmus (vgl. [Douglas und Peucker, 1973] ) so vereinfacht, dass sich die Punktezahl verringert, die Form jedoch weitestgehend erhalten bleibt. Die Parameter wurden dabei empirisch ermittelt, wobei darauf geachtet wurde, dass stets eine hohe Anzahl an Punkten im Polygon erhalten bleibt. Subjektiv visuell waren die Abweichungen von der originalen Kontur bei keiner der Sequenzen wahrnehmbar.

Für jeden der Punkte des Polygons wird anschließend eine Disparitätsanalyse durchgeführt. Um den Aufwand für diesen Verarbeitungsschritt gering zu halten, wird auf die bereits bei der dichten Disparitätsanalyse verwendete Methode von Geiger in [Geiger u. a., 2010] zurückgegriffen (vgl. Abschnitt 5.2.3). Jedoch wurde auf die globale MAP-Optimierung verzichtet. Für den hier beschriebenen Zweck reicht der Vorverarbeitungsschritt zur Bestimmung robuster spärlich verteilter Korrespondenzen auf Basis von Sobel-Filter-Antworten aus. Aus den gefundenen Korrespondenzen werden nicht robuste entfernt. Dies geschieht mittels einer Konsistenzprüfung im Sinne von Gleichung (5.30) mit $\xi = 2$ sowie auf Basis des Verhältnisses zwischen den geringsten Kosten an Punkt $\mathbf{p_1}$ des *best matches* $C(x_1, y_1)$ und den zweitgeringsten Kosten $C(x_2, y_2)$ an Punkt $\mathbf{p_2}$:

$$robust(\mathbf{p_1}) = \begin{cases} 1 & C(x_1, y_1) < \zeta C(x_2, y_2) \\ 0 & \text{, sonst} \end{cases} \tag{5.57}$$

$\zeta$ wurde in den Experimenten auf $0.6$ gesetzt. Die robusten Punkte und somit die Kontur werden nun mittels 3D-Warping (Gl. (5.54),(5.55) und (2.18) - (2.20)) in die virtuelle Ansicht transformiert. Eine Umsetzung mittels des trifokalen Transfers fand nicht statt, da dieses Verfahren explizit für die Synthese mittels 3D-Warping entwickelt wurde. Die transformierte Kontur wird nun einfach

(a) (b) (c) (d)

**Abbildung 5.23.:** Finden von Löchern beim konturbasierten Füllen. (a) Maske der originalen, rektifizierten linken Ansicht. Grüne Punkte sind robust, rote Ausreißer. (b) Die transformierte Kontur $M_c$. Obwohl auch in der Kontur einige Fehlplatzierungen vorhanden sind, ist sie gut als Basis zur Identifizierung von Löchern geeignet. (c) Die Maske $M_v$, aus der virtuellen Ansicht erstellt. (d) Die Kombination von (b) und (c) ergibt die Lochkarte $M_h$.

gefüllt, so dass sich das Maskenbild[8] $M_c$ ergibt. Die (löchrige) Kontur, die sich aus den Pixeln der synthetisierten Ansicht ergibt, wird mit $M_v$ bezeichnet. Sie ergibt sich, indem jeder beim 3D-Warping erzeugte Pixel in der Maske auf 1 gesetzt wird. Die zu füllende Lochmaske ergibt sich nun durch folgende Rechenschritte. Anhand von $M_c$ werden zunächst alle Ausreißer in $M_v$ entfernt. D. h. falsch transformierte Pixel werden eliminiert:

$$M_{v'}(\mathbf{p}) = \begin{cases} 0 & \text{, wenn } M_c(\mathbf{p}) = 0 \\ M_v(\mathbf{p}) & \text{, sonst} \end{cases} \tag{5.58}$$

Die binäre Lochkarte $M_h$ ergibt sich anschließend zu:

$$M_h = M_{v'}(\mathbf{p}) \oplus M_c(\mathbf{p}) \tag{5.59}$$

wobei $\oplus$ eine exklusiv-oder Operation ist. Der gesamte Vorgang ist in Abb. 5.23 dargestellt. Anhand der Lochkarte wird wiederum der Algorithmus von Telea eingesetzt [Telea, 2004], um die Löcher in der synthetisierten Ansicht zu füllen (vgl. Abschnitt 5.2.5, Gl. (5.31)). Wie bereits bei der „Ausbesserung“ der Disparitätskarte bietet der Algorithmus ein gutes Verhältnis zwischen Qualität und Berechnungszeit. Er wurde vom Autor in [Weigel und Treutner, 2012] vorgestellt.

### 5.3.5. Vertikale Anpassung

Um die Plausibilität der Ansicht weiter zu erhöhen, wird eine einfache und dennoch wirkungsvolle Maßnahme durchgeführt. In den Interviews der ersten Studie wurde häufig das „Schweben“ der Person bemängelt. Durch die Translation der virtuellen Kamera nach unten und deren

[8] im Folgenden einfach „Maske“

leichter Rotation nach oben, werden bei der Bildsynthese ursprünglich nicht sichtbare Bildbereiche unterhalb der Person aufgedeckt. Da diese nicht innerhalb oder am seitlichen Rand der Kontur der Person liegen, wirken diese zwar nicht störend als Löcher. Jedoch ist der Kontakt der Kontur mit der unteren Bildkante nicht mehr gegeben, welcher jedoch üblicherweise bei Videokommunikationssituationen existiert.

Daher wird eine vertikale Verschiebung der gesamten virtuellen Ansicht durchgeführt, so dass der untere Bildrand einen bestimmten Bereich der Person abdeckt. Für die Testdatensets lässt sich die Anzahl der Zeilen für die Verschiebung einfach über die Anzahl der Pixel innerhalb der Kontur in jeder Zeile bestimmen.

### 5.3.6. Zusammenfassende Übersicht

Die in den vorherigen Abschnitten vorgestellten Algorithmen zur Erzeugung der virtuellen Ansicht sowie anschließend durchgeführte qualitätsverbessernde Maßnahmen zeigt Tabelle 5.1.

| Variante | Synthese aus | | Algorithmus | | 3DW-Rendering | | | | Nachverarbeitung | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | DRL | RL | TFT | 3DW | QPF | APG | TS | MS | FBL | FKB | MF | VA |
| 06 | • | | • | | | | | | • | | | |
| 07 | • | | • | | | | | | • | | • | |
| 08 | • | | • | | | | | | | | • | |
| 09 | | • | | • | • | 5 | • | | • | | | • |
| 10 | | • | | • | • | 5 | • | | | • | | • |
| 11 | | • | | • | • | 25 | | | • | | | • |
| 12 | | • | | • | • | 25 | | | | • | | • |
| 15 | | • | | • | • | 30 | • | • | • | | | • |
| 16 | | • | | • | • | 30 | • | • | | • | | • |
| 19 | | • | | • | • | 30 | | • | • | | | • |
| 20 | | • | | • | • | 30 | | • | | • | | • |

DRL Derektifizierte linke Ansicht & Derektifizierte Disparitätskarte
RL Rektifizierte linke Ansicht & Rektifizierte Disparitätskarte
TFT Trifokaler Transfer
3DW 3D-Warping
QPF Quadratische Punktform
APG 3D-Warping: Ausgangspunktgröße (vgl. Gl. (5.56))
TS 3D-Warping: Tiefensortierung
MS 3D-Warping: Multisampling 16x
FBL Füllen von Löchern: Nachbar bilinear
FKB Füllen von Löchern: Konturbasiert, Telea
MF Glättung: Medianfilter 3x3
VA Vertikale Anpassung

**Tabelle 5.2.:** Übersicht über verschiedene, relevante Algorithmenkombinationen für die Synthese blickkorrigierter Ansichten.

# 6. Evaluation, Ergebnisse und Vergleich

Die im vorangegangen Kapitel entwickelten Algorithmen zur Erzeugung einer blickrichtungskorrigierenden virtuellen Ansicht von Videokommunikationspartnern werden in diesem Abschnitt evaluiert. Viele Entscheidungen hinsichtlich Parameter- und Methodenauswahl konnten bereits anhand von Erfahrungen, technischen Einschränkungen und Literaturrecherchen getroffen werden. Sie wurden an den entsprechende Stellen bereits erläutert und begründet. Wie Kapitel 5 zeigt, stehen am Ende dennoch eine Vielzahl von Möglichkeiten, das gesetzte Ziel zu erreichen. Die erreichbare Bildqualität der Synthese als Ergebnis der Verarbeitungskette ist letztlich das ausschlaggebende Kriterium für den Erfolg der durchgeführten Arbeiten. Deren Qualitätsbeurteilung ist daher Hauptbestandteil dieses Kapitels. Die Arbeiten hierzu wurden ebenfalls im Projektrahmen dieser Dissertation teilweise von einer Kollegin des Autors durchgeführt [Kepplinger, vorr. 2014]. Die Entwicklung der verwendeten Methodik und deren Durchführung sind Gegenstand einer in Arbeit befindlichen zweiten Dissertation und werden entsprechend referenziert.

Im ersten Abschnitt wird kurz die Problematik der Verwendung etablierter 2D-Videoqualitätsmaße für den Anwendungsfall dieser Arbeit erläutert. Anschließend erfolgt die Beschreibung der Vorselektion von Testmaterialien auf Grund subjektiver Einschätzung des Autors. Die Ergebnisse zweier im Rahmen des Forschungsprojektes durchgeführter subjektiver Qualitätsexperimente werden in den folgenden Abschnitten präsentiert. Darauf aufbauend erfolgt ein qualitativer Vergleich mit dem Stand der Technik gefolgt von einer kritischen Betrachtung der vorliegenden Arbeit.

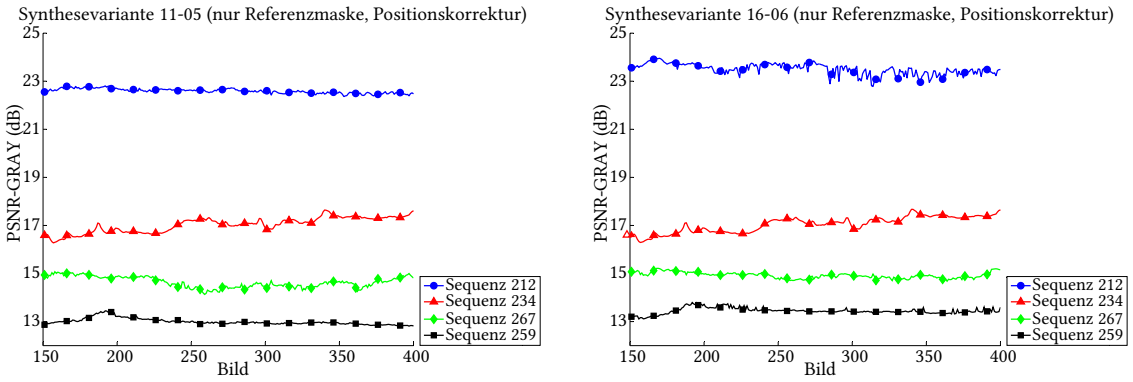
## 6.1. Verwendung von Metriken zur Qualitätsbeurteilung

Für eine objektive Beurteilung der Bildqualität von digitalen Bildern und Videos kommen im Medienbereich verschiedene Metriken zum Einsatz. Diese ermitteln – meist basierend auf einer als ideal angenommen Referenz – anhand einer Berechnungsvorschrift Werte, die eine Aussage über die Qualität des Bildes oder Videos zulässt. Dabei erheben diese Maße oft den Anspruch, der menschlichen Wahrnehmung der Qualität zu entsprechen. Algorithmen und Skalen werden durch eine Vielzahl von subjektiven Tests hinsichtlich dieser Eigenschaft angepasst und optimiert. Ein nahe liegendes und häufig verwendetes Maß im Bereich der Videoqualitätsbeurteilung (insbesondere bei der Videokodierung) ist die euklidische Distanz, beschrieben als mittlere

quadratische Abweichung (*Mean Squared Error* – MSE) und das sich daraus ergebende Spitzen-Signal-Rausch-Verhältnis (*Peak Signal to Noise Ratio* – PSNR). Mitunter wird zur Vereinfachung auch die Absolute Abweichung (*Mean Absolute Difference* – MAD oder MSAD) verwendet. Neuere, komplexere Maße korrelieren besser mit der subjektiven Wahrnehmung. So wurden z. B. von der ITU-VQEG (*Video Quality Experts Group*) Standards wie J.247 (PEVQ – *Perceptual evaluation of video quality*) [ITU-T, 2008] oder für HD-Video J.341 (VQuad-HD) [ITU-T, 2011] etabliert. Nicht standardisiert, aber dennoch häufig verwendet wird SSIM (*Structural Similarity*) [Wang u. a., 2004] und dessen Erweiterungen (hierarchisch [Wang u. a., 2003], regionenbasiert [Li und Bovik, 2010] und Integration von Bewegungsanalyse [Moorthy und Bovik, 2010]). Liegt keine Referenz vor, kommen meist merkmalsbasierte Maße zum Einsatz. Es wird versucht, bestimmte Eigenschaften, wie Schärfe, „Blockigkeit“ oder Rauschen zu messen. Details dazu finden sich in [Rittermann, 2007]. Es existieren noch viele weitere Verfahren, welche an dieser Stelle jedoch nicht Gegenstand der Betrachtung sein sollen.

Hinsichtlich der virtuellen Bildsynthese im Allgemeinen und der Verarbeitungskette für die Blickkorrektur im Speziellen und ergeben sich bei den oben genannten Metriken zwei grundsätzliche Probleme: Für eine beliebig platzierbare virtuelle Ansicht existiert üblicherweise keine Referenz und die qualitätsmindernden Fehlerausprägungen wie in Abschnitt 3.3 beschrieben, sind oft nicht hinreichend gut durch etablierte Maße aus dem 2D-Bereich messbar. Letzteres Problem wurde vor allem in der Dissertation von Rittermann mit der Entwicklung des 3DVQM (3D-Videoobjektqualitätsmaß) angegangen [Rittermann, 2007]. Neue Fehlertypen wurden identifiziert und Metriken zu deren Messung entwickelt. Dennoch kommen auch diese Methoden letztlich nicht ohne Referenz aus. Auch haben sich seither die Algorithmen verändert und eine Anpassung der Metriken sowie die Adaption an die subjektive menschliche Wahrnehmung müsste erneut durchgeführt werden.

Im Bereich modellbasierter Ansätze (vgl. Abschnitt 2.3) wird die Evaluation anhand einer Modellreferenz durchgeführt, wobei Metriken für die Ähnlichkeit von 3D-Modellen verwendet werden [Seitz u. a., 2006]. Einen pixelbasierten Ansatz für modellbasierte Verfahren präsentieren Starck und Kilner in [Starck u. a., 2008; Kilner u. a., 2009], zielen jedoch ebenfalls auf den geometrischen Fehler ab. Andere Methoden, die mehr als zwei Kameras verwenden, synthetisieren die Ansicht einer realen Kamera und vergleichen diese dann wiederum mit herkömmlichen Metriken aus dem 2D-Bereich. Letzteres wurde auch für diese Arbeit in Betracht gezogen. Wie bereits im Abschnitt zur Stereoaufnahme beschrieben (vgl. Abschnitt 4), existiert neben der Aufnahme des Stereokamerapaares stets die Aufnahme mindestens einer weiteren Kamera. Die Kamera ist an der Stelle des Kommunikationsfensters und damit an der Position der virtuellen Kamera platziert. Sie diente als Orientierung für die Schauspieler und ihre durch Kalibrierung bestimmten externen und internen Parameter liefern die Daten für die Synthese.




**(a)** **(b)**

**Abbildung 6.1.:** PSNR verschiedener Synthesevarianten über die Zeit. Gemessen wurden der PSNR zwischen Referenz und translatorisch manuell korrigierter Synthese. Der Bewertungsbereich wurde durch die Maske der Referenz eingeschränkt. Gemessen wurde für verschiedene Testsequenzen (vgl. Abb. 6.3).

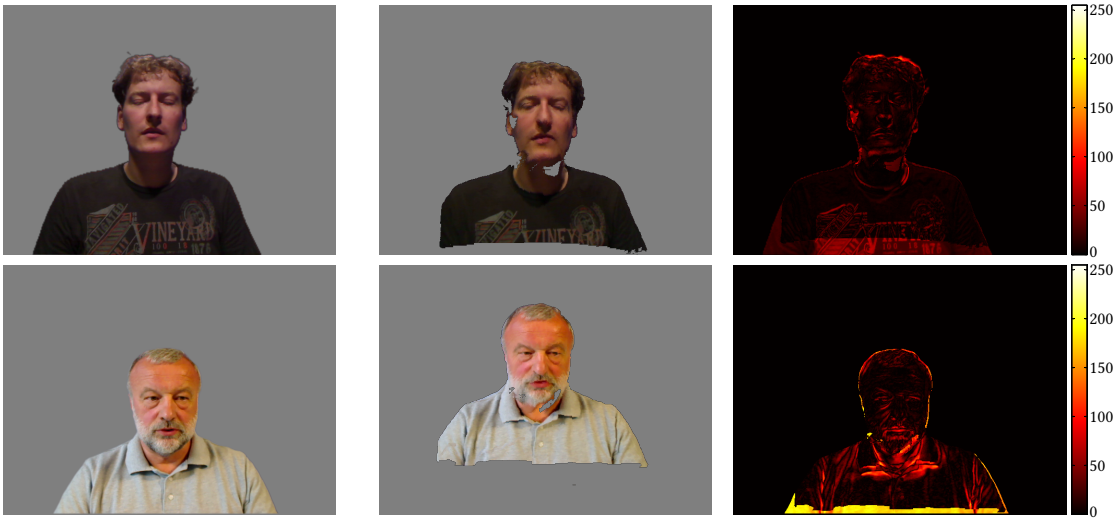
Versuche, die Qualität mittels PSNR für zwei Synthesevarianten auf alle Sequenzen der zweiten Testreihen zu bestimmen, zeigen jedoch, dass diese Metrik – und somit auch ähnliche Metriken auf Pixelbasis – für den Zweck nicht geeignet sind, wie in folgenden Ausführungen beschrieben. Im Versuch wurden die PSNR-Werte der Graustufenbilder unter der Berücksichtigung der Maske der Referenzansicht ermittelt. Zunächst wurde eine manuelle Positionskorrektur durchgeführt. Das Bild der synthetisierten Kamera wurde iterativ nach Augenmaß so positioniert, dass eine möglichst große Deckung mit dem Referenzbild erfolgt. Dieser Schritt muss durchgeführt werden, da die Synthese bei den gewählten Testsequenzen aus den rektifizierten Stereo-Ansichten erzeugt wurde (vgl. Abschnitt 5.3.3). Die Kalibrierung der Referenzkamera erfolgte hingegen bzgl. des originalen Kamerapaares. Eine translatorische Korrektur ist im Videobereich legitim und wird z. B. auch für beschnittene 2D-Videos eingesetzt [ITU-T, 2008]. Dennoch ist dem Autor bewusst, dass die manuelle Korrektur anhand nur eines Bildes der Sequenz nicht optimal ist. Der PSNR-Wert wurde aus dem jeweils in ein Intensitätsbild konvertierten[1] (vgl. Gl. (5.24)) Referenzbild $R$ und Synthesebild $S$ folgendermaßen ermittelt:

$$PSNR = 10log_{10}\left(\frac{v^2}{MSE}\right), \text{mit} \quad MSE = \frac{1}{|\mathcal{P}|}\sum_{\mathcal{P}}(R(\mathbf{p}) - S(\mathbf{p}))^2 \tag{6.1}$$

$v$ ist der maximale Intensitätswert, $\mathcal{M}$ die Menge der Pixel, die in der binären Alphamaske $M$ der Referenz sichtbar sind: $\mathcal{P} = \{\mathbf{p}|M(\mathbf{p}) = 1\}$.

Die sich ergebenden PSNR-Werte sind höchstens als Indiz zu verwenden (vgl. Abb. 6.1). So unterscheiden sie sich bei identischen Synthesevarianten sehr stark voneinander. Es ist offensichtlich,

[1] Präziser wäre eine Berechnung pro Farbkanal. Im aktuellen Kontext reicht jedoch diese Vereinfachung aus.




(a) (b) (c)

**Abbildung 6.2.:** Absolute Pixeldifferenzen als Qualitätsmaß der Synthese. Exemplarische Darstellung für Sequenzen 212(oben) und 259(unten) des Testdatensatzes. (a) Referenzbild der virtuellen Kamera. (b) Syntheseergebnis der Variante 11-05, Bild 180 ohne Positionskorrektur. (c) Absolute Differenz zwischen Referenz und Synthese mit manueller Positionskorrektur und Beschränkung auf die Alphamaske der Referenz.

dass die Werte extrem inhaltsabhängig sind, was eine Bewertung der Ergebnisse erschwert. Insbesondere Effekte wie Löcher oder nicht synthetisierte Bereich aufgrund fehlender Bildinformationen führen zu unterschiedlich stark ausgeprägten Differenzbildern. Dies ist exemplarisch in Abbildung 6.2 gezeigt. Weiterhin fällt auf, dass eine Synthesevariante mit Füllalgorithmus den PSNR-Wert erwartungsgemäß erhöht, dies jedoch nicht bei alle Sequenzen gleichmäßig geschieht. Schlussendlich ist festzuhalten, dass Maßnahmen wie *automatische* Positionskorrektur oder die Einbeziehung der Löcher und Ausreißer notwendig wären, um eine inhaltsunabhängige, geeignete Metrik zu entwickeln. Auch muss überprüft werden, wie weitere Fehlerausprägungen durch verschiedene Metriken erfassbar sind. Diese Maßnahmen in Kombination mit einer Anpassung der Wichtung an die subjektive Wahrnehmung führen letztlich wieder zur eingangs genannten Problemstellung, die Rittermann in seiner Dissertation behandelt [Rittermann, 2007]. Die vorliegende Arbeit bietet nicht den Rahmen für eine solch umfangreiche Fortführung der Arbeit, da der Fokus klar auf der Algorithmenentwicklung für Stereoanalyse und Bildsynthese liegt. Somit ist die Verwendung von 2D-Qualitätsmetriken oder des 3DVQVM zu diesem Zeitpunkt nicht realistisch. Es müssen andere Methoden der Evaluation verwendet werden.

| Variante Disparitätsalgorithmus | Variante Synthesealgorithmus | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 15 | 16 | 19 | 20 |
| 05 | 07-02 | 07-03 | 07-04 | 07-05 | 07-06 | 07-07 | 07-08 | 07-11 | 07-12 | 07-15 | 07-16 |
| 05-01 | 08-02 | 08-03 | 08-04 | 08-05 | 08-06 | 08-07 | 08-08 | 08-11 | 08-12 | 08-15 | 08-16 |
| 05-02 | 09-02 | 09-03 | 09-04 | 09-05 | 09-06 | 09-07 | 09-08 | 09-11 | 09-12 | 09-15 | 09-16 |
| 05-03 | 10-02 | 10-03 | 10-04 | 10-05 | 10-06 | 10-07 | 10-08 | 10-11 | 10-12 | 10-15 | 10-16 |
| 05-04 | 11-02 | 11-03 | 11-04 | 11-05 | 11-06 | 11-07 | 11-08 | 11-11 | 11-12 | 11-15 | 11-16 |
| 05-05 | 12-02 | 12-03 | 12-04 | 12-05 | 12-06 | 12-07 | 12-08 | 12-11 | 12-12 | 12-15 | 12-16 |
| 05-06a | 13-02 | 13-03 | 13-04 | 13-05 | 13-06 | 13-07 | 13-08 | 13-11 | 13-12 | 13-15 | 13-16 |
| 05-06b | 14-02 | 14-03 | 14-04 | 14-05 | 14-06 | 14-07 | 14-08 | 14-11 | 14-12 | 14-15 | 14-16 |
| 05-07a | 15-02 | 15-03 | 15-04 | 15-05 | 15-06 | 15-07 | 15-08 | 15-11 | 15-12 | 15-15 | 15-16 |
| 05-07b | 16-02 | 16-03 | 16-04 | 16-05 | 16-06 | 16-07 | 16-08 | 16-11 | 16-12 | 16-15 | 16-16 |
| 07 | 19-02 | 19-03 | 19-04 | 19-05 | 19-06 | 19-07 | 19-08 | 19-11 | 19-12 | 19-15 | 19-16 |
| 07-01 | 20-02 | 20-03 | 20-04 | 20-05 | 20-06 | 20-07 | 20-08 | 20-11 | 20-12 | 20-15 | 20-16 |
| 07-02a | 21-02 | 21-03 | 21-04 | 21-05 | 21-06 | 21-07 | 21-08 | 21-11 | 21-12 | 21-15 | 21-16 |
| 07-02b | 22-02 | 22-03 | 22-04 | 22-05 | 22-06 | 22-07 | 22-08 | 22-11 | 22-12 | 22-15 | 22-16 |

**Tabelle 6.1.:** Überblick über alle Algorithmusvarianten. Hellgrau hinterlegt sind die verwendeten Algorithmen für Experiment 1, dunkelgrau die Algorithmen für Experiment 2. Für Details sein auf die Tabellen 5.1 und 5.2 verwiesen. Zur schnellen Orientierung auf Tabelle 6.2 in diesem Kapitel.

## 6.2. Subjektive Einschätzung und Vorauswahl

Die Ausführungen im vorherigen Abschnitt deuten bereits darauf hin, dass eine subjektive Bewertung durch Personen notwendig ist, um eine Qualitätsbewertung vorzunehmen. Da solche Studien sehr umfangreich und zeitaufwändig sind, muss die Anzahl der Testitems (der synthetisierten Videos) gering gehalten werden. Die in den folgenden Abschnitten vorgestellten Experimente wurden in dem die Arbeit begleitenden DFG-Projekt von Sara Kepplinger im Rahmen ihrer Dissertation durchgeführt[Kepplinger, vorr. 2014]. Die Auswertung und Interpretation der Daten erfolgte durch den Autor. Die Tabellen 5.1 und 5.2 mit allen Algorithmusvarianten für Stereoanalyse und Synthese zeigen jedoch, dass eine sehr große Anzahl von Kombinationsmöglichkeiten existiert. Hinzu kommt, dass aufgrund der Abhängigkeit der Algorithmen vom Sequenzinhalt möglichst mehrere verschiedene Sequenzen für die Testdurchführung genutzt werden sollten. Eine Vorauswahl ist demnach unabdingbar. Diese sollte durch einen Experten geschehen, der die Prinzipien der verwendeten Algorithmen versteht und Artefakte einschätzen sowie deren Ursache deuten kann. Die Auswahl der in Tabelle 6.1 markierten Algorithmusvarianten erfolgte bei beiden Experimenten durch den Autor der Arbeit in Absprache mit Projektbeteiligten. Neben

der reinen Entscheidung hinsichtlich der Algorithmen waren noch andere Faktoren bei der Auswahl involviert. So war bei Experiment eins neben der reinen Bewertung der Algorithmen auch der Anwendungskontext „Videokommunikation“ auswahlbestimmend. Es wurden Sequenzen, in denen verschiedene Kommunikationsrollen gespielt wurden, ausgewählt. Auch wurden, um den Einfluss des Schauspielers bei der Bewertung gering zu halten, identische Kleidungsstücke bei allen Sequenzen verwendet. Letzteres ist hinsichtlich der Algorithmen eher als ungünstig einzuschätzen, da somit durch die Kleidung verursachte negative Effekte (z. B. homogene Regionen) wie auch positive Effekt bei allen Testitems wiederholt vorkommen. Experiment zwei war wesentlich fokussierter und primär auf die Algorithmen ausgerichtet. Neben der Vorauswahl anhand von Sichtung des Materials durch den Autor spielten insbesondere beim Testdatensatz für Experiment zwei auch die Art der Algorithmen (lokale, globale Disparitätsschätzung) sowie der Erkenntnisse aus dem ersten Experiment eine Rolle.

## 6.3. Erstes Experiment

### 6.3.1. Testmethodik

An dem im Salzburg stattfindenden Experiment nahmen 350 so genannte „unbedarfte Testteilnehmer“ (*naïve test participants*) zwischen 17 und 38 Jahren als Probanden teil. Diese sind „not directly involved in picture quality evaluation as part of their work and should not be experienced assessors“ [ITU-T, 2008]. Die Testitems (Videos) wurden für 10 s ohne Referenz (*single stimulus*) gezeigt und konnten vom Probanden genau einmal angesehen werden. Die Präsentation der Videos erfolgte auf einem 19 ″ LCD-Bildschirm in standardisierter Umgebung. Dieses Vorgehen entspricht den Empfehlungen ITU-R BT.500 für Videoevaluation [ITU-R, 2012]. Neben der Erhebung demographischer Daten wurde vorab die Anwendungssituation eines Videokommunikationsgespräches erläutert.

Innerhalb des Experiments wurden unter der Überschrift „Wie bewerten Sie das dargestellte Video?“ folgenden Daten erhoben:

- **dichotomes Merkmal**: Antwort auf die Frage: „Die Qualität des Videos ist akzeptabel? Ja/Nein?“
- **dichotomes Merkmal**: Antwort auf die Frage: „Ich fühle mich von der Person im Bild angesehen? Ja/Nein? “
- **kontinuierliches Merkmal (*Absolute Category Rating*)**: „Die Qualität des Videos ist...“ (Skala von „schlecht“ bis „gut“ normiert auf Werte von $0.0 \ldots 10.0$)
- **qualitative Einschätzung**: durch freie Formulierungen/Eingabe von Qualitätsattributen

| Var. | Kurzbeschreibung Disparität | Kurzbeschreibung Synthese |
|---|---|---|
| 09-02 | Vorfilter, lokales ZNCC, Gauß-Aggregation, einfache zeitl. Glättung, binäre Maske | Trif. Transfer, Bilineares Füllen |
| 09-03 | wie 09-02 | Trif. Transfer, Bilineares Füllen und Medianfilter |
| 11-02 | Vorfilter, lokales ZNCC, Gauß-Aggregation, Füllen, binäre Maske | Trif. Transfer, Bilineares Füllen |
| 11-03 | wie 11-02 | Trif. Transfer, Bilineares Füllen und Medianfilter |
| 12-02 | Vorfilter, lokales ZNCC, Gauß-Aggregation, Füllen und einfache zeitl. Glättung, binäre Maske | Trif. Transfer, Bilineares Füllen |
| 12-03 | wie 12-02 | Trif. Transfer, Bilineares Füllen und Medianfilter |

**Tabelle 6.2.:** Kurzbeschreibung der Algorithmen in Experiment 1. Für Details sei auf die Tabellen 5.1, 5.2 und 6.1 verwiesen.

### 6.3.2. Testdaten

Zum Zeitpunkt des ersten Experiments waren viele der in den Kapiteln 5.2 und 5.3 beschriebenen Methoden noch nicht entwickelt. Es wurden drei Disparitätsalgorithmen sowie zwei Synthesevarianten ausgewählt. Bezugnehmend auf die Tabellen 5.1 und 5.2, sind die Varianten in Tabelle 6.2 nochmals kurz beschrieben.

Die Position der virtuellen Kamera wurde am Ort einer der Referenzkameras (GT1) gewählt und durch Kalibrierung dieser bestimmt. Der Translationsvektor der Koordinatentransformation des System der linken zur virtuellen Kamera betrug $(2.79, -15.77, 13.66)^\top$ `cm` Die Euler Winkel der Rotation der Koordinatentransformation $(\epsilon_x, \epsilon_y, \epsilon_z) = (-13.9206, -3.7282, 0.4343]) \,^\circ$ im Koordinatensystem der Matlab Camera Calibration Toolbox (vgl. Abb. 4.7). Die virtuelle Kamera unterlag somit einer Translation sowie einer Rotation und lag weder auf der Basislinie der Stereokameras noch auf einer der vertikalen oder horizontalen Kameraachsen.

Für den Test wurden acht Testsequenzen aus dem Datensatz (vgl. Abschnitt 4.4) mit einer Länge von je 10 s ausgewählt. Die vier Schauspieler wurden bei vier der Sequenzen aufgefordert, in die Referenzkamera zu schauen in den vier verbleibenden Sequenzen den Blick über den Bildschirm wandern zu lassen, ohne dabei in die Referenzkamera zu sehen. Für tiefer gehende Untersuchungen bekamen die Schauspieler außerdem verschiedene Rollen zugewiesen, die jedoch für die in dieser Arbeit vorgenommenen Auswertungen zu Synthesequalität und Augenkontakt irrelevant sind. Bildausschnitte sind in Abb. 6.3 zu sehen. Ebenso wurden noch weitere Sequenzvarianten

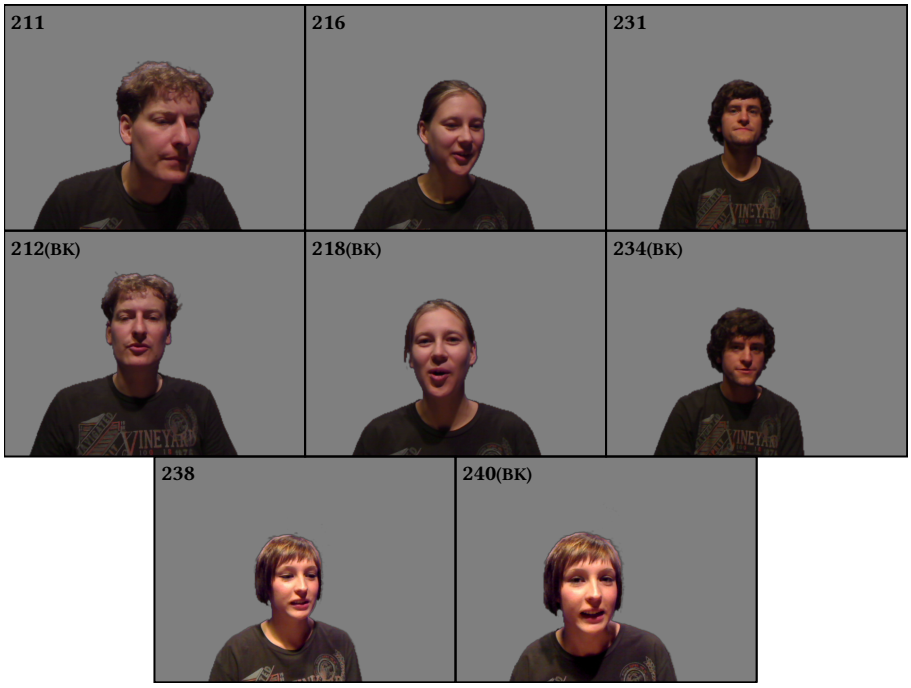



**Abbildung 6.3.:** Bildausschnitte der Videos der Referenzkamera des Testdatensets für Experiment 1. Mit „BK“ markierte Bilder sind diejenigen, bei denen die Schauspieler aufgefordert wurde, Blickkontakt herzustellen.

mit einer alternativen virtuellen Kameraposition untersucht, die jedoch auch nur im psychologischen Kontext der Videokommunikation eine Rolle spielten und daher hier nicht weiter betrachtet werden.

### 6.3.3. Ergebnisse

Die Anzahl gültiger Bewertungen für Akzeptanz und Augenkontakt (dichotomes Merkmal) ist in Tabelle 6.3 dargestellt. Da pro Proband nur ein knapper Zeitraum für den Test vorgesehen war, wurden die insgesamt 56 Testitems auf verschiedene Gruppen verteilt, wodurch sich eine unterschiedliche Verteilung der Bewertungen ergibt. Dies hat natürlich auch Implikationen für die Auswahl von Signifikanztests für diese Ergebnisse. Die Ergebnisse für Akzeptanz- und Augenkontaktraten sind in den Diagrammen 6.4 sowie in Tabelle 6.4 dargestellt. Die Signifikanz der Unterschiede, sowohl zwischen Akzeptanz als auch zwischen Augenkontaktraten bei Algorithmen und Sequenzen können für diese Werte nur mittels Clopper-Pearson Konfidenzintervallen abgeschätzt werden, die in den Diagrammen auch eingezeichnet sind. Für dichotome Merkmale bieten sich zwar auch parameterfreie statistische Test wie der Cochrane’s Q-Test an (vgl. [Strohmeier, 2011, S. 35]), diese setzen jedoch voraus, dass die Probanden auch alle Items bewertet haben, was aufgrund der hohen Anzahl der Testdatenitems bei diesem Experiment nicht gegeben war.

| Sequenz | Algorithmusvariante | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 09-02 | | 09-03 | | 11-02 | | 11-03 | | 12-02 | | 12-03 | | Orig | |
| **211** | 65 | 65 | 65 | 65 | kB | kB | 31 | 31 | 96 | 96 | 31 | 31 | 31 | 31 |
| **212** | 65 | 64 | 65 | 65 | 65 | 64 | 31 | 31 | 31 | 31 | 17 | 17 | 65 | 65 |
| **216** | 57 | 57 | 56 | 56 | 57 | 57 | 44 | 43 | 44 | 44 | 44 | 44 | 44 | 44 |
| **218** | 57 | 57 | 57 | 57 | 57 | 57 | 44 | 44 | 30 | 30 | 44 | 44 | 57 | 57 |
| **231** | 40 | 40 | 40 | 40 | 40 | 39 | 41 | 40 | 41 | 41 | 41 | 41 | 41 | 41 |
| **234** | 40 | 39 | 40 | 40 | 40 | 40 | 41 | 41 | 41 | 41 | 27 | 27 | 40 | 40 |
| **238** | 36 | 37 | 37 | 37 | 37 | 37 | 49 | 49 | 49 | 49 | 49 | 49 | 49 | 49 |
| **240** | 37 | 37 | 37 | 37 | 25 | 25 | 49 | 49 | 29 | 29 | 48 | 48 | 37 | 37 |

**Tabelle 6.3.:** Anzahl der gültigen Bewertungen für die dichotomen Merkmale Akzeptanz und Augenkontakt (grau hinterlegt) in Experiment 1.

| | Algorithmusvariante | | | | | | |
|---|---|---|---|---|---|---|---|
| | **09-02** | **09-03** | **11-02** | **11-03** | **12-02** | **12-03** | **Orig** |
| ∅ **Akzeptanzrate (BK) (%)** | 9.55 | 13.07 | 14.44 | 18.18 | 12.21 | 10.29 | 74.87 |
| ∅ **Akzeptanzrate (kein BK) (%)** | 11.62 | 12.12 | 16.42 | 22.42 | 12.17 | 11.52 | 84.85 |
| ∅ **Akzeptanzrate (gesamt)(%)**[2] | 10.58 | 12.59 | 15.26 | 20.30 | 12.19 | 10.96 | 79.40 |
| ∅ **Augenkontaktrate (BK) (%)** | 63.96 | 66.33 | 59.14 | 66.06 | 69.47 | 56.62 | 86.93 |
| ∅ **Augenkontaktrate (kein BK) (%)** | 17.59 | 19.19 | 27.07 | 19.02 | 18.26 | 21.82 | 29.09 |

**(a)**

| | Sequenz | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | **211** | **212** | **216** | **218** | **231** | **234** | **238** | **240** |
| ∅ **Akzeptanzrate (ohne Ref.) (%)** | 2.43 | 19.34 | 6.62 | 1.73 | 47.33 | 28.38 | 4.28 | 4.00 |
| ∅ **Akzeptanzrate (nur Ref.) (%)** | 87.10 | 78.46 | 88.64 | 61.40 | 85.37 | 85.00 | 79.59 | 78.38 |
| ∅ **Augenkontaktrate (ohne Ref.) (%)** | 0.35 | 71.69 | 17.94 | 40.14 | 62.24 | 68.86 | 5.04 | 78.67 |
| ∅ **Augenkontaktrate (nur Ref.) (%)** | 6.45 | 84.62 | 34.09 | 80.70 | 65.85 | 90.00 | 8.16 | 97.30 |

**(b)**

**Tabelle 6.4.:** Mittlere Akzeptanz- und Augenkontaktraten Experiment 1. Hellgrün markiert sind die Höchst-, rot die Minimalwerte. Die komplette Tabelle mit Angabe der Konfidenzintervallgrenzen befindet sich im Anhang in den Tabellen A.1 und A.2.

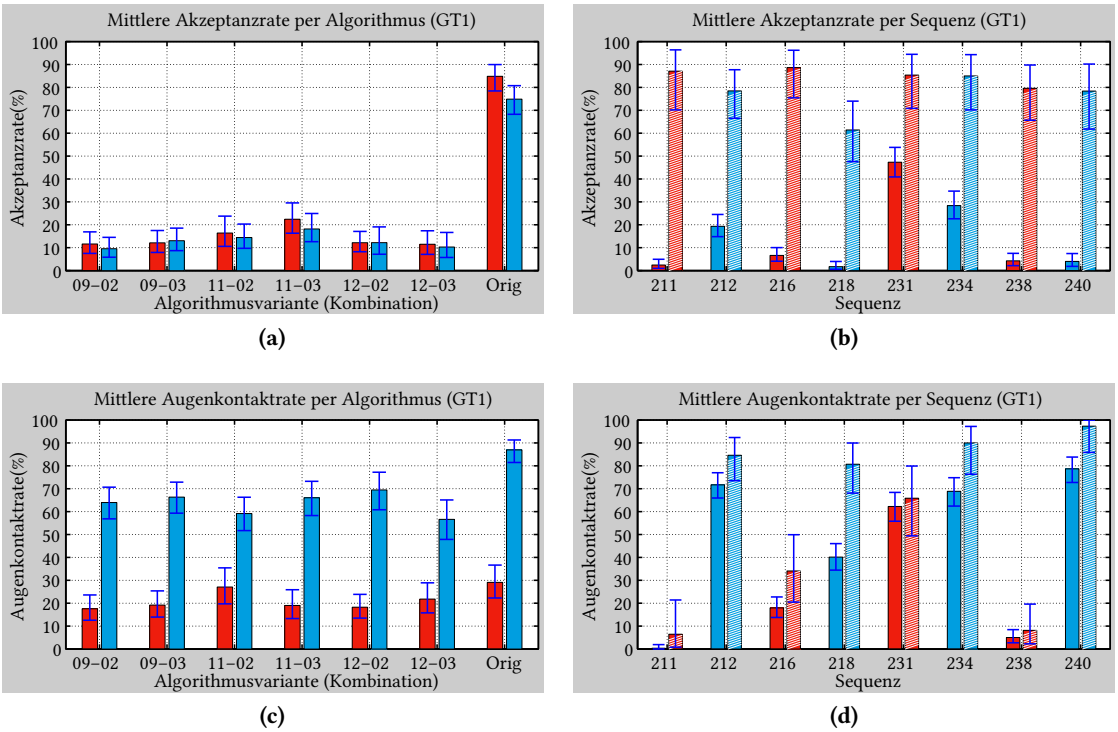



**Abbildung 6.4.:** Akzeptanz- und Augenkontaktraten Experiment 1. Blaue Balken beinhalten Daten aller Sequenzen, bei denen der Schauspieler Blickkontakt herstellen sollte. Rote Balken diejenigen ohne Blickkontakt. In Abb. b und d beinhalten die nicht schraffierten Balken die Daten aller synthetisierten Sequenzen, die schraffierten diejenigen für die Referenzen (GT1). Blau eingezeichnet ist das Clopper-Pearson Konfidenzintervall ($\alpha = 0.05$).

Es ist festzustellen, dass die Grundannahme, dass durch eine virtuelle Bildsynthese ein vorher nicht vorhandener Augenkontakt wieder hergestellt werden kann, bestätigt wird. Bei allen Algorithmusvarianten wurden für Sequenzen mit Augenkontakt in die Referenzkamera mittlere Augenkontaktraten von über $50\%$ mit einem Maximum von knapp $70\%$ für Algorithmus 12-02 erreicht. Dieser Wert ging bei Sequenzen, bei denen die Schauspieler aufgefordert wurden, nicht in die Kamera zu schauen, nicht über 28% Prozent hinaus. Im Vergleich zur Wahrnehmung der originalen Aufnahmen der Referenzkameras ist jedoch eine Reduzierung der Augenkontaktrate über alle Sequenzen und Algorithmen zu beobachten. Über die Algorithmen betrachtet fällt sie bei Algorithmus 12-03 (-30.32%) am stärksten aus. Ein Korrelation mit den Akzeptanzraten der Algorithmen ist nicht festzustellen (Pearson, c=-0.17, p=0.74). Bei den Sequenzen mit Blickkontakt fällt die Augenkontaktrate am stärksten bei Sequenz 218 (- 40.56%). Dies ist nicht überraschend, da diese Sequenz von allen die niedrigste mittlere Akzeptanzrate hat und der Augenkontakt durch Artefakte somit umso mehr zerstört wird. Eine Korrelation zwischen der Akzeptanzrate der Sequenzen mit Blickkontakt und der Reduzierung der Augenkontaktrate gegenüber dem Original ist zwar ebenfalls nicht gegeben. Jedoch ist der Korrelationskoeffizient (Pearson, c=-0.55,

p=0.45) höher als derjenige zwischen den Algorithmen und der Akzeptanzrate.

Werden die Akzeptanzraten betrachtet, so fällt zunächst auf, dass bereits die originalen Aufnahmen im Mittel Akzeptanzwerte von nur knapp 80% erreichen, keine Sequenz erreicht über 90%. Die Verwendung von Webcams kann dafür die Ursache sein. Es existiert ein geringer, jedoch nicht signifikanter Unterschied zwischen Sequenzen mit Blickkontakt und solchen ohne. Über die Algorithmen betrachtet, ist festzustellen, dass die höchste mittlere Akzeptanzrate von 20.3% durch Algorithmusvariante 11-03 erreicht wird und somit im Vergleich zur Referenz um rund 60% durch den Algorithmus reduziert wird. Interessant ist, dass über alle Algorithmen die Reduzierung der Akzeptanzrate bei Sequenzen ohne Blickkontakt im Mittel um knapp 10% höher ausfällt als bei Sequenzen mit Blickkontakt (-61.92% (BK) vs. -70.47%(KBK)). Denkbar ist, dass der vorhandene Blickkontakt die Wahrnehmung der Reduzierung der Qualität abschwächt.

Ob sich Algorithmus 11-03 von den anderen Algorithmen signifikant unterscheidet, kann – wie bereits begründet – nur anhand des berechneten Clopper-Pearson-Konfidenzintervalls ($\alpha = 0.05$) abgeschätzt werden. Werden die Akzeptanzbewertungen über alle Bewertungen für Varianten „BK“ und „kein BK“ analysiert, so liegt der zweitbeste Algorithmus 11-02 mit 15.62% unter der unteren Konfidenzintervallgrenze von 11-03 und kann somit als signifikant interpretiert werden. Einzeln betrachtet, trifft dies aufgrund der geringeren Anzahl von Bewertungen jedoch nicht zu. Daher wird dies nur als Tendenz gewertet. Allgemein ist festzustellen, dass die Akzeptanzraten sehr niedrig sind, und die Algorithmen in dieser Form nicht zum Einsatz kommen können. Unter der Beachtung der Aussagen zur Signifikanz im vorherigen Abschnitt kann jedoch festgehalten werden, dass die Verwendung der einfachen zeitlichen Glättung der Disparitätswerte eher negative Auswirkungen hat. Das konsistenzbasierte Füllen von Löchern in der Disparitätskarte hat hingegen einen positiven Effekt.

Wie bereits bei den Betrachtungen zur objektiven Bewertung festgestellt, ergeben sich zwischen den verschiedenen Testsequenzen signifikante Unterschiede zwischen den Testsequenzen. Prinzipbedingt wurde dies vom Autor auch so erwartet, wenn auch nicht in so starkem Ausmaß. So ist Sequenz 231 die mit 47.3% am besten bewertete. Die Sequenz ist eine sehr ruhige. Zudem sitzt die Person etwas weiter entfernt von der Kamera, so dass Verdeckungseffekte und Löcher durch Aufdeckungen nicht so starke Auswirkung haben. Bei den schlechtesten Sequenzen 218 und 211 bewegen sich die Personen sehr stark in Richtung Kamera, wodurch die negativen Effekte zunehmen.

Nach der Betrachtung der Algorithmenkombinationen werden die Daten nach Disparitäts- und Synthesealgorithmus gruppiert. Bei den Disparitätsalgorithmen ist der in den Varianten 11-xx verwendete Algorithmus mit einer Akzeptanzrate von 18.5% der am besten bewertete. Es bestätigt sich die bereits bei den Kombinationen gemachte Annahme, dass die Füllung zum

[2] Die geringe Abweichung vom arithm. Mittel aus den „BK„ und „kein BK“-Varianten ergibt sich durch die jeweils separate Berechnung des Erwartungswertes mittels Betaverteilung.

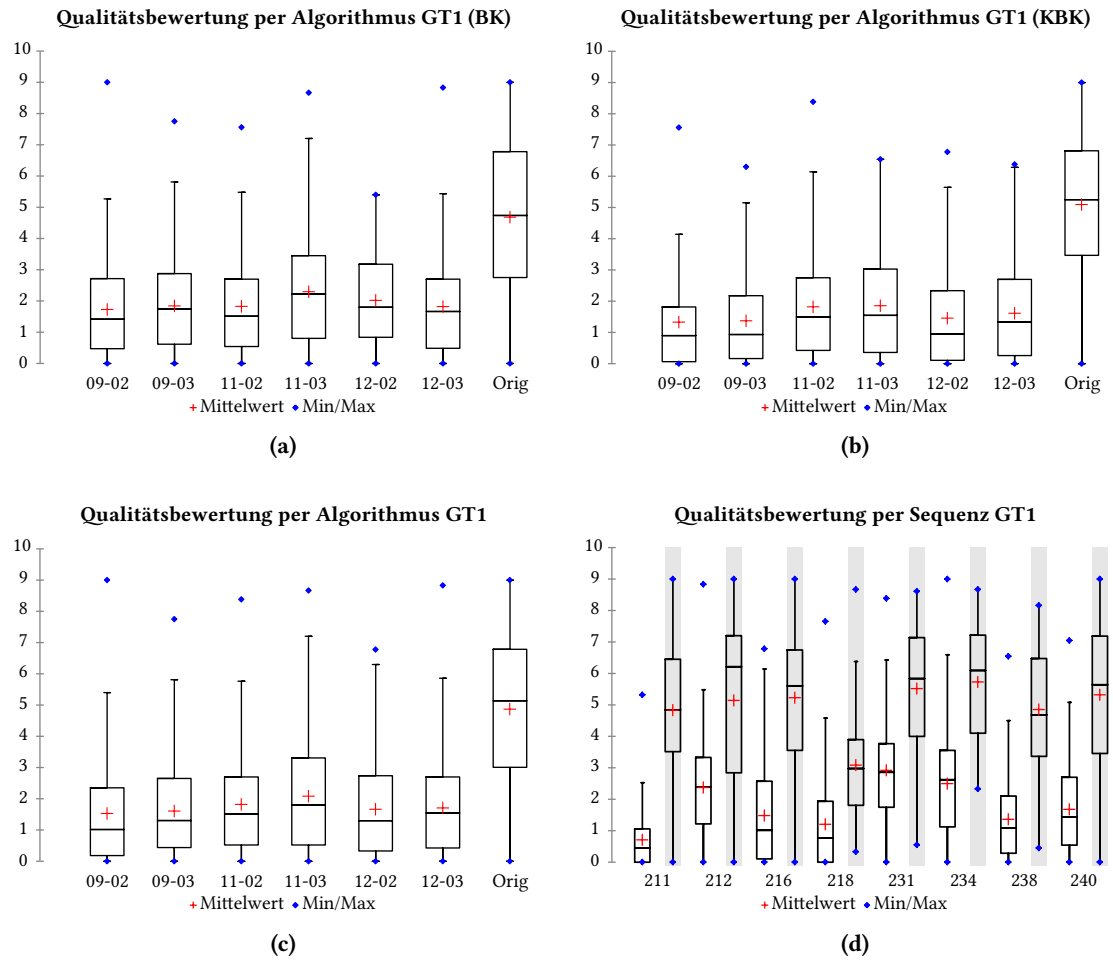



**Abbildung 6.5.:** Boxplots der Qualitätswahrnehmung Experiment 1. In Abb. (d) sind die nicht farbig hinterlegten Balken die Bewertungen ohne Referenzsequenz und die grau hinterlegten die Bewertungen nur für die Referenzsequenzen. Dargestellt sind unteres und oberes Quartil, Median, Mittelwert(rot), Antennen und Ausreißer(blau).

Qualitätsgewinn beiträgt, die einfach zeitliche Glättung sich jedoch negativ auswirkt. Für die Synthesevarianten, die sich nur in der anschließenden Medianfilterung unterscheiden, konnte kein signifikanter Unterschied festgestellt werden. Die kompletten Tabellen dieser Evaluation sind im Anhang (Tab. A.3 und A.4) zu finden.

Die Ergebnisse der Auswertung der Qualitätsbewertung auf der kontinuierlichen Skala decken sich mit den Werten für die Akzeptanzraten. Abb. 6.5 zeigt ausgewählte Boxplots. Auch hier geht als bester Algorithmus 11-03 hervor. Die Analyse der Daten erfolgte aufgrund der unverbundenen Stichproben mittels des parameterfreien Kruskal-Wallis-Tests [Kruskal und Wallis, 1952] als Alternative zur ANOVA. Die Signifikanz der Unterschiede zwischen einzelnen Variablen wurde mittels paarweisen Vergleich nach Dwass-Steel-Critchlow-Fligner [Critchlow

und Fligner, 1991] überprüft. Wird der Algorithmenvergleich gesamtheitlich durchgeführt (BK-*und* KBK-Bewertungen), so kann der Unterschiede von 11-03 zu allen anderen außer zu 11-02 und (erstaunlicherweise) 12-03 als signifikant bezeichnet werden (Wahrscheinlichkeiten der Falschannahme der Hypothese, dass die Daten aus unterschiedlichen Populationen stammen: $p_{09-02} \approx 0, p_{09-03} \approx 0.008, p_{11-02} \approx 0.0525, p_{12-02} \approx 0.038, p_{12-03} \approx 0.083, \alpha = 0.05$). Werden die Algorithmen jedoch getrennt nach Blickkontakt / kein Blickkontakt untersucht, so ergibt sich ein signifikanter Unterschied von 11-03 nur noch zu Algorithmus 09-02, was vermutlich auf die größere Varianz des geringeren Stichprobenumfangs zurückzuführen ist. Diese Unterschiede an den Grenzen zu signifikanten Beobachtungen decken sich mit den Konfidenzintervallen der Akzeptanzraten und es sind somit dieselben Schlussfolgerungen zu ziehen.

Zusammenfassend ist festzustellen, dass die Qualität der synthetisierten Videos nicht hoch genug ist, wenn auch für einzelne Sequenzen Akzeptanzraten bis zu 50% erreicht wurden. Dies ist besonders problematisch, da sich vermutlich durch das geringe Akzeptanzniveau geringe Unterschieden zwischen den in Experiment 1 gewählten Algorithmusvarianten schlechter feststellen lassen. Technische Gründe, die bereits in Abschnitt 3.3 thematisiert wurden, werden durch die erhobenen Aussagen der Probanden bestätigt. Wenn auch nicht systematisiert erhoben, so lassen sich doch die wichtigsten Gründe daraus ableiten. Eine Analyse und Zusammenfassung der Angaben zu Kernaussagen ergaben folgende Attribute (geordnet in der Häufigkeit ihrer Nennung):

1. unvollständige Darstellung
2. Verzerrungen
3. langsame, verzögerte Bewegungen
4. verschwommene Darstellung
5. Verpixelungen
6. Flecken
7. Unschärfe
8. Konturfehler
9. Flimmern
10. Falsche Proportionen
11. Hintergrund
12. Farbqualität

Punkte 1 und 2 sind klar den Fehlern Löchern und Verzerrungen und ihren jeweiligen Ursachen zuzuordnen (vgl. Abb. 3.3). Punkt 3 kann als einziger der Liste durch die Testumgebung verursacht worden sein, da teilweise Rechner zum Einsatz kamen, die das flüssige Abspielen der Videos nicht immer garantierten. Eine prinzip- oder algorithmenbezogene Ursache ist kaum anzunehmen, da die Berechnung für die Videos vorab erfolgte. Punkt 4 - 8 sind wiederum klar einzelnen Fehlerausprägungen zuzuordnen. Punkt 9 (Flimmern) ist ein temporales Artefakt, welches sich

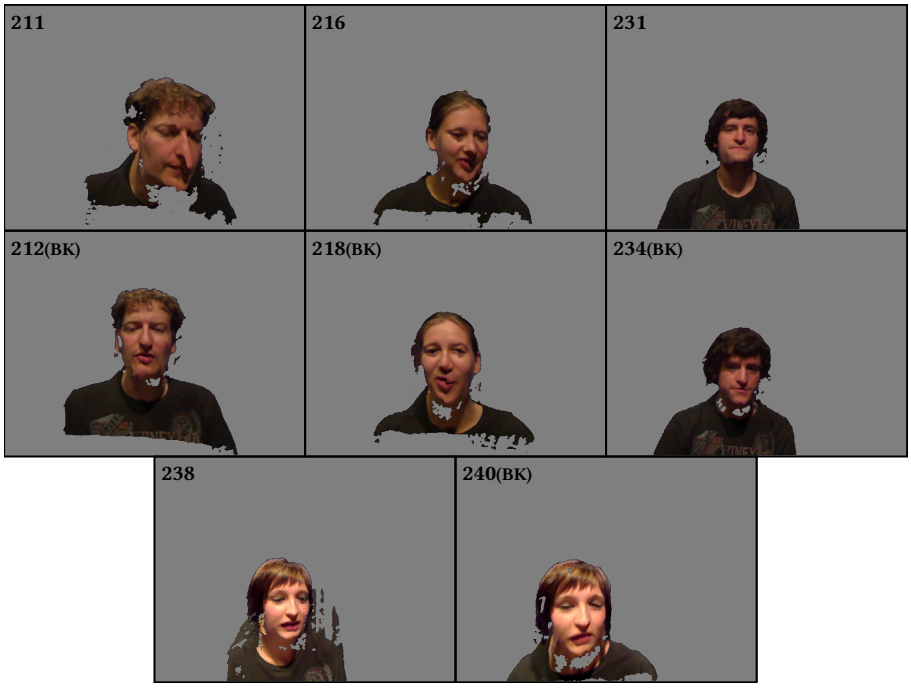



**Abbildung 6.6.:** Bildausschnitte der synthetisierten Videos der Referenzkamera für Experiment 1 (Algorithmusvariante 11-03). Die Bildnummern (Zeitpunkt der Aufnahme) entsprechen denen aus Abb. 6.3

durch die schnelle zeitliche Aneinanderreihung der verschiedenen nicht deterministischen Fehlerausprägungen ergibt. Punkt 9 (Falsche Proportionen) lassen sich den Verzerrungen zuordnen, während die beiden zuletzt genannten Fehlerquellen durch Parameter des System (Kamera und bewusst gewählte Darstellung ohne Hintergrund) verursacht werden. Die Entwicklung weiterer Algorithmen innerhalb der Arbeit nach Durchführung von Experiment 1 basieren, wenn auch nicht ausschließlich, auf diesen Informationen. Ziel war die Beseitigung der meistgenannten Fehlerausprägungen. Dies führt zu den Algorithmen, die in Experiment 2 untersucht wurden (vgl. Tab. 6.1).

## 6.4. Zweites Experiment

### 6.4.1. Testmethodik

Im Experiment kamen 32 (14 weibliche und 18 männliche) „unbedarfte Testteilnehmer“ zwischen 19 und 53 Jahren als Probanden zum Einsatz. Die Präsentation der Videos im zweiten Experiment erfolgte auf einem kalibrierten 19 ″ Bildschirm in abgedunkelter Umgebung mit konstanter Umgebungsbeleuchtung. Testaufbau und Umgebung wurden nach den Empfehlungen der ITU gestaltet [ITU-R, 2012]. Neben der Erhebung von demographischen Daten und Vorkenntnissen

im Anwendungsbereich wurde die Anwendungssituation eines Videokommunikationsgespräches erläutert. Die quantitativen Bewertungen fanden in zwei Abschnitten statt, wobei jeweils alle 44 Testvideos jeweils zweimal in zufälligen Reihenfolgen allen Probanden gezeigt wurden. Dabei wurden nach der Qualität und Akzeptanz der Videos gefragt. Anschließend wurde ein *Open Profiling of Quality - OPQ* durchgeführt, bei dem die Probanden 40 Videos anhand selbst gewählter Attribute bewerten. Für Details sei auf [Strohmeier, 2011] verwiesen. Im dritten Teil wurde die Wahrnehmung des Augenkontaktes erhoben. Die vier zusätzlichen Videos in der quantitativen Evaluation waren Videos, mit denen der Einfluss der nicht vorhandenen horizontalen Korrektur des Synthesebildes untersucht wurde, der zu einem „Schweben" der Person führt.

Innerhalb des quantitativen Teils des Experiments wurden unter der Überschrift „Wie bewerten Sie das dargestellte Video?" folgenden Daten erhoben:

- **dichotomes Merkmal**: Antwort auf die Frage: „Die Qualität des Videos ist akzeptabel? Ja/Nein?"
- **dichotomes Merkmal**: Antwort auf die Frage: „Ich fühle mich von der Person im Bild angesehen? Ja/Nein? "
- **Absolute Category Rating**: Videoqualität (unbenannte 11-Punkt-Skala $(0 \ldots 10)$ von „schlecht" bis „exzellent")

### 6.4.2. Testdaten

Basierend auf den quantitativen und qualitativen Ergebnissen aus Experiment 1 wurden die Algorithmen im Verlauf der Arbeiten weiter entwickelt, wie in den Kapiteln 5.2 und 5.3 beschrieben. Dabei standen vor allem das Füllen von Löchern und die Entfernung von Ausreißern in der Synthese im Mittelpunkt. Auch diese Varianten sind zwecks besserer Übersicht in Tabelle 6.5 nochmals kurz beschrieben. Variante 11-05 entspricht bis auf die Art des Warpings den besten Varianten 11-02 und 11-03 in Experiment 1. Die Auswirkungen des unterschiedlichen Warping-Prinzips machen sich aus mathematischer Sicht nur durch eine einheitliche Verschiebung des Synthesebildes bemerkbar (vgl. Abschnitt 5.3.3). Sie haben, wenn überhaupt, eher einem geringen negativen Einfluss auf die Qualität. Variante 11-05 kann somit als Referenz (Anker) für die durch die Algorithmenentwicklung erreichten Verbesserungen dienen. Die originalen Sequenzen wurden in diesem Experiment nicht gezeigt. Sie hätten die OPQ Auswertung erschwert bzw. unmöglich gemacht.

Die Position der virtuellen Kamera wurde am Ort einer der Referenzkameras (GT1) gewählt und durch deren Kalibrierung bestimmt. Für Sequenzen 212 und 234 betrug der Translationsvektor der Koordinatentransformation des System der linken zur virtuellen Kamera $(2.79, -15.77, 13.66)^\top$ `cm` Die Euler Winkel der Koordinatenrotation$(\epsilon_x, \epsilon_y, \epsilon_z) = (-13.9206, -3.7282, 0.4343])$ ° im Koordinatensystem der Matlab Camera Calibration Toolbox (vgl. Abb. 4.7). Für die Sequenzen 259

| Var. | Kurzbeschreibung Disparität | Kurzbeschreibung Synthese |
|---|---|---|
| 11-05 | Vorfilter, lokales ZNCC, Gauß-Aggregation, Füllen, Medianfilter und binäre Maske | 3D-Warping (APG 5), Tiefensortierung, Bilineares Füllen, vertikale Korrektur |
| 16-06 | Vorfilter, lokales ZNCC, Gauß-Aggregation, Füllen, Medianfilter und binäre Maske, Bewegungskomp. zeitl. Glättung | 3D-Warping (APG 5), Tiefensortierung, konturbasiertes Füllen, vertikale Korrektur |
| 16-16 | wie 16-06 | 3D-Warping (APG 30), Multisampling 16x, konturbasiertes Füllen, vertikale Korrektur |
| 22-06 | Vorfilter, globaler MAP-Schätzer (Sobel), Bewegungskomp. zeitl. Glättung | wie 16-06 |
| 22-16 | wie 22-06 | wie 16-16 |

**Tabelle 6.5.:** Kurzbeschreibung der Algorithmen in Experiment 2. Für Details sei auf die Tabellen 5.1, 5.2 und 6.1 verwiesen.

und 267 waren die Koordinatentransformation $(0.2732, -15.349319, 3.9565)^\top$ cm, und die Euler Winkel $(\epsilon_x, \epsilon_y, \epsilon_z) = (-12.2498, -4.0130, 0.6663)\,^\circ$.

Für den Test wurden vier Testsequenzen aus dem Datensatz (vgl. Abschnitt 4.4) mit einer Länge von je 10 s ausgewählt. Zwei davon wurden bereits in Experiment 1 genutzt. Drei der Schauspieler wurden aufgefordert, in die Referenzkamera zu schauen. Der Schauspieler in Sequenz 259 sollte den Blick über den Bildschirm wandern lassen, ohne dabei in die Referenzkamera zu sehen. Bildausschnitte der Referenzkameras sind in Abb. 6.7 zu sehen.

### 6.4.3. Ergebnisse

Die Ergebnisse für Akzeptanz- und Augenkontaktraten sind in den Diagrammen 6.8 dargestellt. Die Signifikanz der Unterschiede für diese dichotomen Merkmale konnte in diesem Experiment mittels Cochrane's Q-Test und kreuzweisem McNemar-Vergleich ermittelt werden, wobei die Nullhypothese $H_0$ annimmt, die Behandlungen sind gleich. Die Gegenhypothese $H_1$ nimmt an, die Behandlungen sind unterschiedlich. Das Signifikanzniveau $\alpha$ wird, wenn nicht anders angegeben auf $\alpha = 0.05$ gesetzt. $p$ gibt die Wahrscheinlichkeit des Risikos an, die Nullhypothese zurück zu weisen, obwohl sie wahr ist.

Vor einer eingehenden Analyse sei festgestellt, dass kein signifikanter Unterschieden zwischen Algorithmusvarianten mit eingefügtem Hintergrund (HG) und derselben Variante ohne Hintergrund besteht (Cochranes Q, $\alpha = 0.05$, 11-05: $p = 0.423$, 16-06: $p = 0.217$, 16-16: $p = 0.483$,

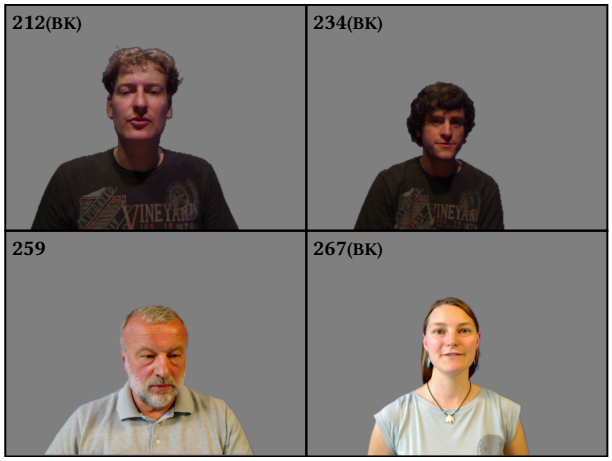



**Abbildung 6.7.:** Bildausschnitte der Videos der Referenzkamera des Testdatensatzes für Experiment 2. Mit „BK“ markierte Bilder sind diejenigen, bei denen die Schauspieler aufgefordert wurde, Blickkontakt herzustellen.

22-06: $p = 0.546$, 22-16: $p = 0.131$). Obwohl in Experiment 1 als störender Einfluss an elfter Stelle genannt, hat demnach das Hinzufügen eines Hintergrundes keinen positiven Effekt. Als Ursache hierfür wird vermutet, dass der Hintergrund nicht der originalen Szene entstammte und als statisches Bild eingefügt wurden. Falsche Beleuchtungseffekte oder geringen Diskrepanzen in den Größenverhältnissen zwischen Hintergrunde und Person könnten zu einer Abwertung geführt und somit kontraproduktiv gewirkt haben. Da die Abweichungen der Mittelwerte der Akzeptanzraten (HG vs. kein HG) pro Algorithmusvariante maximal 3.3% beträgt, werden in den folgenden Auswertungen diese Varianten jeweils derselben Algorithmengruppe zugeordnet (Bsp.: 16-06 und 16-06HG werden zu Gruppe 16-06 zusammengefasst). Die Auswirkung der vertikalen Anpassung wurde nur bei Algorithmus 11-05 untersucht. Hier ergibt sich eine geringe, wenn auch keine signifikante Verbesserung von 7% durch die Verschiebung der Person an den unteren Bildrand. Um die Anzahl der Testsequenzen gering zu halten, wurde diese Untersuchung nur für den Algorithmus 11-05 durchgeführt. Für alle anderen (neuen) Algorithmen wurde diese Verschiebung stets durchgeführt. Die Ergebnisse von Algorithmus 11-05 o.V. (ohne Verschiebung) werden im Folgenden nicht weiter analysiert.

Eine Betrachtung der Akzeptanzraten über alle gezeigten Algorithmusvarianten zeigt, dass die Maßnahmen zur Verbesserung der Synthese eine positive Wirkung haben und somit als Erfolg dieser Arbeit zu werten sind. Algorithmusvariante 16-06 (lokaler Disparitätsalgorithmus, kleine Synthesepunktgröße und konturbasiertes Füllen) wird am besten bewertet. Die mittlere Akzeptanzrate liegt bei ca. 60% und verbessert damit Algorithmus 11-05 aus dem ersten Experiment signifikant um 35% (McNemar, $p < 0.0001$). Auch gegenüber Varianten 16-16 (18.4%, McNemar, $p < 0.0001$) und 22-16 (22.6%, McNemar, $p < 0.0001$) ist Variante 16-06 der klare Gewinner. Gegenüber Algorithmus 22-06 besteht eine Verbesserung von 12.3%, die nicht als signifikant

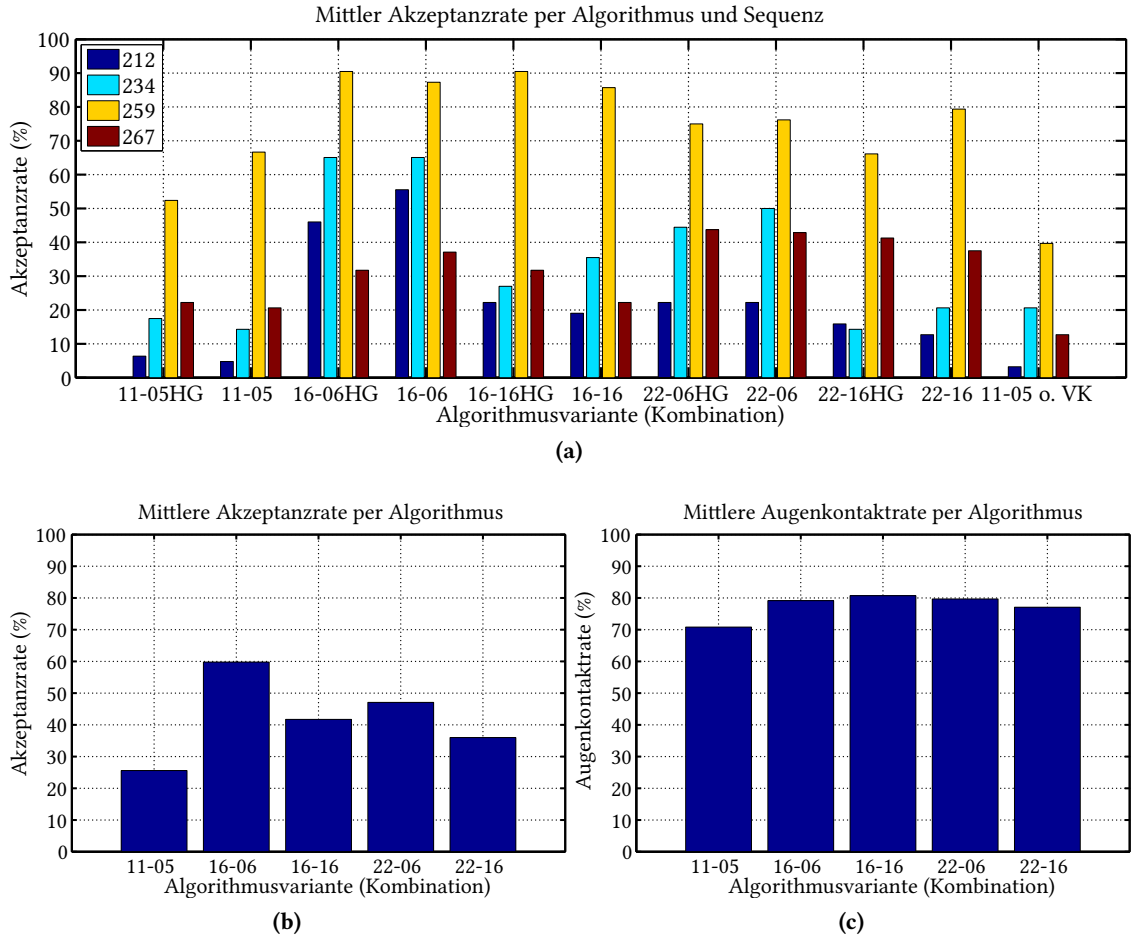



**Abbildung 6.8.:** Akzeptanz- und Augenkontaktraten Experiment 2.

gewertet wird. Die Algorithmen teilen sich dieselbe Synthesemethode und unterscheiden sich nur im Disparitätsalgorithmus.

Die Wahrnehmung des Augenkontaktes fällt bei den neuen Algorithmen nicht unter 79% bei den Szenen, bei denen die Schauspieler Augenkontakt in die Referenzkamera / zu synthetisierende Kamera herstellen sollten. Algorithmus 11-05 aus dem ersten Experiment fällt um ca. 10% gegenüber den neuen Algorithmen auf 70% ab, wenn auch nicht signifikant (McNemar, $p = 0.126$). Dennoch lässt der Wert die Vermutung zu, dass die Synthesequalität mit der Wahrnehmung des Augenkontaktes zusammenhängt.

Auch in Experiment 2 fällt wieder der starke Einfluss der verwendeten Testsequenz auf das Ergebnis auf. So erreichen die synthetisierten Videos der Sequenz 259 Akzeptanzraten von bis zu 90% während Sequenz 267 wesentlich schlechter abschneidet. Es ist zudem festzustellen, dass Algorithmusvariante 16-06 für diese Sequenz nicht das Optimum darstellt. Ursachen dafür können nur vermutet werden. Die in Sequenz 267 dargestellte Person trägt Ohrringe, die bei der Stereoanalyse zu Fehlern führen. Es mag sein, dass daraus resultierende Synthesefehler an

| | Algorithmusvariante | | | | |
|---|---|---|---|---|---|
| | **11_05** | **16_06** | **16_16** | **22_06** | **22_16** |
| **Akzeptanzrate (%)** | 24.848 | 59.394 | 41.010 | 47.071 | 35.758 |
| **Augenkontaktrate(%)** | 70.833 | 79.167 | 80.729 | 79.688 | 77.083 |

**Tabelle 6.6.:** Mittlere Akzeptanz- und Augenkontaktraten Experiment 2. Hellgrün markiert sind die Höchst-, rot die Minimalwerte.

so exponierter Stelle gerade bei den Varianten 16-X stärkere qualitätsminderne Auswirkungen haben, als bei den Varianten 22-X.

Die Auswertung der Qualitätsbewertung[3] auf der 11-Punkt-Skala $(0 \ldots 10)$ sollte mittels Varianzanalyse erfolgen (ANOVA). Die Vorbedingungen einer Normalverteilung der Messwerte sind jedoch nicht erfüllt (Lilliefors, Shapiro-Wilk, Anderson-Darling-Tests) so dass die Anwendung der ANOVA diskussionswürdig ist. Da von Experten oft argumentiert wird, dass eine ANOVA bei hinreichend großer Stichprobenmenge dennoch durchführbar ist, wird sie in dieser Arbeit probeweise auf die Daten angewendet. Zusätzlich wird der parameterfreie Friedman-Test durchgeführt, der keine Normalverteilung voraussetzt.

Die Ergebnisse bestätigen die Aussagen der Messung der Akzeptanzraten hinsichtlich des besten Algorithmus. So wird Algorithmusvariante 16-06 bzw. 16-06HG mit durchschnittlich 3.6 signifikant am besten bewertet (Friedman, Q-Beobachtet: 666.724, Q-Kritisch: 18.307, p-Wert (Zweiseitig) < 0.0001). Die Ergebnisse sowohl der ANOVA als auch des Friedman-Tests decken sich hinsichtlich der Signifikanz der Unterschiede. Die Ergebnisse des paarweisen Vergleichs aller Algorithmusvarianten sind in Tabelle A.5 im Anhang dargestellt. Das ermittelte Qualitätsniveau hinterlässt einen zwiespältigen Eindruck. Einerseits ist ein Wert von 3.6 auf der Qualitätsskala nicht zufriedenstellend, andererseits sind mittlere Akzeptanzraten von bis zu 60%. und für einzelne Sequenzen bis zu 90% als durchaus positiv zu bewerten. Abb. 6.10 zeigt Bildausschnitte der Ergebnissequenzen.

Zusammenfassend ist festzustellen, dass die in dieser Arbeit entwickelten Algorithmen einen signifikanten Beitrag zur Verbesserung der subjektiv wahrgenommenen Qualität der synthetisierten Bilder geleistet haben. Testsequenzen und Testmethodik wurden innerhalb zweier Experimente so gewählt, dass sie vergleichbar sind und diesen Nachweis mittels statistischer Auswertung erbringen können. Im folgenden Abschnitt werden die Ergebnisse, so weit möglich, in den Stand der Technik eingeordnet.

[3] Die Bewertung auf der 11-Punkt-Skala kann auch als *mean opinion score* interpretiert werden, wenn auch dieser üblicherweise auf einer 5-Punkt-Skala ermittelt wird.

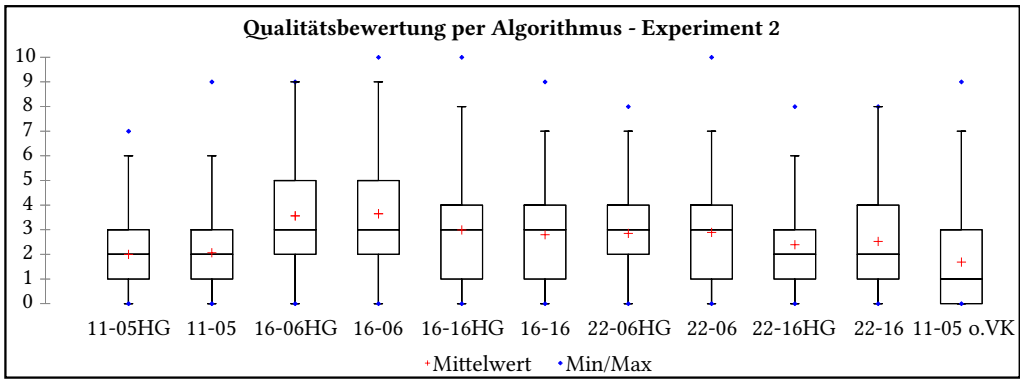



**Abbildung 6.9.:** Qualitätswahrnehmung Experiment 2.


**Abbildung 6.10.:** Bildausschnitte der synthetisierten Videos der Referenzkamera für Experiment 2 (Algorithmusvariante 16-06). Die Bildnummern (Zeitpunkt der Aufnahme) entsprechen denen aus Abb. 6.7. Der weiße Bereich bei den Sequenzen 259 und 267 ergibt sich durch das Abschneiden bei der vertikalen Korrektur. Die Testvideos beinhalten nur die grau hinterlegten Bereiche.

## 6.5. Vergleich mit dem Stand der Technik

Bereits in den vorherigen Abschnitten 6.1 bis 6.4 wurde angedeutet, dass ein objektiver Vergleich von Algorithmen zur Bildsynthese problematisch ist. Fehlende umfangreich evaluierte und somit anerkannte Metriken sind dafür der Hauptgrund. Dadurch ist auch eine Vergleichbarkeit zwischen verschiedenen Algorithmen kaum praktikabel durchführbar. Für den Teilbereich der Stereoanalyse des in dieser Arbeit beschriebenen Systems existiert mit dem Middlebury Stereo Evaluation Framework eine anerkannte Methode [Scharstein und Szeliski, 2002]. Jedoch beschränkt sich diese auf Standbilder und der Anwendungsfall wird außer Acht gelassen. Zudem sind in den verschiedenen Veröffentlichungen zum Thema Blickkorrektur unterschiedlichste Kameraaufbauten im Einsatz, die wiederum verschiedene algorithmische Ansätze im System erfordern. Die Erzeugung bzw. Verwendung eines gemeinsamen Basisdatensatzes ist damit extrem schwierig. Eine Analyse und ein Vergleich der Ergebnisse mittels subjektiver Tests hätte demnach die stärkste Aussagekraft. Werden die in Abschnitt 2.4 vorgestellten Veröffentlichungen hinsichtlich der Evaluation analysiert, ergibt sich ein eher ernüchterndes Bild.

Ott und Lewis geben in [Ott u. a., 1993] und [Lewis u. a., 1994] nur eine kurze qualitative Aussage. Weder objektive Maße noch eine subjektive Evaluation werden erwähnt. In der Veröffentlichung von Liu [Liu u. a., 1995] wird hingegen das Ergebnis eines subjektiven Tests vorgestellt. Jedoch wird nur die verbesserte Wahrnehmung des Augenkontaktes festgestellt. Mittels 12 Probanden wird eine Augenkontaktrate von 85% erreicht. Etwas geringere Raten wurden in dieser Arbeit erreicht. Ein Qualitäts- oder Akzeptanzwert wurde nicht erhoben. Die Methoden gleichen den in dieser Arbeit verwendeten insofern, als dass eine Stereoanalyse gefolgt von einer pixelbasierte Bildsynthese die Hauptbestandteile sind. Zum Zeitpunkt der Veröffentlichungen stellten bereits diese Schritte extrem zeitaufwändige Berechnungen auf damals verfügbarer Hardware dar. Auf verbessernde Nachbearbeitungsalgorithmen wie in dieser Arbeit vorgestellt, wurde verzichtet.

Die Veröffentlichung von Gemmel aus dem Jahr 2000 (vgl. [Gemmell u. a., 2000]) enthält eine Sammlung von guten Ideen das Problem des Blickkontaktes zu lösen. Im Gegensatz zur vorliegenden Arbeit ist dieser Ansatz eher den Verfahren unter der Nutzung expliziter Geometrie zuzuordnen. Die Autoren legen Wert auf grundlegend neue Ideen, die mitunter auch nur Teilbereiche der Problematik betreffen. Eine Evaluation findet weder mittels objektiver Maße noch subjektiver Experimente statt.

Die Veröffentlichung von Wolf aus dem Jahr 2010 konzentriert sich ausschließlich auf die Korrektur der Augen [Wolf u. a., 2010]. Wie bereits in Abschnitt 2.4 erwähnt, ergeben sich durch diesen Ansatz weniger Fehler im Gesamtbild. Einige Artefakte sind direkt im Augenbereich zu sehen. Die subjektive Wahrnehmung der Ergebnisse ist beeindruckend. Ein Experiment hinsichtlich Augenkontakt oder Qualität fand nicht statt. Eine quantitative Auswertung findet mittels eines Pixelfehler-Maßes gegenüber manuell annotierten Daten des verwendeten Augenmodells statt.

Die ganzheitliche Auswertung wird qualitativ vorgenommen und spricht Erfolge, aber auch Problemaspekte an. Da in der Arbeit keine Stereoanalyse und Bildsynthese stattfindet, ist sie aus Sicht der Algorithmen nicht mit dieser Arbeit vergleichbar. Dennoch ist der Ansatz sehr vielversprechend und kann, wenn der Aufwand minimiert wird, durchaus ein zukünftiges Mittel der Wahl sein.

Auch die Veröffentlichung von Yang aus dem Jahr 2002 ([Yang und Zhang, 2002]) lässt eine umfangreiche Evaluation vermissen. Hinsichtlich des Algorithmus ist wohl die Einbeziehung der Silhouette ein wichtiger Aspekt, der auch in dieser Arbeit zu Verbesserungen geführt hat. In dieser Arbeit konnten nur grundlegende Möglichkeiten zur Segmentierung der Person betrachtet und aus Zeitgründen nicht weiter verfolgt werden. Auch in der Veröffentlichung von Yang wird eine „Hintergrundsubtraktion“ durchgeführt, die jedoch nicht genauer beschrieben wird. Während die Vereinfachung der Silhouette durch Liniensegmente durch die Arbeit von Yang inspiriert wurde (vgl. Abschnitt 5.3.4), ist die weitere Verwendung in der Synthese in der vorliegenden Arbeit anders gelöst worden. Zudem ist festzuhalten, dass die virtuelle Kameraposition bei Yang auf die Basislinie zwischen oberer und unterer Kamera beschränkt wird.

Hinsichtlich des systematischen Gesamtkonzeptes ist diese Arbeit den Arbeiten von Criminisi et al. am nächsten [Criminisi u. a., 2003, 2007]. Die ersten Umsetzungen erfolgen ebenso wie bei Criminisi rein unter der Verwendung impliziter Geometrie. Die von Criminisi verwendete modifizierte dynamische Programmierung (DP) zur Stereoanalyse wurden auch in dieser Arbeit untersucht. Erste Ergebnisse der einfachen DP führten mit dem Material der einfachen Webkameras jedoch zu eher enttäuschenden Ergebnissen. Selbst die testweise Verwendung einer aufwändigeren Kostenaggregation mittels ASW (vgl. Abschnitt 2.2.5) führte zu keiner Ergebnisverbesserung. Der Schritt hin zu Criminisis Drei-Ebenen-Modell wurde daher nicht gegangen. Weiter gehend als in dieser Arbeit ist die Segmentierung über temporale Betrachtung des Hinter- und Vordergrundes bei Criminisi. Die Synthese ist jedoch auf rein translatorische Bewegungen limitiert. Hier geht die vorliegende Arbeit weiter, indem auch Rotationen der virtuellen Kamera ermöglicht werden. Während die erste Veröffentlichung von 2003 noch eine quantitative Evaluation vermissen lässt, so hat Criminisis in der späteren Veröffentlichung von 2007 die Qualität der Disparitätsanalyse mittels des bereits erwähnten Frameworks durchgeführt. Dabei bestätigt er auch die Aussage dieser Arbeit, dass eine gute Evaluation in diesem Framework nicht das oberste Ziel ist: „However, being at the top of this table is not the main objective of this paper; while accurate new-view synthesis is.“ [Criminisi u. a., 2007]. Eine gesamtheitliche Betrachtung findet sich leider auch in diesen Veröffentlichungen weder für die Wahrnehmung des Augenkontaktes noch für die Qualität der Synthesevideos.

Forschungsansätze, die das Plane Sweep-Verfahren verwenden, setzen mehr Kameras als die vorliegende Arbeit ein. In der Arbeit von Dumont et al. (vgl. [Dumont u. a., 2008, 2009]) wird ebenfalls eine Segmentierung, basierend auf eine Farbsegmentierung (*Green Screening*) oder

der Hintergrundsubtraktion angewendet. Durch die Anzahl der Eingangsbilder unterscheidet sich das weitere Vorgehen stark von dieser Arbeit. Ähnlichkeiten liegen in der Vorgehensweise, prinzipbedingte Fehler über Nachverarbeitungsprozesse zu korrigieren. So werden auch bei Dumont eine Konsistenzprüfung sowie ein Gaußfilter für die Verbesserung der Tiefenkarten eingesetzt. Anstelle des Füllalgorithmus nach Telea [Telea, 2004], der in dieser Arbeit zum Einsatz kommt, werden fehlerhafte Bereiche mittels morphologischen *Growings* gefüllt. Als weitergehend ist die stärkere Systembetrachtung zu nennen. So wird bereits die Augenposition für die virtuelle Bildsynthese übertragen sowie umfangreiche Betrachtungen der Berechnungsgeschwindigkeit durchgeführt. Eine objektive oder subjektive Evaluation von Augenkontakt oder Videoqualität bleiben jedoch auch Dumont et al. schuldig.

Der Plane Sweep-Ansatz von Muarayama/Mukai et al. ist im Vergleich zum vorher betrachteten wesentlich einfacher gehalten [Mukai u. a., 2009; Murayama u. a., 2010]. Auf Basis des SAD Kostenmaßes werden die günstigsten Tiefenwerte selektiert. Die Synthese erfolgt über Interpolation. Konkrete Fehler, die selbst in den Beispielbildern der Veröffentlichung zu sehen sind, werden nicht adressiert. Da die Evaluation auch hier nur qualitativ anhand einer Sequenz erfolgt, kann über die Qualität der Ergebnisse dieses Ansatzes nur spekuliert werden.

Die im letzten Jahrzehnt entwickelten Verfahren des Fraunhofer HHI für Videokonferenzsysteme können als wegweisend bezeichnet werden (vgl. Abhandlung im Abschnitt 2.4). Der HRM-Ansatz für die Tiefenanalyse wurde fortwährend weiter entwickelt. Die Arbeiten im „3D Presence“-Projekt nutzen ebenso wie diese Arbeit einen Konsistenzcheck bei der Tiefenanalyse. Da mehrere Kameras verwendet werden, wird die Stereoanalyse mit unterschiedlichen Basisabständen sowie sowohl horizontal also auch vertikal durchgeführt. Hinzu kommt die Kombination (*data fusion*) mit einer volumetrischen Rekonstruktion. Dies war unter den eingangs festgelegten Systemparametern in dieser Arbeit nicht in diesem Umfang möglich. Interessant sind die Arbeiten von Waizenegger et al. in [Waizenegger u. a., 2011]. Hier wird eine Tiefenanalyse mittels *Patch Sweep* durchgeführt. HRM kommt nicht mehr zum Einsatz. Ähnlichkeit zu dieser Arbeit besteht im verwendeten Kostenmaß NCC. Jedoch ist der Ansatz des modifizierten Plane Sweep und des Block-Matching grundsätzlich unterschiedlich. Die Syntheseposition beschränkt sich bei dem Patch Sweep-Ansatz offenbar auf die vertikale Achse der Kamera. Die in der Arbeit von Waizenegger dargestellten Ergebnisse sind subjektiv überzeugend. Auch das Demonstrationssystem namens „Eye Contact Engine“ zeigt sehr gute Ergebnisse. Ein subjektives Experiment zur Wahrnehmung des Augenkontakts sowie der wahrgenommenen Qualität der virtuellen Ansicht wird nicht durchgeführt. Innerhalb des Projektes, in dem die Arbeiten zu dieser Dissertation durchgeführt wurden, gab es Bestrebungen, das System auch subjektiv zu evaluieren. Leider war der Transport- und technische Aufwand zu hoch für eine Realisierung dieses Vorhabens.

Zusammenfassend ist festzustellen, dass die vorliegende Arbeit eine konsequente Fortführung verschiedener Veröffentlichungen zum Thema Blickkorrektur in Videokonferenzsystemen ist.

So sind grundlegende Vorverarbeitungsschritte wie die Segmentierung bei fast allen anderen Systemen zu finden. Hinsichtlich der Stereoanalyse wird auf bewährte Verfahren zurückgegriffen. Verschiedene Ansätze wurden implementiert (lokales Matching mit verschiedenen Kostenmaßen, DP, ein schnelles globales Verfahren). Manche davon wurden bereits nach ersten Experimenten verworfen, manche entsprechend erweitert (temporale Glättung, konsistenzbasiertes Füllen nach Telea) und in das Gesamtsystem für eine eingehende Evaluation integriert. Im Vergleich zu vielen aufgeführten Veröffentlichungen wird innerhalb der vorliegenden Arbeit eine ganzheitliche Systembetrachtung durchgeführt. Eine konkrete und systematische Analyse von Problemen und Fehlertypen (vgl. Abschnitt 3) findet statt und entsprechende Schlüsse werden daraus gezogen. Diese determinieren das Vorgehen bei der Entwicklung weiterer Algorithmen. Während viele Veröffentlichungen besonders die Stereoanalyse in den Vordergrund stellen, wird in dieser Arbeit der Fokus auf die Verbesserung der Synthese gelegt. Ein neues konturbasiertes Verfahren zur Beseitigung von prinzip- und algorithmenbedingen Fehlern wird vorgestellt. Verschiedenste Varianten der Bildsynthese (Trifokaler Transfer, 3D-Warping, unterschiedliche Renderingvarianten zur Vermeidung von Aliasing) werden entwickelt und implementiert. Die Verwendung objektiver Maße war mangels einer geeigneten Methode in keiner der Arbeiten ganzheitlich nutzbar. Ein klares Alleinstellungsmerkmal der vorliegenden Arbeit ist die umfassende Evaluation der Ergebnisse des Gesamtsystems mittels zweier subjektiver Experimente. In keiner der im Stand der Technik aufgeführten Veröffentlichungen wurde eine solche durchgeführt. Selbstkritisch muss angemerkt werden, dass die Experimente nur für die selbst entwickelten Algorithmen durchgeführt wurden. Eine direkte Vergleichbarkeit ist daher leider nicht möglich. Der dafür zu betreibende technische Aufwand wäre jedoch immens gewesen, und hätte den Rahmen dieser Arbeit gesprengt. Die eigene Vorgehensweise, die Ergebnisse sowie Ideen, die aus Zeitgründen nicht mehr den Weg der Realisierung gingen, werden im folgenden Kapitel diskutiert.

## 6.6. Diskussion, Kritik und Ausblick

Im Folgenden sollen die in den vorigen Abschnitten beschriebenen Vorgehensweisen, Algorithmen und deren Ergebnisse zusammenfassend kritisch betrachtet werden. Ideen, die während der Arbeit entstanden, jedoch nicht realisiert wurden, werden hier genannt. Die Abhandlung orientiert sich dabei an der Reihenfolge der Verarbeitungskette (vgl. Abschnitt 3.1). Abschließend wird – auch bezugnehmend auf neue technische Möglichkeiten, die in der Arbeit nicht verfügbar waren oder erst in deren Verlauf verfügbar wurden – ein Ausblick gegeben.

**Aufnahmeverfahren und Kameras:** Die in Abschnitt 4 vorgestellten Arbeiten und Entwicklungen zur Aufnahme wurden primär durch die Notwendigkeit der Erzeugung von Testmaterial motiviert. Die Spezifikation des Anwendungsfalls Heimvideokommunikation führte daher schnell zu der Entscheidung für Webcams. Auch wenn zur Zeit der Entscheidung die qualitativ besten

Modelle in diesem Produktsegment gewählt wurden, führte deren Verwendung zu Problemen. Sensor- und Bildqualität, mangelnde robuste Befestigungsmöglichkeiten beeinflussten akkurate Kalibrierung und die Algorithmen selbst. Als positiv ist festzustellen, dass dadurch die Algorithmusauswahl implizit in Richtung der realen Anwendung gelenkt wurden. Werden jedoch die aktuellen Verbesserungen bei den Entwicklungen von Webcams betrachtet, so kann auch anders argumentiert werden. Höhere Anforderungen an die Qualität der Kameras hätten bessere Ergebnisse bei denselben Algorithmen erzielt und wären heutzutage auch mit Webcams erreichbar.

Die Positionierung der Kameras wurde bereits thematisiert. Auch hier wurde vom Anwendungsfall gesteuert entschieden. Zum Zeitpunkt der Testdatenerzeugung kamen erste stereoskopische Webcams wie die Minoru 3D auf den Markt, die durch ihren festen Aufbau ideal für die Blickkorrektur schien, jedoch die Qualität nicht lieferte. Anordnungen links und rechts bzw. ober- und unterhalb des Monitors wurden aufgrund zunehmender Bildschirmmaße sowie der unpraktischen Befestigungsmöglichkeiten weniger favorisiert. Sicherlich bieten sie, auch mit Blick auf den Stand der Technik (vgl. vorherigen Abschnitt) gewisse Vorteile. Probleme wie Verdeckungen treten jedoch ebenso, wenn auch an anderer Stelle, auf. Nach Abschluss der hier beschriebenen Arbeiten hat nach Ansicht des Autors wohl eine Kombination dreier Kameras das größte Potenzial zur Qualitätsverbesserung. Dadurch ließen sich die sehr kritischen, prinzipbedingten Löcher durch Aufdeckungen im Halsbereich durch Hinzunahme der zusätzlichen Information reduzieren. Praktische Probleme sowie durch Ungenauigkeiten der Kalibrierung verursachte neue Artefakte bei der Fusion der Bilder dürfen dabei jedoch nicht außer Acht gelassen werden. Ansätze, wie Hermite-Spline-basiertes Überblenden wurden vom Autor bzw. unter dessen Anleitung in [Kreibich, 2005] und [Weigel und Kreibich, 2006] untersucht, jedoch für die Anwendung Videokommunikation nicht weiter verfolgt.

Eine spannende Entwicklung hinsichtlich der Aufnahmetechniken wurde durch die Firma Microsoft mit dem Produkt *Kinect* angestoßen. Mit der Kinect gibt es erstmals eine preiswerte Tiefenkamera mit erstaunlichen Fähigkeiten. Wenn sich die Technologie weiter entwickelt, dann ist diese Technologie evtl. einer rein Kamera-basierten Lösung vorzuziehen. Erste Ansätze im Bereich der Videokommunikation wurden kürzlich in [Kuster u. a., 2012] vorgestellt. Die vorliegende Arbeit war jedoch schon zu weit fortgeschritten, um die neue Technologie zu integrieren.

**Algorithmen:** Die Auswahl und Weiterentwicklung der Algorithmen der Stereoanalyse und Bildsynthese sind primär durch vorhergehende Arbeiten, Erfahrungswerte, qualitative Kurztests, aber auch durch die Berechnungsgeschwindigkeit motiviert. In den jeweiligen Abschnitten 5.2 und 5.3 sind Gründe bereits erläutert worden.

Ein umfassende Messung und Betrachtung der Berechnungsdauer konnte im Rahmen der Arbeit nicht durchgeführt werden. Neben mangelnder Zeit war auch die teilweise experimentelle

Implementierung in Matlab ein Grund dafür. Daher lassen sich an dieser Stelle nur ungefähre Aussagen machen, die anhand von stichprobenartigen Messungen vorgenommen wurden. Bei den meisten Experimenten kam ein Standard-PC mit Quad Core Prozessor mit je $2.6$ GHz und einer Standard-Grafikkarte zum Einsatz. Der aufwändigste Teil ist die Stereoanalyse. Die ZNCC mit großem Fenster benötigt ca. $200$ ms pro Bild für eine Disparitätsbereich von $1 \ldots 140$. Das konsistenzbasierte Füllen benötigt nur $5$ ms pro Bild. Die 3D-Warping basierte Synthese auf der GPU ist sehr schnell und braucht rund $10ms$. Die nicht optimiert implementierte Konturvereinfachung, deren Warping und die folgende Füllung beansprucht mit $300$ ms pro Bild relativ viel Zeit, birgt jedoch noch viel Optimierungspotenzial. Sehr frühe Arbeiten des Autors hatten auch zum Ziel die aufwändigen Operationen des trifokalen Transfers zu beschleunigen (vgl. [Weigel und Schübel, 2007]), was aber prinzipbedingt an Grenzen stieß. Die Implementierung vieler Algorithmen erfolgte im Rahmen der Arbeit in einem Echtzeit-Framework, das während der Arbeit entstand, welches aus Platzgründen aber nicht weiter thematisiert wurde. Daher sei auf die begleitenden Veröffentlichung verwiesen [Weigel u. a., 2007]. Die Entwicklung der CPUs mit immer mehr Kernen sowie die massive Parallelisierbarkeit von Algorithmen auf Grafikarten lassen die Aussage zu, dass die in dieser Arbeit entwickelten Algorithmen noch viel Potential für eine Beschleunigung haben.

Sowohl allgemeine Beobachtungen als auch die Experimente zeigen, dass im Anwendungsfall der Videokommunikation die Qualität der einzelnen Algorithmen und somit die finale Bildqualität stark inhaltsabhängig ist. Kleidung, Schmuck, Haltung, Position und Bewegung der Person, Position des Kommunikationsfensters (Syntheseposition) haben einen Einfluss, der sich einerseits prinzipbedingt äußert (Aufdeckungen) aber auch algorithmenbezogen ist (homogene Kleidungsregionen). In der Evaluation wurde diese Diskrepanz durch die Wahl problematischer sowie weniger problematischer Testsequenzen zumindest im Ansatz kompensiert. Diese subjektiv getroffene Auswahl gilt es zukünftig zu automatisieren. Können beeinflussende Parameter automatisiert gemessen werden, dann ist es möglich, über bestimmte Einschränkungen die Anwendung bzw. sogar deren Nutzer zu steuern. So kann z. B. eine Region vorgegeben werden, in der sich die Abbildung des Nutzersgesichtes befinden muss. Logitech verfährt beispielsweise ähnlich bei seiner Kameraanwendung, bei der Computergrafikavatare über das Gesicht projiziert werden. Beim Verlassen eines bestimmten Bereichs wird ein Hinweis gegeben. Auch kann dann in schwierigen Fällen (starke Kopfdrehung, sehr schlechte Beleuchtungsverhältnisse) die Blickkorrektur komplett abgeschaltet werden. Getreu dem Motto „besser keine Korrektur als allzu schlechte Qualität“. Hier sind neben den Messverfahren umfangreiche Usability-Experiment vonnöten, die bestimmen, was dem Nutzer in einer solchen Kommunikationssituation zuzumuten ist. Sind Methoden zur Messung bestimmter Aufnahmen vorhanden, so ließen sich diese ebenso zur Steuerung der Algorithmen einsetzen. Adaption von Parametern der Disparitätsanalyse (Disparitätsbereich, Fenstergröße, etc.) bis hin zur adaptiven Anwendung verschiedener Algorithmen wären denkbar. Auch für die Synthese könnten bisher empirisch bestimmte Parameter wie der

Abschwächungsfaktor oder die Ausgangspunktgröße so gesteuert werden.

Als unbefriedigend muss der geringe Effekt der zeitlichen Glättung der Disparitätskarten bezeichnet werden. Der Aufwand der Bewegungsanalyse steht in keinem Verhältnis zum Effekt. Evtl. ist eine zeitliche Kostenraumglättung hier eher Mittel der Wahl, da diese auch schon örtlich einen positiveren Effekt hat, als eine örtliche Glättung der Disparitätskarten.

**Evaluation und Qualität:** Die wichtigste Weiterentwicklung hinsichtlich der Qualitätsbewertung ist die bereits ausführlich thematisierte Etablierung eines objektiven Maßes (vgl. Abschnitt 6.1). Dieses Maß in Kombination mit einem einheitlichen Testdatensatz könnten die Vergleichbarkeit von Verfahren der Stereoanalyse und Bildsynthese zur Blickkorrektur und darüber hinaus erheblich vereinfachen. Die in dieser Arbeit durchgeführte Systematisierung, die durchgeführten Experimente sowie eine im selben Projekt angefertigte Dissertation[4] liefern viele wichtige objektive und subjektive Qualitätsparameter [Kepplinger, vorr. 2014]. Die Verknüpfung von technischen Parametern mit diesen subjektiven Qualitätsmessungen ist Ziel der parallel angefertigten Arbeit.

Ein in dieser Arbeit bewusst ausgeschlossener Parameter ist der fehlende Ton. Gerade bei der Videokommunikation ist der Ton jedoch ein entscheidender Informationsträger. Es darf vermutet werden, dass die Experiment, wären sie mit Ton durchgeführt worden, bessere Ergebnisse durch die multimodale Medienwahrnehmung ergeben hätten. Dieses Thema ist jedoch ein komplett eigener Forschungsbereich. Überlegungen hierzu wurden beispielsweise bereits in [Beerends und Caluwe, 1999] angestellt.

Abschließend ist festzustellen, dass trotz der umfangreichen Betrachtungen innerhalb dieser Arbeit noch zukünftiger Forschungsbedarf besteht. Die Möglichkeiten neuer Aufnahme-, Berechnungs- und Darstellungstechnologien werden gewiss noch langjährige zukünftige Arbeiten auf diesem Gebiet motivieren.

[4] ggw. noch nicht veröffentlicht.

# 7. Zusammenfassung

Die vorliegende Arbeit leistet einen Beitrag zum Forschungsbereich der Stereoanalyse und Bildsynthese im speziellen Kontext der Videokommunikation. Die Arbeit wird motiviert durch die Idee, den typischerweise verlorenen Blickkontakt in der privaten Videokommunikation mittels der Synthese der Ansicht einer virtuellen Kamera – platziert in der Blickrichtung der Kommunizierenden – wieder herzustellen. Ziel der Arbeit ist es, System und Algorithmen so zu entwickeln, dass das erzeugte Video in akzeptabler Qualität vorliegt.

Mittels einer kurzen historischen Betrachtung der Videokommunikation wird der positive Einfluss des Blickkontaktes in der zwischenmenschlichen Kommunikation und der Videokommunikation verdeutlicht. Ziele, Thesen und Beitrag der Arbeit werden anschließend definiert. Es folgt eine tiefgehende Betrachtung der notwendigen technischen Grundlagen im Bereich Stereoanalyse und Bildsynthese. Aufbauend auf diesen Grundlagen wird der der Stand der Technik im Bereich des bildbasierten Renderings im Allgemeinen sowie der Blickkorrektur mittels 3D-Analyse im Speziellen umfassend betrachtet. Die Einteilung folgt dabei einer etablierten Systematik.

Die Forschungsarbeit wird durch eine Systematisierung der Problemdimensionen begonnen. Die Definition der Verarbeitungskette, Randbedingungen und erste Qualitätsparameter werden diskutiert. Daraus werden Entscheidungen hinsichtlich Kameraanordnung und Aufnahmesystem getroffen. Notwendige Messungen hinsichtlich Synchronizität und Datenspeicherung werden durchgeführt. Diese Arbeiten führen zur Erstellung eines umfangreichen Video-Datensatzes, anhand dessen die Algorithmen entwickelt und getestet werden.

Im Bereich der Algorithmen der Stereoanalyse werden etablierte lokale und globale Algorithmen analysiert und adaptiert. Verschiedene Kostenmaße, konsistenzbasiertes Füllen, zeitliche und örtliche Glättung sowie eine abschließende Segmentierung werden hinsichtlich des konkreten Anwendungsfalls der Blickkorrektur in der privaten Videokommunikation entwickelt. Darauf aufbauend werden die beiden Syntheseverfahren des trifokalen Transfers sowie des 3D-Warpings weiter entwickelt. Ein wichtiger Bestandteil dieser Arbeit ist ein konturbasiertes Füllverfahren sowie Maßnahmen im Bereich der Punktglättung. Die Entwicklung dieser Maßnahmen wird durch eigene qualitative Einschätzungen des Autors sowie anhand von Aussagen aus dem ersten subjektiven Experiment gesteuert.

Mithilfe der entwickelten Algorithmen werden mittels eines ebenfalls in der Arbeit entstanden umfangreichen Software-Frameworks Testsequenzen für subjektive Experimente erstellt. Die Entscheidung zugunsten einer aufwändigeren subjektiven Evaluation erfolgt nach der unbefriedigenden Recherche zu und stichprobenartigen Versuchen mit objektiven Qualitätsmaßen. Zwei umfangreiche Experimente mit zahlreichen Probanden bestätigen die Korrektheit der Annahme, Blickkontakt durch eine Bildsynthese herzustellen. Sie demonstrieren sowohl die sehr gute Wahrnehmung des Augenkontaktes als auch die signifikante Verbesserung der Akzeptanz und subjektiven Qualitätswahrnehmung durch die entwickelten Algorithmen. Eine qualitativer Vergleich mit dem Stand der Technik und eine Diskussion der Ergebnisse, gepaart mit einem Ausblick in die Zukunft des behandelten Forschungsgebietes, schließen die Arbeit ab.

# Mathematische Symbole

### Räume, Punkte und Geraden

| | |
|---|---|
| $\mathbb{R}^2$ | Zweidimensionaler Vektorraum über $\mathbb{R}$ (euklidsche Ebene). |
| $\mathbb{R}^3$ | Dreidimensionaler Vektorraum über $\mathbb{R}$ (euklidscher Raum). |
| $\mathbb{P}^2$ | Projektiver Raum des $\mathbb{R}^2$. |
| $\mathbf{x} = (x, y)^\top$ | Punkt im $\mathbb{R}^2$. |
| $\mathbf{x} = (x_1, x_2, x_3)^\top$ | Punkt im $\mathbb{P}^2$ (homogene Koordinaten). Es erfolgt *keine* separate Kennzeichnung von Punkten in homogenen Koordinaten. Ob homogene Koordinaten verwendet werden, ergibt sich aus dem Kontext oder wird explizit erwähnt.. |
| $\mathbf{l} = (a, b, c)^\top$ | Gerade im $\mathbb{R}^2$, auch Epipolarlinien. |
| $\mathbf{X} = (X, Y, Z)^\top$ | Punkt im $\mathbb{R}^3$. |
| $\mathbf{X} = (X_1, X_2, X_3, X_4)^\top$ | Punkt im $\mathbb{P}^3$ (homogene Koordinaten). Es erfolgt *keine* separate Kennzeichnung von Punkten in homogenen Koordinaten. Ob homogene Koordinaten verwendet werden, ergibt sich aus dem Kontext oder wird explizit erwähnt.. |

### Stereogeometrie

| | |
|---|---|
| $b$ | Basisabstand. |
| $\mathbf{l} = (a, b, c)^\top$ | Epipolarlinie, auch allg. Gerade im $\mathbb{R}^2$. |
| $\mathbf{e}$ | Epipol. |
| $\boldsymbol{\pi}$ | Epipolarebene. |
| $\mathtt{F}$ | Fundamentalmatrix. |

### Trifokale Geometrie

| | |
|---|---|
| $x^i$ | Tensor erster Stufe (Vektor) - Punkt. |

| | |
|---|---|
| $l_j$ | Tensor erster Stufe (Vektor) - Gerade. |
| $a^i_j$ | Tensor zweiter Stufe (Matrix). |
| $\mathcal{T}_i^{jk}$ | Trifokaler Tensor. |

**Besondere Matrizen & Vektoren**

| | |
|---|---|
| `I` | Einheitsmatrix. |
| `H` | Matrix einer projektiven Transformation (Homographie). |
| `R` | Rotationsmatrix. |
| `P` | Projektionsmatrix. |
| `K` | Kamerakalibrierungsmatrix (intrinsische Matrix). |
| `T` | allg. Transformationsmatrix. |
| $\mathbf{t}$ | Translationsvektor. |
| $\mathbf{K}$ | Vektor des Linsenverzerrungsmodells (5 Elemente). |

**Diskrete Bilder**

| | |
|---|---|
| $I(x, y)$ | Diskret abgetastetes Bild. |
| $M(x, y)$ | Alphamaske. |
| $D(x, y)$, $D(\mathbf{p})$ | Disparitätskarte. |
| $R(x, y)$ | binäre Ausschlusskarte. |

**Kamera**

| | |
|---|---|
| $f$ | Kamerabrennweite. |
| $(p_x, p_y)^\top$ | Kamerahauptpunkt. |
| $\mathbf{C}$ | Kamerazentrum. |
| $\delta_x$ | Breite eines Sensorelementes in x-Richtung. |

**Segmentierung**

| | |
|---|---|
| $P(I_t)$ | Pixelprozess - Segmentierung. |
| $\nu()$ | Normalverteilung - Segmentierung. |

**Rektifizierung**

| | |
|---|---|
| $\mathcal{Z}$ | Menge der Pixelindizes der rekt. linken Ansicht. |

| | |
|---|---|
| $\mathcal{Q}$ | Menge der Pixelindizes der linken Ansicht. |
| $(q_z)_{z\in\mathcal{Z}}$ | Familie der Pixelindizes der linken Ansicht m. Bezug zur rekt. linken Ansicht. |
| $(k_r)_{r\in\mathcal{R}}$ | Familie der Pixelindizes der rekt. rechten Ansicht m. Bezug zur rechten Ansicht. |
| $(r_q)_{q\in\mathcal{Q}}$ | Familie der Pixelindizes der rechten Ansicht m. Bezug zur linken Ansicht. |
| $(m_z)_{z\in\mathcal{Z}}$ | Familie der durch Analyse bestimmten korr. Pixelindizes der rechten rekt. Ansicht m. Bezug zur linken rekt. Ansicht. |

**Stereoanalyse**

| | |
|---|---|
| $d$ | Disparität. |
| $C(x,y,d)$**,** $C(\mathbf{p},d)$**,** $C(\mathbf{p},\hat{\mathbf{p}})$ | Kostenfunktion (Korrespondenzsuche). |
| $\mathbf{p}$ | (diskreter) Punkt eines Bildes im $\mathbb{R}^2$ (Pixel). |
| $\hat{\mathbf{p}}$ | Korrespondierender Punkt zu $\mathbf{p}$ (Pixel). |
| $\mathbf{q}$ | (diskreter) Punkt eines Bildes im $\mathbb{R}^2$ (Pixel). |
| $\hat{\mathbf{q}}$ | Korrespondierender Punkt zu $\mathbf{q}$ (Pixel). |
| $G(u,v)$ | Gaußscher Glättungsfilterkernel (Kostenglättung). |
| $w(\mathbf{p},\mathbf{q})$ | Adaptives Kostengewicht (ASW). |
| $\Delta c$ | Farbähnlichkeitsmaße (ASW). |
| $\Delta g$ | Maß für die Nähe (ASW). |
| $\mathbf{r}$ | (diskreter) Punkt eines Bildes im $\mathbb{R}^2$ (Pixel) (Füllung nach [Telea, 2004]). |
| $B_\epsilon$ | Pixelumgebung (Füllung nach [Telea, 2004]). |
| $w(\mathbf{p},\mathbf{r})$ | Wichtungsfunktion (Füllung nach [Telea, 2004]). |
| $\mathbf{b}$ | Bewegungsvektor (zeitliche Glättung Disparitätskarte. |

**Globale Stereoanalyse**

| | |
|---|---|
| $\mathcal{S}$ | Menge von Sites (globale Stereoanalyse). |
| $f$ | Bestimmtes Labeling (globale Stereoanalyse). |

| | |
|---|---|
| $\mathbb{F}$ | Konfigurationsraum - Menge aller möglichen Labelings (globale Stereoanalyse). |
| $P(f)$ | Verbundwahrscheinlichkeit einer Konfiguration von Labels (globale Stereoanalyse). |
| $\mathcal{C}$ | Menge aller Cliquen (globale Stereoanalyse). |
| $U(f)$ | Energiefunktion (globale Stereoanalyse). |
| $E(d)$ | Energiefunktion, alt. Schreibweise (globale Stereoanalyse). |
| $P(f\|b)$ | A-posteriori Wahrscheinlichkeit (globale Stereoanalyse). |
| $P(b\|f)$ | Likelihood (globale Stereoanalyse). |
| $\mathbf{f}$ | Merkmalsvektor (globale Stereoanalyse, nach [Geiger u. a., 2010]). |
| $\mu(\mathbf{S}, \mathbf{o}^{(l)})$ | Stückweise lineare Funktion (globale Stereoanalyse, nach [Geiger u. a., 2010]). |

**Sonstige Symbole**

| | |
|---|---|
| $\mu$ | Mittelwert - Kontext unterschiedlich. |
| $\omega$ | Wichtungsfaktor - Kontext unterschiedlich. |
| $\sigma$ | Standardabweichung - Kontext unterschiedlich. |

# Abkürzungen

| | |
|---|---|
| 3DW | 3D-Warping. |
| AD | Absolute Difference – absolute Intensitätsdifferenz. |
| ANOVA | Analysis of Variance – Varianzanalyse. |
| APG | 3D-Warping: Ausgangspunktgröße (vgl. Gl. (5.56)). |
| ASW | Adaptive Support Windows. |
| BK | Blickkontakt. |
| BKZG | bewegungskompensiertes zeitliches Glätten. |
| BM | binäre Maskierung. |
| CCD | Charge-coupled Device. |
| CMOS | Complementary Metal Oxide Semiconductor. |
| CSAA | Coverage Sample Anti-Aliasing. |
| DP | Dynamische Pogrammierung. |
| `dpi` | Dots per inch. |
| DRL | Derektifizierte linke Ansicht & Derektifizierte Disparitätskarte. |
| FBL | Füllen von Löchern: Nachbar bilinear. |
| FKB | Füllen von Löchern: Konturbasiert, Telea. |
| F-Matrix | Fundamentalmatrix. |
| FSAA | Full Screen Anti-Aliasing. |
| GF | Gaußscher Filter. |
| GMAP | globaler MAP Schätzer. |
| GT | Ground Truth – Referenz. |
| HDTV | High Definition Television. |
| HG | Hintergrund. |
| HRM | Hybrid Recursive Matching. |
| HTML | Hypertext Markup Language. |
| ICM | Iterative Conditional Modes. |
| ITU | International Telecommunication Union. |
| JPEG | Joint Photographic Experts Group, Bildkompressionsverfahren. |
| KBF | Konsistenz-basiertes Füllen. |

| | |
|---|---|
| KBK | Kein Blickkontakt. |
| KLT-Tracker | Kanade-Lucas-Tomasi Tracker. |
| LCD | Liquid Crystal Display – Flüssigkristallanzeige. |
| LDI | Layered Depth images. |
| LWTA | lokale Korrespondenzsuche WTA. |
| MAD | Mean Absolute Difference. |
| MAP | Maximum A-posteriori. |
| MF | Medianfilter. |
| M-JPEG | Motion-JPEG, Videokompressionsverfahren. |
| MRF | Markov Random Field - Markowsches Zufallsfeld. |
| MS | 3D-Warping: Multisampling 16x. |
| MSAA | Multi-Sampled Anti-Aliasing. |
| MSAD | Mean Squared Difference. |
| MSE | Mean Squared Errors. |
| NCC | Normalized Cross-Correlation – normalisierte Kreuzkorrelation. |
| OPQ | Open Profiling of Quality. |
| PC | Personalcomputer. |
| PEVQ | Perceptual Evaluation of Video Quality. |
| PNG | Portable Network Graphics. |
| QPF | Quadratische Punktform. |
| RANSAC | Random Sample Consensus. |
| RGB | Farbmodell - Rot, Grün, Blau. |
| RL | Rektifizierte linke Ansicht & Rektifizierte Disparitätskarte. |
| SAD | Sum of Absolute Differences – Summe absoluter Intensitätsdifferenzen. |
| SD | Squared Difference – quadrierte Intensitätsdifferenz. |
| SIFT | Scale-invariant feature transform. |
| SSAA | Super-Sampled Anti-Aliasing. |
| SSD | Sum of Squared Difference – Summe quadrierter Intensitätsdifferenzen. |
| SSIM | Structural Similarity. |
| SVT | Shared Virtual Table Environment. |
| TFT | Trifokaler Transfer. |
| TS | 3D-Warping: Tiefensortierung. |
| USB | Universal Serial Bus. |
| UXGA | Ultra Extended Graphics Array (1600 px $\times$ 1200 px). |

| | |
|---|---|
| VA | Vertikale Anpassung. |
| VGA | Video Graphics Array (typisch $640$ px $\times$ $480$ px). |
| VoIP | Voice over Internet Protocol. |
| VQEG | Video Quality Experts Group. |
| WTA | Winner Takes All. |
| $YC_RC_B$ | Farbmodell - Helligkeit und Farbdifferenzsignale. |
| ZG | einfache zeitliche Glättung. |
| ZNCC | Zero-mean Normalized Cross-Correlation. |

# Literaturverzeichnis


[Adelson und Bergen 1991] Adelson, Edward H. ; Bergen, James R.: The Plenoptic Function and the Elements of Early Vision. In: Landy, Michael S. (Hrsg.) ; Movshon, J. A. (Hrsg.): *Computational models of visual processing*. Cambridge and Mass : MIT Press, 1991, S. 3–20. – ISBN 9780262121552

[Alatan u. a. 2007] Alatan, A. A. ; Yemez, Yucel ; Gudukbay, Ugur ; Zabulis, Xenophon ; Müller, Karsten ; Erdem, Cigdem E. ; Weigel, Christian ; Smolic, Aljoscha: Scene Representation Technologies for 3DTV - A Survey. In: *IEEE Transactions on Circuits and Systems for Video Technology* 17 (2007), Nr. 11, S. 1587–1605

[Anandan 1989] Anandan, P.: A computational framework and an algorithm for the measurement of visual motion. In: *International Journal of Computer Vision* 2 (1989), Nr. 3, S. 283–310

[Ansar u. a. 2004] Ansar, Adnan ; Castano, Andres ; Matthies, Larry: Enhanced real-time stereo using bilateral filtering. In: *International Symposium on 3D Data Processing, Visualization, and Transmission*. Los Alamitos and Calif : IEEE Computer Society, 2004, S. 455–462. – ISBN 9780769522234

[Argyle 1988] Argyle, Michael: *Bodily communication*. 2. London and and New York : Routledge, 1988. – ISBN 9780415051149

[Argyle und Dean 1965] Argyle, Michael ; Dean, Janet: Eye-Contact, Distance and Affiliation. In: *Sociometry* 28 (1965), Nr. 3, S. 289–304

[Argyle und Ingham 1972] Argyle, Michael ; Ingham, Roger: Gaze, Mutual Gaze, and Proximity. In: *Semiotica* 6 (1972), Nr. 1, S. 32–49

[Atzpadin u. a. 2004] Atzpadin, Nicole ; Kauff, Peter ; Schreer, Oliver: Stereo Analysis by Hybrid Recursive Matching for Real-Time Immersive Video Conferencing. In: *IEEE Transactions on Circuits and Systems for Video Technology* 14 (2004), Nr. 3, S. 321–334

[Avidan und Shashua 1997a] Avidan, Shai ; Shashua, Amnon: Novel view synthesis in tensor space. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE Computer Society, 1997, S. 1034–1040. – ISBN 9780818678240

[Avidan und Shashua 1997b] Avidan, Shai ; Shashua, Amnon ; The Hebrew University (Hrsg.): *Unifying Two-View and Three-View Geometry: Forschungsbericht.* 1997

[Avidan und Shashua 1998] Avidan, Shai ; Shashua, Amnon: Novel view synthesis by cascading trilinear tensors. In: *IEEE Transactions on Visualization and Computer Graphics* 4 (1998), Nr. 4, S. 293–306

[Bacon 1968] Bacon, W. S.: Amazing New Picturephone. In: *Popular Science* 192 (1968), Nr. 6, S. 46–47

[Baker und Binford Thomas 1981] Baker, H. H. ; Binford Thomas: Depth from Edge and Intensity Based Stereo. In: *International Joint Conference on Artificial Intelligence*, 1981, S. 631–636

[Beerends und Caluwe 1999] Beerends, John G. ; Caluwe, Frank E. d.: The Influence of Video Quality on Perceived Audio Quality and Vice Versa. In: *J. Audio Eng. Soc* 47 (1999), Nr. 5, S. 355–362. – URL http://www.aes.org/e-lib/browse.cfm?elib=12105

[Belhumeur 1996] Belhumeur, Peter N.: A Bayesian approach to binocular steropsis. In: *International Journal of Computer Vision* 19 (1996), Nr. 3, S. 237–260

[Belhumeur und Mumford 1992] Belhumeur, Peter N. ; Mumford, David: A Bayesian treatment of the stereo correspondence problem using half-occluded regions. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1992, S. 506–512. – ISBN 9780818628559

[Bell 1891] Bell, Alexander G.: *Editorial and Articles: On The Possibility Of Seeing By Electricity: unveröffentlichtes Manuskript.* 1891. – unveröffentlichtes Manuskript

[Besag 1986] Besag, Julian: On the Statistical Analysis of Dirty Pictures. In: *Journal of the Royal Statistical Society* Series B 48 (1986), Nr. 3, S. 259–302

[Birchfield und Tomasi 1998] Birchfield, Stan ; Tomasi, Carlo: A pixel dissimilarity measure that is insensitive to image sampling. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20 (1998), Nr. 4, S. 401–406

[Birchfield und Tomasi 1999] Birchfield, Stan ; Tomasi, Carlo: Depth Discontinuities by Pixel-to-Pixel Stereo. In: *International Journal of Computer Vision* 35 (1999), Nr. 3, S. 269–293

[Blais 2004] Blais, François: Review of 20 Years of Range Sensor Development. In: *Journal of Electronic Imaging* 13 (2004), Nr. 1, S. 231–243

[Bleyer und Gelautz 2005a] Bleyer, Michael ; Gelautz, Margrit: Graph-based surface reconstruction from stereo pairs using image segmentation. In: Beraldin, J.-Angelo (Hrsg.):

*Videometrics VIII*. Bellingham and Wash and Springfield and Va : SPIE and IS&T, 2005, S. 288–299. – ISBN 9780819456380

[Bleyer und Gelautz 2005b] Bleyer, Michael ; Gelautz, Margrit: A layered stereo matching algorithm using image segmentation and global visibility constraints. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 59 (2005), Nr. 3, S. 128–150. – ISSN 09242716

[Bobick und Intille 1999] Bobick, Aaron F. ; Intille, Stephen S.: Large Occlusion Stereo. In: *International Journal of Computer Vision* 33 (1999), Nr. 3, S. 181–200

[Bouguet 2010] Bouguet, Jean-Yves: *Camera Calibration Toolbox for Matlab*. 2010. – URL http://www.vision.caltech.edu/bouguetj/calib_doc/. – Zugriffsdatum: 01.09.2011

[Boykov u. a. 2001] Boykov, Yuri ; Veksler, Olga ; Zabih, Ramin: Fast approximate energy minimization via graph cuts. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23 (2001), Nr. 11, S. 1222–1239

[Brown 1966] Brown, Duane C.: Decentering Distortion of Lenses. In: *Photometric Engineering* 32 (1966), Nr. 3, S. 444–462

[Carranza u. a. 2003] Carranza, Joel ; Theobalt, Christian ; Magnor, Marcus A. ; Seidel, Hans-Peter: Free-viewpoint video of human actors. In: *ACM Transactions on Graphics* 22 (2003), Nr. 3, S. 569

[Chen 1995] Chen, Shenchang E.: QuickTime VR. In: *SIGGRAPH '95 Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*. New York and NY : ACM Press, 1995, S. 29–38. – ISBN 0897917014

[Chen und Williams 1993] Chen, Shenchang E. ; Williams, Lance: View interpolation for image synthesis. In: *SIGGRAPH '93 Proceedings of the 20th annual conference on Computer graphics and interactive techniques*. New York and NY : ACM Press, 1993, S. 279–288. – ISBN 0897916018

[Cheung u. a. 2005a] Cheung, Kong-man ; Baker, Simon ; Kanade, Takeo: Shape-From-Silhouette Across Time Part I: Theory and Algorithms. In: *International Journal of Computer Vision* 62 (2005), Nr. 3, S. 221–247

[Cheung u. a. 2005b] Cheung, Kong-man ; Baker, Simon ; Kanade, Takeo: Shape-From-Silhouette Across Time Part II: Applications to Human Modeling and Markerless Motion Tracking. In: *International Journal of Computer Vision* 63 (2005), Nr. 3, S. 225–245

[Clark 2005] Clark, Roger N.: *Digital Cameras: Does Pixel Size Matter? Factors in Choosing a Digital Camera (Does Sensor Size Matter?)*. 2005. – URL http://www.clarkvision.com/articles/does.pixel.size.matter/. – Zugriffsdatum: 03.03.2011

[Colburn u. a. 2000] Colburn, R. A. ; Cohen, Michael F. ; Drucker, Steven M. ; Microsoft Research (Hrsg.): *The Role of Eye Gaze in Avatar Mediated Conversational Interfaces: Technichal Report MSR-TR-2000-81.* 2000

[Comaniciu und Meer 2002] Comaniciu, Dorin ; Meer, Peter: Mean shift: a robust approach toward feature space analysis. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24 (2002), Nr. 5, S. 603–619. – ISSN 0162-8828

[Cox 1992] Cox, Ingemar J.: Stereo Without Disparity Gradient Smoothing: a Bayesian Sensor Fusion Solution. In: *Proceedings of the British Machine Vision Conference.* London [u.a.] : Springer, 1992, S. 337–346. – ISBN 9783540197775

[Cox u. a. 1996] Cox, Ingemar J. ; Hingorani, Sunita L. ; Rao, Satish B. ; Maggs, Bruce M.: A Maximum Likelihood Stereo Algorithm. In: *Computer Vision and Image Understanding* 63 (1996), Nr. 3, S. 542–567. – ISSN 10773142

[Criminisi u. a. 2007] Criminisi, Antonio ; Blake, Andrew ; Rother, Carsten ; Shotton, Jamie ; Torr, Philip H. S.: Efficient Dense Stereo with Occlusions for New View-Synthesis by Four-State Dynamic Programming. In: *International Journal of Computer Vision* 71 (2007), Nr. 1, S. 89–110

[Criminisi u. a. 2005] Criminisi, Antonio ; Blake, Andrew ; Torr, Philip H. S. ; Shotton, Jamie: *Virtual Camera Translation.* 2005

[Criminisi u. a. 2003] Criminisi, Antonio ; Shotton, Jamie ; Blake, Andrew ; Torr, Philip H. S.: Gaze manipulation for one-to-one teleconferencing. In: *IEEE International Conference on Computer Vision.* Los Alamitos and Calif : IEEE Computer Society, 2003, S. 191–198. – ISBN 9780769519500

[Critchlow und Fligner 1991] Critchlow, Douglas E. ; Fligner, Michael A.: On distribution-free multiple comparisons in the one-way analysis of variance. In: *Communications in Statistics - Theory and Methods* 20 (1991), Nr. 1, S. 127–139. – ISSN 0361-0926

[Cyganek und Siebert 2009] Cyganek, Boguslaw ; Siebert, J. P.: *An introduction to 3D computer vision techniques and algorithms.* Chichester : Wiley, 2009. – ISBN 978-0-470-01704-3

[Décoret u. a. 2003] Décoret, Xavier ; Durand, Frédo ; Sillion, François X. ; Dorsey, Julie: Billboard clouds for extreme model simplification. In: *ACM Transactions on Graphics* 22 (2003), Nr. 3, S. 689–696

[Dempster u. a. 1977] Dempster, A. P. ; Laird, N. M. ; Rubin, D. B.: Maximum likelihood from incomplete data via the EM algorithm. In: *JOURNAL OF THE ROYAL STATISTICAL SOCIETY, SERIES B* 39 (1977), Nr. 1, S. 1–38

[Deng u. a. 2005] Deng, Yi ; Yang, Qiong ; Lin, Xueyin ; Tang, Xiaoou: A symmetric patch-based correspondence model for occlusion handling. In: *IEEE International Conference on Computer Vision*. Los Almamitos and Calif : IEEE Computer Society, 2005, S. 1316–1322 Vol. 2. – ISBN 9780769523347

[Divorra 2011] Divorra, Oscar: *3D Presence: The 3D Telepresence and 3D-Aware Next Generation Immersive Videoconferencing Project*. 2011. – URL www.3dpresence.eu/. – Zugriffsdatum: 09.12.2011

[Döhring 2009] Döhring, Rene: *Evaluierung von Verfahren zur Stereo-Korrespondenzanalyse und Untersuchung ausgewählter Ansätze hinsichtlich deren Parallelisierbarkeit auf Basis von Cuda*. Ilmenau, TU Ilmenau, Diplomarbeit, 2009

[Douglas und Peucker 1973] Douglas, D. H. ; Peucker, T. K.: Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. In: *Cartographica The International Journal for Geographic Information and Geovisualization* 10 (1973), Nr. 2, S. 112–122

[Dumont u. a. 2008] Dumont, Maarten ; Maesen, Steven ; Rogmans, Sammy ; Bekaert, Philippe: A Prototype for Practical Eye-Gaze Corrected Video Chat on Graphics Hardware. In: Assuncao, Pedro (Hrsg.) ; Faria Sergio (Hrsg.): *SIGMAP 2008*, 2008, S. 236–243

[Dumont u. a. 2009] Dumont, Maarten ; Rogmans, Sammy ; Maesen, Steven ; Bekaert, Philippe: Optimized Two-Party Video Chat with Restored Eye Contact Using Graphics Hardware. In: *International Conference e-Business and Telecommunications* Bd. 48. Berlin and Heidelberg : Springer Berlin Heidelberg, 2009, S. 358–372. – ISBN 978-3-642-05196-8

[van Eijk u. a. 2010] Eijk, Rob van ; Kuijsters, Andre ; Dijkstra, Klaske ; IJsselsteijn, Wijnand A.: Human Sensitivity to Eye Contact in 2D and 3D Videoconferencing. In: *Second International Workshop on Quality of Multimedia Experience (QoMEX)*. Piscataway and NJ : IEEE, 2010, S. 76–81

[Ellgring 1995] Ellgring, Heiner: Nonverbale Kommunikation: Einführung und Überblick. In: Rosenbusch, Heinz S. (Hrsg.) ; Schober, Otto (Hrsg.): *Körpersprachen in der schulischen Erziehung*. Baltmannsweiler : Schneider-Verl. Hohengehren, 1995, S. 7–48. – ISBN 9783871169724

[Faugeras u. a. 2001] Faugeras, Olivier ; Luong, Quang-Tuan ; Papadopoulo, Théo: *The geometry of multiple images: The laws that govern the formation of multiple images of a scene and some of their applications*. Cambridge and Mass : MIT Press, 2001. – ISBN 0262062208

[Feldmann u. a. 2009a] Feldmann, Ingo ; Atzpadin, Nicole ; Schreer, Oliver ; Pujol-Acolado, Jose-Carlos ; Landabaso, Jose L. ; Divorra, Oscar: Multi-view depth estimation based on visual-hull enhanced Hybrid Recursive Matching for 3D video conference systems. In: *IEEE International Conference on Image Processing*, IEEE, 2009, S. 745–748. – ISBN 1424456541

[Feldmann u. a. 2009b] Feldmann, Ingo ; Schreer, Oliver ; Schäfer, Ralf ; Zuo, Fei ; Belt, Harm ; Divorra, Oscar: Immersive Multi-User 3D Video Communication. In: *Proc. of International Broadcast Conference (IBC 2009)*, 2009, S. –

[Feldmann u. a. 2010] Feldmann, Ingo ; Waizenegger, Wolfgang ; Atzpadin, Nicole ; Schreer, Oliver: Real-time depth estimation for immersive 3D videoconferencing. In: *3DTV Conference*. [Piscataway and N.J.] : IEEE, 2010, S. 1–4. – ISBN 1424463785

[Felzenszwalb und Huttenlocher 2006] Felzenszwalb, Pedro F. ; Huttenlocher, Daniel P.: Efficient Belief Propagation for Early Vision. In: *International Journal of Computer Vision* 70 (2006), Nr. 1, S. 41–54

[Fischler und Bolles 1981] Fischler, Martin A. ; Bolles, Robert C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. In: *Communications of the ACM* 24 (1981), Nr. 6, S. 381–395. – ISSN 00010782

[Foley u. a. 1991] Foley, James D. ; Dam, Andries van ; Feiner, Steven K. ; Hughes, John F.: *Computer graphics: Principles and practice.* 2. Reading and Mass : Addison-Wesley, 1991. – ISBN 9780201121100

[Fua 1993] Fua, Pascal: A parallel stereo algorithm that produces dense depth maps and preserves image features. In: *Machine Vision and Applications* 6 (1993), Nr. 1, S. 35–49–49. – ISSN 0932-8092

[Gamble und Poggio 1987] Gamble, Ed ; Poggio, Tomaso: *Visual Integration and Detection of Discontinuities: The Key Role of Intensity Edges: MIT AI Technical Report.* 1987

[Garau u. a. 2001] Garau, Maia ; Slater, Mel ; Bee, Simon ; Sasse, Martina A.: The impact of eye gaze on communication using humanoid avatars. In: *SIGCHI conference on Human factors in computing systems.* New York : ACM Press, 2001, S. 309–316. – ISBN 1581133278

[Geiger u. a. 2010] Geiger, Andreas ; Roser, Martin ; Urtasun, Raquel: Efficient Large-Scale Stereo Matching. In: *Asian Conference on Computer Vision*, 2010, S. 25–38

[Geiger u. a. 1995] Geiger, Davi ; Ladendorf, Bruce ; Yuille, Alan: Occlusions and binocular stereo. In: *International Journal of Computer Vision* 14 (1995), Nr. 3, S. 211–226

[Geman und Geman 1984] Geman, Stuart ; Geman, Donald: Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-6 (1984), Nr. 6, S. 721–741

[Gemmell u. a. 2000] Gemmell, Jim ; Toyama, Kentaro ; Zitnick, C.Lawrence ; Kang, Thomas ; Seitz, Steven: Gaze awareness for video-conferencing: a software approach. In: *IEEE Multimedia* 7 (2000), Nr. 4, S. 26–35. – ISSN 1070986X

[Gokturk u. a. 2004] Gokturk, S. B. ; Yalcin, Hakan ; Bamji, Cyrus: A Time-Of-Flight Depth Sensor - System Description, Issues and Solutions. In: *Computer Vision and Pattern Recognition Workshop* 3 (2004), S. 35. – ISSN 1063-6919

[Gortler u. a. 1996] Gortler, Steven J. ; Grzeszczuk, Radek ; Szeliski, Richard ; Cohen, Michael F.: The lumigraph. In: *SIGGRAPH '96 Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. New York and NY : ACM Press, 1996, S. 43–54. – ISBN 0897917469

[Grau 2003] Grau, Oliver: Studio production system for dynamic 3D content. In: *Visual Communications and Image Processing* Bd. 5150, 2003, S. 80–89

[Gulich 1998] Gulich, Andre: *Video-Konferenztechnik: Alles Wissenswerte über die Videokonferenztechnik und die Videokommunikation über das öffentliche Telefonnetz und das Internet : mit 28 Tabellen*. Poing : Franzis, 1998. – ISBN 3-7723-6864-6

[Hartley und Zisserman 2008] Hartley, Richard ; Zisserman, Andrew: *Multiple view geometry in computer vision*. 2nd ed., 6th printing. Cambridge : Cambridge University Press, 2008. – ISBN 0521540518

[Hartley 1999] Hartley, Richard I.: Theory and Practice of Projective Rectification. In: *International Journal of Computer Vision* 35 (1999), Nr. 2, S. 115–127

[Heikkila und Silven 1997] Heikkila, Janne ; Silven, Olli: A four-step camera calibration procedure with implicit image correction. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE Computer Society, 1997, S. 1106–1112. – ISBN 9780818678240

[Hirschmüller und Scharstein 2009] Hirschmüller, Heiko ; Scharstein, Daniel: Evaluation of Stereo Matching Costs on Images with Radiometric Differences. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31 (2009), Nr. 9, S. 1582–1599

[Holst 1998] Holst, Gerald C.: *CCD arrays, cameras, and displays*. 2nd. Winter Park and FL and Bellingham and Wash. and USA : JCD Pub. and SPIE Optical Engineering, 1998. – ISBN 9780819428530

[Hong und Chen 2004] Hong, Li ; Chen, George: Segment-based stereo matching using graph cuts. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, 2004, S. 74–81. – ISBN 0-7695-2158-4

[Hörchens 2005] Hörchens, Lars: *Segmentation of Video Sequences for Compositing Applications in Television Production*. Ilmenau, TU Ilmenau, Diplomarbeit, 2005

[Ishii und Kobayashi 1992] Ishii, Hiroshi ; Kobayashi, Minoru: ClearBoard: a Seamless Medium for Shared Drawing and Conversation with Eye Contact. In: *SIGCHI conference on Human factors in computing systems*. New York : ACM Press, 1992, S. 525–532. – ISBN 0897915143

[ITU-R 2012] ITU-R: *BT.500(01/2012): Methodology for the subjective assessment of the quality of television pictures.* 2012

[ITU-T 1993] ITU-T: *H.261 : Video codec for audiovisual services at p x 64 kbit/s.* 1993

[ITU-T 1996a] ITU-T: *H.263 : Video coding for low bit rate communication.* 1996

[ITU-T 1996b] ITU-T: *H.323 : Visual telephone systems and equipment for local area networks which provide a non-guaranteed quality of service.* 1996

[ITU-T 2008] ITU-T: *P.910: Subjective video quality assessment methods for multimedia applications.* 2008

[ITU-T 2011] ITU-T: *J.341: Objective perceptual multimedia video quality measurement of HDTV for digital cable television in the presence of a full reference.* 2011

[Jähne 2005] Jähne, Bernd: *Digitale Bildverarbeitung.* 6., überarb. und erw. Berlin and and Heidelberg and and New York : Springer, 2005. – ISBN 9783540249993

[Jiang und Bunke 1997] Jiang, Xiaoyi ; Bunke, Horst: *Dreidimensionales Computersehen: Gewinnung und Analyse von Tiefenbildern ; mit 3 Tafeln.* Berlin [u.a.] : Springer, 1997. – ISBN 9783540607977

[Kanade u. a. 1995] Kanade, Takeo ; Kano, Hiroshi ; Kimura, Shigeru ; Yoshida, Atsushi ; Oda, Kazuo: Development of a video-rate stereo machine. In: *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*, IEEE Comput. Soc. Press, 1995, S. 95–100. – ISBN 0-8186-7108-4

[Kanade und Okutomi 1994] Kanade, Takeo ; Okutomi, Masatoshi: A stereo matching algorithm with an adaptive window: theory and experiment. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16 (1994), Nr. 9, S. 920–932

[Kang u. a. 2001] Kang, Sing B. ; Szeliski, Richard ; Chai, Jinxiang: Handling Occlusions in Dense Multi-View Stereo. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2001, S. I–103–I–110

[Kass u. a. 1988] Kass, Michael ; Witkin, Andrew ; Terzopoulos, Demetri: Snakes: Active contour models. In: *International Journal of Computer Vision* 1 (1988), Nr. 4, S. 321–331

[Kauff und Schreer 2002] Kauff, Peter ; Schreer, Oliver: An immersive 3D video-conferencing system using shared virtual team user environments. In: *Proceedings of the 4th International Conference on Collaborative Virtual Environments.* New York and NY : ACM Press, 2002, S. 105–112. – ISBN 9781581134896

[Kendon 1967] Kendon, Adam: Some functions of gaze-direction in social interaction. In: *Acta Psychologica* 26 (1967), S. 22–63. – ISSN 00016918

[Kepplinger vorr. 2014] Kepplinger, Sara: *Quality taxonomy for scaleable algorithms of 3D video objects.* Ilmenau, TU Ilmenau, Dissertation (noch unveröffentlicht), vorr. 2014

[Kilner u. a. 2009] Kilner, Joe ; Starck, Jonathan ; Guillemaut, Jean-Yves ; Hilton, Adrian: Objective quality assessment in free-viewpoint video production. In: *Signal Processing: Image Communication* 24 (2009), Nr. 1-2, S. 3–16. – ISSN 09235965

[Klaus u. a. 2006] Klaus, Andreas ; Sormann, Mario ; Karner, Konrad: Segment-Based Stereo Matching Using Belief Propagation and a Self-Adapting Dissimilarity Measure. In: *International Conference on Pattern Recognition.* Los Alamitos and California [etc.] : IEEE Computer Society Press, 2006, S. 15–18. – ISBN 9780769525211

[Kleinke 1986] Kleinke, Chris L.: Gaze and eye contact: A research review. In: *Psychological Bulletin* 100 (1986), Nr. 1, S. 78–100. – ISSN 1939-1455

[Kolmogorov und Zabih 2001] Kolmogorov, Vladimir ; Zabih, Ramin: Computing Visual Correspondence with Occlusions via Graph Cuts. In: *IEEE International Conference on Computer Vision.* Los Alamitos and Calif : IEEE Computer Society, 2001, S. 508–515. – ISBN 9780769511450

[Kolmogorov und Zabih 2002] Kolmogorov, Vladimir ; Zabih, Ramin: Multi-camera Scene Reconstruction via Graph Cuts. In: *European Conference on Computer Vision* Bd. 3. Berlin and Heidelberg : Springer Berlin Heidelberg, 2002, S. 82–96. – ISBN 978-3-540-43744-4

[Kolmogorov und Zabih 2004] Kolmogorov, Vladimir ; Zabih, Ramin: What Energy Functions can be Minimized via Graph Cuts. In: *EEE Transactions on Pattern Analysis and Machine Intelligence* 26 (2004), S. 65–81

[Korn 2009] Korn, Torsten: *Kalibrierung und Blickrichtungsanalyse für ein 3D Videokonferenzsystem.* Ilmenau, TU Ilmenau, Diplomarbeit, 2009

[Kreibich 2005] Kreibich, Leif L.: *Analyse und Umsetzung alternativer Ansätze zur erweiterten Ansicht von 3D Videoobjekten.* Ilmenau, TU Ilmenau, Diplomarbeit, 2005

[Kruskal und Wallis 1952] Kruskal, William H. ; Wallis, W. A.: Use of Ranks in One-Criterion Variance Analysis. In: *Journal of the American Statistical Association* 47 (1952), Nr. 260, S. 583. – ISSN 01621459

[Kuster u. a. 2012] Kuster, Claudia ; Popa, Tiberiu ; Bazin, Jean-Charles ; Gotsman, Craig ; Gross, Markus: Gaze correction for home video conferencing. In: *ACM Transactions on Graphics* 31 (2012), Nr. 6, S. 1. – ISSN 07300301

[Laveau und Faugeras 1994] Laveau, Stephane ; Faugeras, Olivier: 3-D scene representation as a collection of images. In: *International Conference on Pattern Recognition*, 1994, S. 689–691

[Lei und Hendriks 2001] Lei, Bang J. ; Hendriks, Emile A.: Multi-step View Synthesis with Occlusion Handling. In: *Vision, modeling, and visualization 2001*. Berlin and Amsterdam : AKA and IOS Press, 2001, S. 185–192. – ISBN 1586032216

[Lei und Hendriks 2002] Lei, Bang J. ; Hendriks, Emile A.: Real-Time Multi-Step View Reconstruction for a Virtual Teleconference System. In: *EURASIP Journal on Advances in Signal Processing* 2002 (2002), Nr. 10, S. 1067–1087

[Levoy 2000] Levoy, Marc: *An excerpt from the center of a light field*. 2000. – URL http://graphics.stanford.edu/projects/dli/white-paper/dragon_uvplane-onwhite.jpg. – Zugriffsdatum: 29.11.2011

[Levoy und Hanrahan 1996] Levoy, Marc ; Hanrahan, Pat: Light field rendering. In: *SIGGRAPH '96 Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. New York and NY : ACM Press, 1996, S. 31–42. – ISBN 0897917469

[Lewis u. a. 1994] Lewis, John P. ; Ott, Maximilian ; Cox, Ingemar J.: *Videoconference System Using a Virtual camera image*. 1994

[Lhuillier und Quan 1999] Lhuillier, Maxime ; Quan, Long: Image interpolation by joint view triangulation. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1999, S. 139–145. – ISBN 0769501494

[Lhuillier und Quan 2002] Lhuillier, Maxime ; Quan, Long: Match propagation for image-based modeling and rendering. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24 (2002), Nr. 8, S. 1140–1146

[Lhuillier und Quan 2003] Lhuillier, Maxime ; Quan, Long: Image-based rendering by joint view triangulation. In: *IEEE Transactions on Circuits and Systems for Video Technology* 13 (2003), Nr. 11, S. 1051–1063

[Li und Bovik 2010] Li, Chaofeng ; Bovik, Alan C.: Content-weighted video quality assessment using a three-component image model. In: *Journal of Electronic Imaging* 19 (2010), Nr. 1, S. 011003

[Li 2009] Li, Stan Z.: *Markov random field modeling in image analysis*. 3rd. London : Springer, 2009. – ISBN 1848002793

[Lipski u. a. 2011] Lipski, Christian ; Klose, Felix ; Ruhl, Kai ; Magnor, Marcus A.: The virtual video camera: Simplified 3DTV acquisition and processing. In: *3DTV Conference*. [Piscataway and N.J.] : IEEE, 2011, S. 1–4. – ISBN 1612841619

[Lipski u. a. 2010a] Lipski, Christian ; Linz, Christian ; Berger, Kai ; Sellent, Anita ; Magnor, Marcus: Virtual Video Camera: Image-Based Viewpoint Navigation Through Space and Time. In: *Computer Graphics Forum* 29 (2010), Nr. 8, S. 2555–2568

[Lipski u. a. 2010b] Lipski, Christian ; Linz, Christian ; Neumann, Thomas ; Wacker, Markus ; Magnor, Marcus: High Resolution Image Correspondences for Video Post-Production. In: *Proc. European Conference on Visual Media Production (CVMP) 2010* Bd. 7. Los Alamitos and CA and USA : IEEE Computer Society, 2010, S. 33–39

[Liu u. a. 1995] Liu, Jin ; Beldie, Ion P. ; Wöpking, Matthias: A Computational Approach To Establish Eye-Contact In Videocommunication. In: *Int. Workshop on Stereoscopic and Three Dimentional Imaging*, 1995, S. 229–234

[Liu und Huang 1993] Liu, Jin ; Huang, Shaoguang: Using topological information of images to improve stereo matching. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Washington : IEEE Computer Society Press, 1993, S. 653–654. – ISBN 9780818638824

[Lowe 2004] Lowe, David G.: Distinctive Image Features from Scale-Invariant Keypoints. In: *International Journal of Computer Vision* 60 (2004), Nr. 2, S. 91–110

[Lucas und Kanade 1981] Lucas, Bruce D. ; Kanade, Takeo: An Iterative Image Registration Technique with an Application to Stereo Vision. In: *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*. Los Altos and Calif : William Kaufmann, 1981, S. 674–679. – ISBN 9780865760592

[Marr und Poggio 1976] Marr, David ; Poggio, Tomaso: Cooperative Computation of Stereo Disparity. In: *Science* 194 (1976), Nr. 4262, S. 283–287. – ISSN 0036-8075

[McMillan und Bishop 1995] McMillan, Leonard ; Bishop, Gary: Plenoptic modeling. In: *SIGGRAPH '95 Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*. New York and NY : ACM Press, 1995, S. 39–46. – ISBN 0897917014

[Merkle u. a. 2007a] Merkle, Philipp ; Smolic, Aljoscha ; Müller, Karsten ; Wiegand, Thomas: Efficient Prediction Structures for Multiview Video Coding. In: *IEEE Transactions on Circuits and Systems for Video Technology* 17 (2007), Nr. 11, S. 1461–1473

[Merkle u. a. 2007b] Merkle, Philipp ; Smolic, Aljoscha ; Muller, Karsten ; Wiegand, Thomas: Multi-View Video Plus Depth Representation and Coding. In: *IEEE International Conference on Image Processing*, IEEE, 2007, S. I – 201–I – 204. – ISBN 9781424414376

[Möllering und Slansky 1993] Möllering, Detlef ; Slansky, Peter C.: *Handbuch der professionellen Videoaufnahme: mit 44 Tabellen*. 2. Essen : Edition Filmwerkstatt, 1993. – ISBN 9783980258135

[Molnar 1969] Molnar, Julius P. ; Bell Laboratories (Hrsg.): *Bell Laboratories Record: PICTUREPHONE Service-A New Way of Communicating*. 1969

[Moorthy und Bovik 2010] Moorthy, Anush K. ; Bovik, Alan C.: Efficient motion weighted spatio-temporal video SSIM index. In: *Human Vision and Electronic Imaging XV*, 2010, S. –. – ISBN 9780819479204

[Mukai u. a. 2009] Mukai, Shigeki ; Murayama, Daisuke ; Kimura, Keiichi ; Hosaka, Tadaaki ; Hamamoto, Takayuki ; Shibuhisa, Nao ; Tanaka, Seiichi ; Sato, Shunichi ; Saito, Sakae: Arbitrary view generation for eye-contact communication using projective transformations. In: *Proceedings of the 8th International Conference on Virtual Reality Continuum and its Applications in Industry*, ACM Press, 2009, S. 305–306. – ISBN 9781605589121

[Mukawa u. a. 2005] Mukawa, Naoki ; Oka, Tsugumi ; Arai, Kumiko ; Yuasa, Masahide: What is connected by mutual gaze? User's Behavior in Video-mediated Communication. In: *SIGCHI conference on Human factors in computing systems*. New York : ACM Press, 2005, S. 1677–1680. – ISBN 1581139985

[Murayama u. a. 2010] Murayama, Daisuke ; Kimura, Keiichi ; Hosaka, Tadaaki ; Hamamoto, Takayuki ; Shibuhisa, Nao ; Tanaka, Seiichi ; Sato, Shunichi ; Saito, Sakae: Virtual view image synthesis for eye-contact in TV conversation system. In: *Three-Dimensional Image Processing (3DIP) and Applications*, 2010, S. –. – ISBN 9780819479198

[Ohta und Kanade 1985] Ohta, Yuichi ; Kanade, Takeo: Stereo by Intra- and Inter-Scanline Search Using Dynamic Programming. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 7 (1985), Nr. 2, S. 139–154

[Okada u. a. 1994] Okada, Ken-Ichi ; Maeda, Fumihiko ; Ichikawaa, Yusuke ; Matsushita, Yutaka: Multiparty Videoconferencing at Virtual Social Distance. In: *Conference on Computer Supported Cooperative Work*. New York : ACM Press, 1994, S. 385–393. – ISBN 9780897916899

[Okutomi und Kanade 1992] Okutomi, Masatoshi ; Kanade, Takeo: A locally adaptive window for signal matching. In: *International Journal of Computer Vision* 7 (1992), Nr. 2, S. 143–162

[Oppenheimer 1959] Oppenheimer, Jess: *Prompting Apparatus*. 1959

[Oppenheimer 1960] Oppenheimer, Jess: *Prompting Apparatus for Cameras*. 1960

[Ott u. a. 1993] Ott, Maximilian ; Lewis, John P. ; Cox, Ingemar J.: Teleconferencing eye contract using a virtual camera. In: *International Conference on Human-Computer Interaction INTERCHI '93*. New York and Reading and MA : Association for Computing Machinery and Addison-Wesley, 1993, S. 109–110. – ISBN 9780201588842

[Otto und Chau 1989] Otto, G. P. ; Chau, T. K.: A Region-growing algorithm for matching of terrain images. In: *Image and Vision Computing* 7 (1989), Nr. 2, S. 83–94. – ISSN 02628856

[Owens 2005] Owens, John: Streaming architectures and technology trends. In: Pharr, Matt (Hrsg.) ; Fernando, Randima (Hrsg.): *GPU gems 2.* Upper Saddle River and NJ : Addison-Wesley, 2005, S. 9. – ISBN 9780321335593

[Owens u. a. 2007] Owens, John D. ; Luebke, David ; Govindaraju, Naga ; Harris, Mark ; Krüger, Jens ; Lefohn, Aaron E. ; Purcell, Timothy J.: A Survey of General-Purpose Computation on Graphics Hardware. In: *Computer Graphics Forum* 26 (2007), Nr. 1, S. 80–113. – ISSN 0167-7055

[Pearl 1988] Pearl, Judea: *Probabilistic reasoning in intelligent systems: Networks of plausible inference.* Rev. 2nd print. San Francisco and Calif : Morgan Kaufmann, 1988. – ISBN 9781558604797

[Pei u. a. 2011] Pei, Yin ; Criminisi, Antonio ; Winn, John ; Essa, Irfan: Bilayer Segmentation of Webcam Videos Using Tree-Based Classifiers. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33 (2011), Nr. 1, S. 30–42

[Pollard u. a. 1986] Pollard, Stephen B. ; Porrill John ; Mayhew, John E. W. ; Frisby John P.: Disparity Gradient, Lipschitz Continuity, and Computing Binocular Correspondences. In: *The 3. International Symposium on Robotics Research.* Cambridge M : MIT Press, 1986, S. 19–26. – ISBN 9780262061018

[Power und Schoonees 2002] Power, P. W. ; Schoonees, Johann A.: Understanding Background Mixture Models for Foreground Segmentation Understanding Background Mixture Models for Foreground Segmentation. In: *Image and Vision Computing* 2002 (2002), Nr. November, S. 267–271

[Rittermann 2004] Rittermann, Marco: A Proposal for the Quality Assessment of 3D Video Objects. In: *Proceedings of the 5th Workshop on Image Analysis for Multimedia Interactive Services*, 2004, S. –

[Rittermann 2007] Rittermann, Marco: *Zur Qualitätsbeurteilung von 3D-Videoobjekten.* Ilmenau, TU Ilmenau, Dissertation, 2007

[Scharstein 1994] Scharstein, Daniel: Matching Images by Comparing their Gradient Fields. In: *International Conference on Pattern Recognition*, 1994, S. 572–575

[Scharstein 1999] Scharstein, Daniel: *View Synthesis Using Stereo Vision.* Berlin and New York : Springer, 1999. – ISBN 9783540661597

[Scharstein und Szeliski 2002] Scharstein, Daniel ; Szeliski, Richard: A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms. In: *International Journal of Computer Vision* 47 (2002), Nr. 1-3, S. 7–42

[Scharstein und Szeliski 2012] Scharstein, Daniel ; Szeliski, Richard: *Middlebury Stereo Vision Page*. 2012. – URL http://vision.middlebury.edu/stereo/. – Zugriffsdatum: 16.01.2012

[Schmidt 2009] Schmidt, Heinz U.: *Professionelle Videotechnik: Grundlagen, Filmtechnik, Fernsehtechnik, Geräte- und Studiotechnik in SD, HD, DI, 3D*. 5., aktualisierte und erw. Berlin and Heidelberg : Springer, 2009. – ISBN 3642025072

[Schmidt 2010] Schmidt, Julia: *Bildsegmentierung für ein 3D Videokonferenzsystem*. Ilmenau, TU Ilmenau, Diplomarbeit, 2010

[Schreer u. a. 2009] Schreer, Oliver ; Atzpadin, Nicole ; Feldmann, Ingo ; Kauff, Peter: Multi-baseline Disparity Fusion for Immersive Videoconferencing. In: *International Conference on Immersive Telecommunications*, 2009, S. Articel No. 4. – ISBN 9789639799394

[Schreer u. a. 2001] Schreer, Oliver ; Brandenburg, Nicole ; Kauff, Peter: Real-time disparity analysis for applications in immersive teleconference scenarios-a comparative study. In: *11th International Conference on Image Analysis and Processing*, 2001, S. 346–351. – ISBN 0-7695-1183-X

[Schreer u. a. 2005] Schreer, Oliver ; Kauff, Peter ; Sikora, Thomas: *3D videocommunication: Algorithms, concepts, and real-time systems in human centred communication*. Chichester and England and Hoboken and NJ : Wiley, 2005. – ISBN 9780470022719

[Segal und Akeley 2010] Segal, Mark ; Akeley, Kurt ; The Khronos Group Inc. (Hrsg.): *The OpenGL Graphics System: A Specification (Version 3.3 (Core Profile) - March 11, 2010)*. 2010

[Seitz u. a. 2006] Seitz, Steven M. ; Curless, Brian ; Diebel, James ; Scharstein, Daniel ; Szeliski, Richard: A Comparison and Evaluation of Multi-View Stereo Reconstruction Algorithms. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2006, S. 519–528

[Seitz und Dyer 1996] Seitz, Steven M. ; Dyer, Charles R.: View morphing. In: *SIGGRAPH '96 Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. New York and NY : ACM Press, 1996, S. 21–30. – ISBN 0897917469

[Shade u. a. 1998] Shade, Jonathan ; Gortler, Steven ; He, Li-Wei ; Szeliski, Richard: Layered depth images. In: *SIGGRAPH '98 Proceedings of the 25th annual conference on Computer graphics and interactive techniques*. New York and NY : ACM Press, 1998, S. 231–242. – ISBN 0897919999

[Shashua 1997] Shashua, Amnon: Trilinear tensor: The fundamental construct of multiple-view geometry and its applications. In: *International workshop on Algebraic frames for the perception-action cycle*. Berlin and New York : Springer, 1997, S. 190–206. – ISBN 9783540635178

[Shashua und Riklin-Raviv 2001] Shashua, Amnon ; Riklin-Raviv, Tammy: The quotient image: class-based re-rendering and recognition with varying illuminations. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23 (2001), Nr. 2, S. 129–139

[Shi und Tomasi 1994] Shi, Jianbo ; Tomasi, Carlo: Good features to track. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Los Alamitos and California [etc.] : IEEE Computer Society Press, 1994, S. 593–600. – ISBN 9780818658266

[Shum u. a. 2007] Shum, Heung-Yeung ; Chan, Shing-Chow ; Kang, Sing B.: *Image-based rendering*. New York : Springer, 2007. – ISBN 0387211136

[Shum und He 1999] Shum, Heung-Yeung ; He, Li-Wei: Rendering with concentric mosaics. In: *SIGGRAPH '99 Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. New York and NY : ACM Press, 1999, S. 299–306. – ISBN 0201485608

[Shum u. a. 2003] Shum, Heung-Yeung ; Kang, Sing B. ; Chan, Shing-Chow: Survey of image-based representations and compression techniques. In: *IEEE Transactions on Circuits and Systems for Video Technology* 13 (2003), Nr. 11, S. 1020–1037

[Smith und Blinn 1996] Smith, Alvy R. ; Blinn, James F.: Blue screen matting. In: *SIGGRAPH '96 Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. New York and NY : ACM Press, 1996, S. 259–268. – ISBN 0897917469

[Smolic u. a. 2004] Smolic, Aljoscha ; Mueller, Karsten ; Merkle, Philipp ; Rein, Tobias ; Kautzner, Matthias ; Eisert, Peter ; Wiegand, Thomas: Free viewpoint video extraction, representation, coding, and rendering. In: *IEEE International Conference on Image Processing*, 2004, S. 3287–3290

[Starck und Hilton 2007] Starck, Jonathan ; Hilton, Adrian: Surface Capture for Performance-Based Animation. In: *IEEE Computer Graphics and Applications* 27 (2007), Nr. 3, S. 21–31

[Starck u. a. 2008] Starck, Jonathan ; Kilner, Joe ; Hilton, Adrian: Objective Quality Assessment in Free-Viewpoint Video Production. In: *3DTV Conference*. Piscataway : IEEE, 2008, S. 225–228. – ISBN 1424417600

[Stauffer und Grimson 1999] Stauffer, Chris ; Grimson, W.E.L: Adaptive background mixture models for real-time tracking. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1999, S. 246–252. – ISBN 0769501494

[Stefanoski und Ostermann 2008] Stefanoski, Nikolce ; Ostermann, Jörn: Spatially and temporally scalable compression of animated 3D meshes with MPEG-4 / FAMC. In: *IEEE International Conference on Image Processing*, IEEE, 2008, S. 2696–2699. – ISBN 1424417643

[Strohmeier 2011] Strohmeier, Dominik: *Open profiling of quality a mixed methods research approach for audiovisual quality evaluations*. Ilmenau, TU Ilmenau, Dissertation, 2011

[Sun u. a. 2003] Sun, Jian ; Zheng, Nan-Ning ; Shum, Heung-Yeung: Stereo matching using belief propagation. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25 (2003), Nr. 7, S. 787–800

[Szeliski 1996] Szeliski, Richard: Video mosaics for virtual environments. In: *IEEE Computer Graphics and Applications* 16 (1996), Nr. 2, S. 22–30

[Szeliski 2011] Szeliski, Richard: *Computer vision: Algorithms and applications*. London : Springer, 2011. – ISBN 1848829345

[Szeliski und Golland 1999] Szeliski, Richard ; Golland, Polina: Stereo Matching with Transparency and Matting. In: *International Journal of Computer Vision* 32 (1999), Nr. 1, S. 45–61

[Szeliski und Shum 1997] Szeliski, Richard ; Shum, Heung-Yeung: Creating full view panoramic image mosaics and environment maps. In: *SIGGRAPH '97 Proceedings of the 24th annual conference on Computer graphics and interactive techniques*. New York and NY : ACM Press, 1997, S. 251–258. – ISBN 0897918967

[Tang 2007] Tang, John C.: Approaching and leave-taking: Negotiating Contact in Computer-Mediated Communication. In: *ACM Transactions on Computer-Human Interaction* 14 (2007), Nr. 1, S. 5–es

[Tao u. a. 2001] Tao, Hai ; Sawhney, Harpreet S. ; Kumar, Rakesh: A global matching framework for stereo computation. In: *IEEE International Conference on Computer Vision*. Los Alamitos and Calif : IEEE Computer Society, 2001, S. 532–539. – ISBN 9780769511450

[Tappen und Freeman 2003] Tappen, Marshall F. ; Freeman, William T.: Comparison of graph cuts with belief propagation for stereo, using identical MRF parameters. In: *IEEE International Conference on Computer Vision*. Los Alamitos and Calif : IEEE Computer Society, 2003, S. 900–906 vol.2. – ISBN 9780769519500

[Telea 2004] Telea, Alexandru: An Image Inpainting Technique Based on the Fast Marching Method. In: *journal of graphics, gpu, and game tools* 9 (2004), Nr. 1, S. 25–36

[Tomasi und Manduchi 1998] Tomasi, Carlo ; Manduchi, Roberto: Bilateral filtering for gray and color images. In: *IEEE International Conference on Computer Vision*. New Delhi and [New York] : Narosa Pub. House and Institute for Electrical and Electronics Engineering, 1998, S. 839–846. – ISBN 9780780350991

[Tombari u. a. 2007] Tombari, Federico ; Mattoccia, Stefano ; Stefano, Luigi: Segmentation-Based Adaptive Support for Accurate Stereo Correspondence. In: *Pacific Rim Conference on Advances in Image and Video Technology*, 2007, S. 427–438

[Treutner 2010] Treutner, Niklas: *Blickkorrektur bei Videokonferenzen mittels Stereoanalyse*. Ilmenau, TU Ilmenau, Diplomarbeit, 2010

[Tsai 1987] Tsai, Roger Y.: A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. In: *IEEE Journal on Robotics and Automation* 3 (1987), Nr. 4, S. 323–344

[Unbekannter Autor 1971] Unbekannter Autor: French Tete-a-Tete on Face-to-Face Phone. In: *New Scientist* (1971), S. 140

[Unbekannter Autor 2011] Unbekannter Autor: *History of AT&T and Television*. 2011. – URL http://www.corp.att.com/history/television/. – Zugriffsdatum: 02.11.2011

[USB Implementers Forum 27.04.2000] USB Implementers Forum: *Universal Serial Bus Specification*. 27.04.2000

[Vedula u. a. 2005] Vedula, Sundar ; Baker, Simon ; Kanade, Takeo: Image-based spatio-temporal modeling and view interpolation of dynamic events. In: *ACM Transactions on Graphics* 24 (2005), Nr. 2, S. 240–261

[Vertegaal u. a. 2001] Vertegaal, Roel ; Slagter, Robert ; Veer, Gerrit van der ; Nijholt, Anton: Eye gaze patterns in conversations: There is More to Conversational Agents Than Meets the Eyes. In: *SIGCHI conference on Human factors in computing systems*. New York : ACM Press, 2001, S. 301–308. – ISBN 1581133278

[Vertegaal u. a. 2000] Vertegaal, Roel ; Veer, Gerrit van der ; Vons, Harro: Effects of Gaze on Multiparty Mediated Communication. In: *Proceedings of Graphics Interface*. Montreal and Canada : Morgan Kaufmann Publishers, 2000, S. 95–102

[Waizenegger u. a. 2011] Waizenegger, Wolfgang ; Feldmann, Ingo ; Schreer, Oliver: Real-time Patch Sweeping for High-Quality Depth Estimation in 3D Videoconferencing Applications. In: *Proceedings SPIE - Real-time image and video processing 2011*. Bellingham and Wash and Springfield and Va : SPIE and IS&T, 2011. – ISBN 9780819484086

[Wang 2004] Wang, Kun: Adaptive stereo matching algorithm based on edge detection. In: *IEEE International Conference on Image Processing*, 2004, S. 1345–1348

[Wang u. a. 2007] Wang, Liang ; Liao, Miao ; Gong, Minglun ; Yang, Ruigang ; Nister, David: High-Quality Real-Time Stereo Using Adaptive Cost Aggregation and Dynamic Programming. In: *International Symposium on 3D Data Processing, Visualization, and Transmission*, 2007, S. 798–805. – ISBN 9780769528250

[Wang u. a. 2004] Wang, Zhou ; Bovik, Alan C. ; Sheikh, Hamid R. ; Simoncelli, Eero P.: Image Quality Assessment: From Error Visibility to Structural Similarity. In: *IEEE Transactions on Image Processing* 13 (2004), Nr. 4, S. 600–612

[Wang u. a. 2003] Wang, Zhou ; Simoncelli, Eero P. ; Bovik, Alan C.: Multiscale structural similarity for image quality assessment. In: *Thirty-Seventh Asilomar Conference on Signals, Systems & Computers*. Piscataway and N.J : IEEE, 2003, S. 1398–1402. – ISBN 9780780381049

[Waschbüsch u. a. 2005] Waschbüsch, Michael ; Würmlin, Stephan ; Cotting, Daniel ; Sadlo, Filip ; Gross, Markus: Scalable 3D video of dynamic scenes. In: *The Visual Computer* 21 (2005), Nr. 8-10, S. 629–638

[Weigel u. a. 2010] Weigel, Christian ; Berghaus, Dirk ; Kühhirt Uwe: Development and Analysis of a Small- and a Large-Scale Multi-Camera System. In: *Internation Conference on 3D systems and Applications*, 2010, S. 219–222

[Weigel und Kreibich 2006] Weigel, Christian ; Kreibich, Leif L.: Advanced 3D Video Object Synthesis Based on Trilinear Tensors. In: *IEEE Tenth International Symposium on Consumer Electronics*. [Piscataway and NJ] : IEEE, 2006, S. 1–5. – ISBN 9781424402168

[Weigel und Schübel 2007] Weigel, Christian ; Schübel, Peter: Trifocal Transfer on Commodity Graphics Hardware. In: Domański, Marek (Hrsg.) ; Stasiński, Ryszard (Hrsg.) ; Bartkowiak, Maciej (Hrsg.): *European Signal Processing Conference*, 2007, S. 1686–1690. – ISBN 978-83-921340-2-2

[Weigel und Treutner 2011] Weigel, Christian ; Treutner, Niklas: Flexible OpenCL Accelerated Disparity Estimation for Video Communication Applications. In: *3DTV Conference*. [Piscataway and N.J.] : IEEE, 2011, S. 1–4. – ISBN 1612841619

[Weigel und Treutner 2012] Weigel, Christian ; Treutner, Niklas: Establishing eye contact for home video communication using stereo analysis and free viewpoint synthesis. In: *Three-Dimensional Image Processing (3DIP) and Applications II, Atilla M. Baskurt; Robert Sitnik, Editors, 829003*, 2012. – ISBN 9780819489371

[Weigel u. a. 2007] Weigel, Christian ; Werner, Stefan ; Schübel, Peter: A Real-Time Image-Based Rendering Framework. In: *3DTV Conference*, 2007, S. 1–4. – ISBN 9781424407224

[Weiher und Wagner 1991] Weiher, Sigfrid v. ; Wagner, Bernhard: *Tagebuch der Telekommunikation: Von 1600 bis zur Gegenwart*. 2. Berlin and and Offenbach : Vde-Verl., 1991. – ISBN 9783800716869

[Wilcox und Gibson 2005] Wilcox, James R. ; Gibson, David K.: *Video communications: The whole picture*. 4. San Francisco : CMP Books, 2005. – ISBN 9781578203161

[Wimmer 2005] Wimmer, Peter: Stereoscopic player and stereoscopic multiplexer: a computer-based system for stereoscopic video playback and recording. In: Andrew J. Woods, Mark T. Bolas John O. Merritt Ian E. M. (Hrsg.): *Stereoscopic Displays and Virtual Reality Systems XII*, 2005, S. 400–411

[Wolf u. a. 2010] Wolf, Lior ; Freund, Ziv ; Avidan, Shai: An eye for an eye: A single camera gaze-replacement method. In: *IEEE Conference on Computer Vision and Pattern Recognition*. New York : IEEE Comput. Soc. Press, 2010, S. 817–824. – ISBN 9781424469840

[Würmlin u. a. 2004] Würmlin, Stephan ; Lamboray, Edouard ; Gross, Markus: 3D video fragments: dynamic point samples for real-time free-viewpoint video. In: *Computers & Graphics* 28 (2004), Nr. 1, S. 3–14. – ISSN 00978493

[Würmlin u. a. 2005] Würmlin, Stephan ; Lamboray, Edouard ; Waschbüsch, Michael ; Kaufmann, Peter ; Smolic, Aljoscha ; Gross, Markus: Image-space Free-viewpoint Video. In: *Proceedings of Vision, Modeling, Visualization '05*, 2005, S. 453–460

[Xiao und Shah 2003] Xiao, Jiangjian ; Shah, Mubarak: From Images to Video: View Morphing of Three Images. In: *Vision, modeling, and visualization 2003*. Berlin and Amsterdam : Akad. Verl.-Ges. Aka and IOS Press, 2003, S. 495–502. – ISBN 3-89838-048-3

[Xiao und Shah 2004] Xiao, Jiangjian ; Shah, Mubarak: Tri-view morphing. In: *Computer Vision and Image Understanding* 96 (2004), Nr. 3, S. 345–366. – ISSN 10773142

[Yang u. a. 2003] Yang, Ruigang ; Welch, Greg ; Bishop, Gary: Real-Time Consensus-Based Scene Reconstruction Using Commodity Graphics Hardware. In: *Computer Graphics Forum* 22 (2003), Nr. 2, S. 207–216. – ISSN 0167-7055

[Yang und Zhang 2002] Yang, Ruigang ; Zhang, Zhengyou: Eye Gaze Correction with Stereovision for Video-Teleconferencing. In: *European Conference on Computer Vision* Bd. 2351. Berlin and Heidelberg : Springer Berlin Heidelberg, 2002, S. 479–494. – ISBN 978-3-540-43744-4

[Yarbus 1967] Yarbus, Alfred L.: *Eye Movements and Vision*. New York : Plenum Press, 1967

[Yoon und Kweon 2005] Yoon, Kuk-Jin ; Kweon, In S.: Locally Adaptive Support-Weight Approach for Visual Correspondence Search. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005, S. 924–931. – ISBN 9780769523729

[Yoon und Kweon 2006] Yoon, Kuk-Jin ; Kweon, In S.: Adaptive support-weight approach for correspondence search. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28 (2006), Nr. 4, S. 650–656

[Yoon und Kweon 2007] Yoon, Kuk-Jin ; Kweon, In S.: Stereo Matching with the Distinctive Similarity Measure. In: *IEEE International Conference on Computer Vision*, 2007, S. 1–7. – ISBN 9781424416318

[Yu u. a. 2010] Yu, Wei ; Chen, Tsuhan ; Franchetti, Franz ; Hoe, James C.: High Performance Stereo Vision Designed for Massively Data Parallel Platforms. In: *IEEE Transactions on Circuits and Systems for Video Technology* 20 (2010), Nr. 11, S. 1509–1519. – ISSN 1051-8215

[Zabih und Woodfill 1994] Zabih, Ramin ; Woodfill, John: Non-parametric local transforms for computing visual correspondence. In: *European Conference on Computer Vision*, 1994, S. 151–158

[Zhang 2000] Zhang, Zhengyou: A flexible new technique for camera calibration. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22 (2000), Nr. 11, S. 1330–1334

[Zhang und Yang 2004] Zhang, Zhengyou ; Yang, Ruigang: *Video-Teleconferencing System With Eye-Gaze Correction.* 2004

[Zitnick u. a. 2004] Zitnick, C. L. ; Kang, Sing B. ; Uyttendaele, Matthew ; Winder, Simon ; Szeliski, Richard: High-quality video view interpolation using a layered representation. In: *ACM Transactions on Graphics* 23 (2004), Nr. 3, S. 600

# Abbildungsverzeichnis

# Tabellenverzeichnis

# A. Anhang

| | 09-02 | 09-03 | 11-02 | 11-03 | 12-02 | 12-03 | Orig |
|---|---|---|---|---|---|---|---|
| **max** | 14.03 | 16.27 | 19.67 | 25.05 | 16.02 | 15.05 | 83.43 |
| **∅ Akzeptanzrate (%)** | 10.58 | 12.59 | 15.26 | 20.30 | 12.19 | 10.96 | 79.40 |
| **min** | 7.73 | 9.49 | 11.51 | 16.10 | 9.00 | 7.67 | 74.87 |
| **max** | 14.51 | 18.56 | 20.31 | 24.93 | 19.08 | 16.67 | 80.74 |
| **∅ Akzeptanzrate (BK) (%)** | 9.55 | 13.07 | 14.44 | 18.18 | 12.21 | 10.29 | 74.87 |
| **min** | 5.85 | 8.72 | 9.74 | 12.62 | 7.15 | 5.74 | 68.25 |
| **max** | 16.92 | 17.50 | 23.80 | 29.56 | 17.11 | 17.40 | 89.95 |
| **∅ Akzeptanzrate (kein BK) (%)** | 11.62 | 12.12 | 16.42 | 22.42 | 12.17 | 11.52 | 84.85 |
| **min** | 7.51 | 7.92 | 10.58 | 16.31 | 8.24 | 7.08 | 78.45 |
| **max** | 70.66 | 72.86 | 66.28 | 73.24 | 77.21 | 65.09 | 91.28 |
| **∅ Augenkontaktrate (BK) (%)** | 63.96 | 66.33 | 59.14 | 66.06 | 69.47 | 56.62 | 86.93 |
| **min** | 56.83 | 59.31 | 51.71 | 58.29 | 60.82 | 47.85 | 81.44 |
| **max** | 23.60 | 25.38 | 35.45 | 25.90 | 23.87 | 28.90 | 36.66 |
| **∅ Augenkontaktrate (kein BK) (%)** | 17.59 | 19.19 | 27.07 | 19.02 | 18.26 | 21.82 | 29.09 |
| **min** | 12.57 | 13.95 | 19.73 | 13.30 | 13.49 | 15.77 | 22.29 |

**Tabelle A.1.:** Akzeptanz- und Augenkontaktraten und Konfidenzintervallgrenzen (Clopper-Pearson) per Algorithmus Experiment 1.

| | 211 | 212 | 216 | 218 | 231 | 234 | 238 | 240 |
|---|---|---|---|---|---|---|---|---|
| **max** | 4.94 | 24.52 | 10.04 | 3.99 | 53.81 | 34.70 | 7.53 | 7.46 |
| **∅ Akzeptanzrate (ohne Ref.) (%)** | 2.43 | 19.34 | 6.62 | 1.73 | 47.33 | 28.38 | 4.28 | 4.00 |
| **min** | 0.98 | 14.84 | 4.09 | 0.56 | 40.91 | 22.64 | 2.16 | 1.85 |
| **max** | 96.37 | 87.69 | 96.21 | 74.00 | 94.43 | 94.29 | 89.76 | 90.17 |
| **∅ Akzeptanzrate (nur Ref.) (%)** | 87.10 | 78.46 | 88.64 | 61.40 | 85.37 | 85.00 | 79.59 | 78.38 |
| **min** | 70.17 | 66.51 | 75.44 | 47.57 | 70.83 | 70.16 | 65.66 | 61.79 |
| **max** | 1.92 | 76.97 | 22.75 | 46.04 | 68.38 | 74.81 | 8.46 | 83.83 |
| **∅ Augenkontaktrate (ohne Ref.) (%)** | 0.35 | 71.69 | 17.94 | 40.14 | 62.24 | 68.86 | 5.04 | 78.67 |
| **min** | 0.01 | 65.94 | 13.77 | 34.44 | 55.79 | 62.41 | 2.71 | 72.73 |
| **max** | 21.42 | 92.37 | 49.92 | 89.95 | 79.92 | 97.21 | 19.60 | 99.93 |
| **∅ Augenkontaktrate (nur Ref.) (%)** | 6.45 | 84.62 | 34.09 | 80.70 | 65.85 | 90.00 | 8.16 | 97.30 |
| **min** | 0.79 | 73.52 | 20.49 | 68.09 | 49.41 | 76.34 | 2.27 | 85.84 |

**Tabelle A.2.:** Akzeptanz- und Augenkontaktraten und Konfidenzintervallgrenzen (Clopper-Pearson) per Sequenz Experiment 1.

| | 05-02 | 05-04 | 05-05 | orig |
|---|---|---|---|---|
| **max** | 14.02 | 21.84 | 14.32 | 83.43 |
| ∅ **Akzeptanzrate (%)** | 11.59 | 18.55 | 11.63 | 79.40 |
| **min** | 9.44 | 15.56 | 9.29 | 74.87 |
| **max** | 14.84 | 20.47 | 15.65 | 80.74 |
| ∅ **Akzeptanzrate (BK) (%)** | 11.31 | 16.19 | 11.24 | 74.87 |
| **min** | 8.37 | 12.50 | 7.71 | 68.25 |
| **max** | 15.47 | 27.06 | 15.51 | 89.95 |
| ∅ **Akzeptanzrate (kein BK) (%)** | 11.87 | 21.64 | 11.90 | 84.85 |
| **min** | 8.85 | 16.86 | 8.88 | 78.45 |
| **max** | 69.84 | 67.48 | 68.73 | 91.28 |
| ∅ **Augenkontaktrate (BK) (%)** | 65.15 | 62.39 | 62.92 | 86.93 |
| **min** | 60.23 | 57.10 | 56.82 | 81.44 |
| **max** | 22.56 | 30.96 | 24.02 | 36.66 |
| ∅ **Augenkontaktrate (kein BK) (%)** | 18.39 | 25.28 | 19.75 | 29.09 |
| **min** | 14.70 | 20.16 | 15.93 | 22.29 |

**Tabelle A.3.:** Akzeptanz- und Augenkontaktraten und Konfidenzintervallgrenzen (Clopper-Pearson) per Disparitätsalgorithmus Experiment 1.

| | 06 | 07 | orig |
|---|---|---|---|
| **max** | 16.70 | 16.90 | 83.43 |
| ∅ **Akzeptanzrate (%)** | 14.27 | 14.59 | 79.40 |
| **min** | 12.07 | 12.49 | 74.87 |
| **max** | 15.11 | 17.35 | 80.74 |
| ∅ **Akzeptanzrate (BK) (%)** | 11.99 | 14.00 | 74.87 |
| **min** | 9.32 | 11.08 | 68.25 |
| **max** | 21.26 | 18.50 | 89.95 |
| ∅ **Akzeptanzrate (kein BK) (%)** | 17.21 | 15.15 | 84.85 |
| **min** | 13.64 | 12.20 | 78.45 |
| **max** | 67.79 | 67.83 | 91.28 |
| ∅ **Augenkontaktrate (BK) (%)** | 63.62 | 63.60 | 86.93 |
| **min** | 59.29 | 59.21 | 81.44 |
| **max** | 32.60 | 23.64 | 36.66 |
| ∅ **Augenkontaktrate (kein BK) (%)** | 27.93 | 19.96 | 29.09 |
| **min** | 23.59 | 16.63 | 22.29 |

**Tabelle A.4.:** Akzeptanz- und Augenkontaktraten und Konfidenzintervallgrenzen (Clopper-Pearson) per Synthesealgorithmus Experiment 1.

| | 11-05BG | 11_05 | 16-06BG | 16_06 | 16-16BG | 16_16 | 22-06BG | 22_06 | 22-16BG | 22_16 | 11-05 o. VK |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 11-05BG | 1 | 0.577 | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | 0.013 |
| 11_05 | 0.577 | 1 | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **0.000** | **< 0.0001** | 0.002 |
| 16-06BG | **< 0.0001** | **< 0.0001** | 1 | 0.995 | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** |
| 16_06 | **< 0.0001** | **< 0.0001** | 0.995 | 1 | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** |
| 16-16BG | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | 1 | 0.259 | 0.648 | 0.472 | **< 0.0001** | **0.000** | **< 0.0001** |
| 16_16 | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | 0.259 | 1 | 0.502 | 0.682 | **0.000** | 0.008 | **< 0.0001** |
| 22-06BG | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | 0.648 | 0.502 | 1 | 0.793 | **< 0.0001** | **0.001** | **< 0.0001** |
| 22_06 | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | 0.472 | 0.682 | 0.793 | 1 | **< 0.0001** | 0.002 | **< 0.0001** |
| 22-16BG | **< 0.0001** | **0.000** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **0.000** | **< 0.0001** | **< 0.0001** | 1 | 0.401 | **< 0.0001** |
| 22_16 | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **0.000** | 0.008 | **0.001** | 0.002 | 0.401 | 1 | **< 0.0001** |
| 11-05 o. VK | 0.013 | 0.002 | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | **< 0.0001** | 0 |

**Tabelle A.5.:** Signifikante Unterschiede in der Qualitätsbewertung als Ergebnisse des paarweisen Vergleichs in Experiment 2